Proceedings

The 11th International Conference on Service Science

ICSS 2018

11-13 May 2018
Shanghai University, Shanghai, China

Conference Committee
ICSS 2018

Honorary Chairs
Junliang Chen, Professor of BUPT, Academician of Chinese Academy of Sciences, China

Conference Co-Chairs
Xiaofei Xu, Vice President, Harbin Institute of Technology
Xiaowei Shen, Director, IBM Research China
Hua Ouyang, Vice President, Shanghai University, China

Steering Committee
Guy Doumeingts, Professor, University of Bordeaux I, France
Xiucheng Fan, Professor, Fudan University, China
Yushun Fan, Professor, Tsinghua University, China
Zhiyong Feng, Professor, Tianjin University, China
Yanbo Han, Professor, North China University of Technology, China
Keqing He, Professor, Wuhan University, China
Ying Huang, Professor, Lenovo Group / Peking University, China
Changjun Jiang, Professor, Donghua University, China
Hai Jin, Professor, Huazhong University of Science and Technology, China
Bing Li, Professor, Wuhan University, China
Furen Lin, Professor, National Tsinghua University, Taiwan
Rong N. Chang, IBM T.J. Watson Research Center, USA
Xiaowei Shen, Director, IBM Research China, China
James Spohrer, IBM Global University Programs, USA
Sen Su, Professor, Beijing University of Posts and Telecommunications, China
Jun Wei, Professor, Institute of Software, China Academy of Sciences, China
Zhaohui Wu, Professor, Zhejiang University, China
Zhonghai Wu, Professor, Peking University, China
Xiaofei Xu, Professor, Harbin Institute of Technology, China
Jianwei Yin, Professor, Zhejiang University, China
Liangjie Zhang, VP, Kingdee Research, China
Jie Zhou, Professor, Tsinghua University, China

Program Committee Co-Chairs
Zhiyong Feng, Professor, Tianjin University, China
Zhongjie Wang, Professor, Harbin Institute of Technology, China
Bofeng Zhang, Professor, Shanghai University, China

Program Committee Members
Buqing Cao, Hunan University of Science and technology, China
Liang Chen, Sun Yat-sen University, China
Shizhan Chen, Tianjin University, China
Shuiguang Deng, Zhejiang University, China
Zhijun Ding, Tongji University, China
Yucong Duan, Hainan University, China
Xiaoliang Fan, Lanzhou University, China
Xiaodong Fu, Kunming University of Science and Technology, China
Honghao Gao, Shanghai University, China
Ting He, Huaqiao University, China
Weimin Li, Shanghai University, China
Weiping Li, Peking University, China
Fangfang Liu, Shanghai University, China
Jianxun Liu, Hunan Univeristy of Science and Technology, China
Shijun Liu, Shandong University, China
Zhihui Lu, Fudan University, China
Yutao Ma, Wuhan University, China
Tong Mo, Peking University, China
Sen Niu, Shanghai University, China
Weifeng Pan, Zhejiang Gongshang University, China
Lianyong Qi, Qufu Normal University, China
Wei Song, Nanjing University of Science and Technology, China
Chang-ai Sun, University of Science and Technology Beijing, China

Mingdong Tang, Guangdong University of Foreign Studies, China
Zhiying Tu, Harbin Institute of Technology China, China
Hao Wang, Jiangxi Normal University, China
Jian Wang, Wuhan University, China
Pengwei Wang, Donghua University, China
Shangguang Wang, Beijing University of Posts and Telecommunications, China
Zhongjie Wang, Harbin Institute of Technology, China
Junhao Wen, Chongqing University, China
Shaochun Wu, Shanghai University, China
Yunni Xia, Chongqing University, China
Liang Zhang, Fudan University, China
Pengcheng Zhang, Hohai University, China
Yiwen Zhang, Anhui University, China
Zhongbao Zhang, Beijing University of Posts and Telecommunication, China
Zhuofeng Zhao, North China University of Technology, China
Ao Zhou, Beijing University of Posts and Telecommunications, China
Zhangbing Zhou, China University of Geosciences, China
Guobing Zou, Shanghai University, China

Organization Committee Co-Chairs

Guobing Zou, Associate Professor, Shanghai University, China

Organization Committee Members

Honghao Gao, Shanghai University, China
Ming Jiang, Shanghai University, China
Wang Li, Shanghai University, China
Sen Niu, Shanghai University, China
Shengye Pang, Shanghai University, China
Zhen Qin, Shanghai University, China
Farhan Ullah, Shanghai University, China
Hao Wu, Shanghai University, China
Yang Xiang, Shanghai University, China
Xia Zhang, Shanghai University, China

ICSS 2018
Proceedings

Table of Contents

2018 11th International Conference on Service Science

Mobile Edge Computing-Assisted Biometric Services

Chuntao Ding
State Key Laboratory of
Networking and Switching
Technology, Beijing University
of Posts and Telecommunications,
Beijing, China
Email: ctding@bupt.edu.cn

Shangguang Wang
State Key Laboratory of
Networking and Switching
Technology, Beijing University
of Posts and Telecommunications,
Beijing, China
Email: sgwang@bupt.edu.cn

***Abstract*—Biometric technology has attracted increasing attention from both academia and industry. Utilizing biometric technology to provide biometric services with high quality of service (QoS) is particularly important. Available mature services, such as face recognition and fingerprint recognition, demonstrate how biometric services facilitate our work and study, and help in our daily lives. As an emerging technology, mobile edge computing provides an opportunity to provide biometric services with high QoS. In this paper, we propose a mobile edge computing-assisted biometric services framework and discuss why mobile edge computing can help improve the QoS of biometric services. In addition, we implement a prototype system of mobile edge computing-assisted biometric services and validate the proposed framework based on a real network environment and database. It is demonstrated that the proposed framework can reduce not only response time but also significantly reduce network traffic within the core network.**

***Index Terms*—Biometric services, mobile edge computing, base station, edge servers, cloud computing**

I. Introduction

Biometric technology, which refers to technology focused on recognition of individuals based on their physical characteristics, has emerged and been researched for the past few decades. Unlike conventional recognition technologies (such as passwords or ID cards), biometric technologies based on faces, gaits, irises, or fingerprints (as shown in Fig. 1) are based on 'who you are' rather than 'what you know' or 'what you have,' and such biometric identifiers are difficult to guess, copy, or forge [1]. This makes biometrics more difficult to abuse than conventional recognition technologies. Therefore, biometrics have been successfully applied in many fields. For example, the automated fingerprint identification system (AFIS) has been used by the United States and Canada since the late 1970s and early 1980s, respectively [2].

Deep learning techniques [3], [4] have become increasingly popular in biometric applications owing to their ability to achieve high accuracy for biometric identification tasks, such as face recognition. For example, on the LFW data set (http://vis-www.cs.umass.edu/lfw/), by using deep learning techniques, Sun et al. achieved 97.45% verification accuracy [5] and Liu. et al. achieved 99.77% pairwise verification accuracy [6]. Therefore, biometrics have increasingly made use of deep learning techniques to learn effective feature representations to improve accuracy.

In recent years, we have witnessed an dramatic increase in mobile devices, especially smartphones. Recent analysis shows that the global revenue from smartphone sales in 2016 amounted to 435.1 billion U.S. dollars (https://www.statista.com/statistics/237505/global-revenue-from-smartphones-since-2008/). In addition, communication mobile devices have become new sensing platforms, usually equipped with multiple sensors. These sensors enable us to collect a variety of biometric data in the form of images or video, which are widely used to identify a specific user through unique characteristics. It is known that using deep learning techniques can significantly improve the accuracy of biometric identification based on large amounts of biometric data. Owing to the ubiquitous nature of mobile devices, it is now possible to use these mobile devices to provide biometric services with high quality of service (QoS) anytime and anywhere.

The way we provide and consume biometric services is ever-changing, and mobile cloud computing (MCC) has become an emerging technology [7]. With the help of MCC, biometric applications are deployed on remote cloud servers. Mobile users invoke the biometric applications remotely through their mobile devices. With their powerful computing and storage capabilities, cloud servers can provide enough computation ability to learn effective features by using deep learning techniques and offer sufficient storage capacity to store large amounts of biometric data. However, some problems are incurred, e.g., an enormous amount of network traffic and network transmission delay, because of the frequent interaction between mobile devices and cloud servers, with mobile devices transmitting huge amounts of data to the remote cloud servers.

Motivation: To clearly illustrate the problem, we give an example of using biometrics to assist law enforcement that recognizes photographs of suspects from the police suspect database to help the police narrow down potential suspects under the assumption that the suspect's data are stored on the cloud servers. With the help of cloud computing, pervasive mobile devices, and deep learning techniques, highly accurate

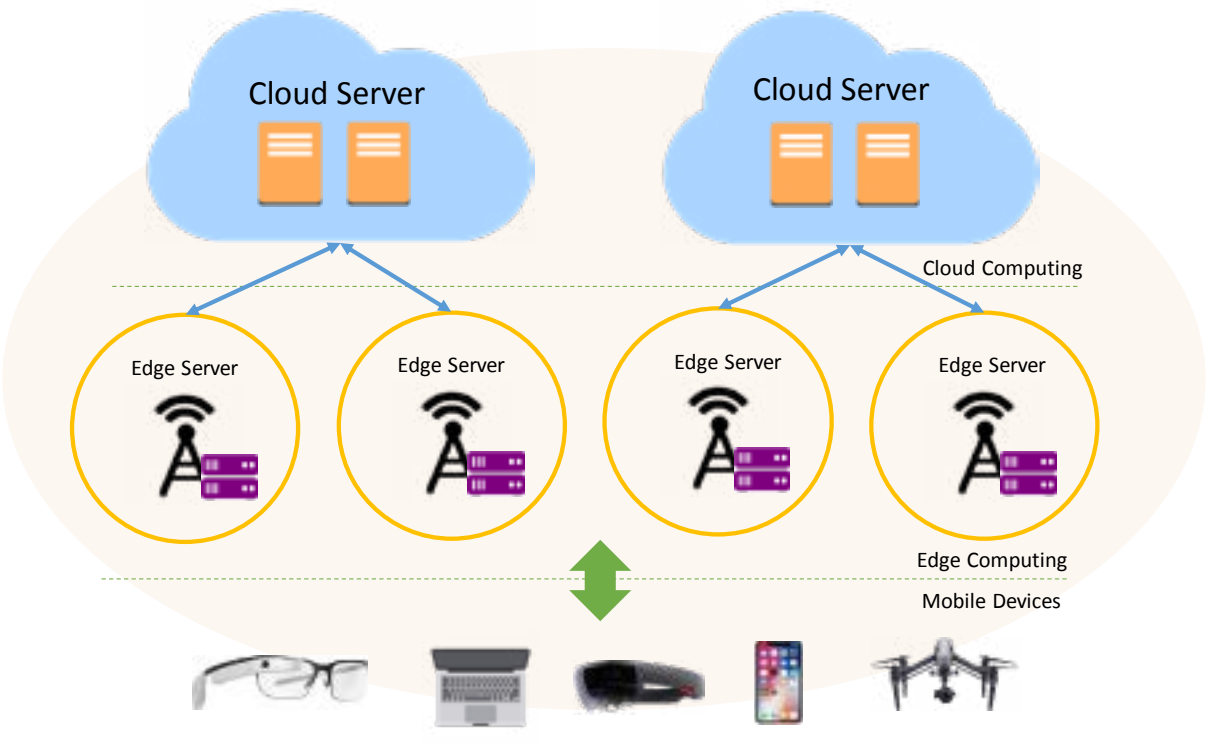


Fig. 2. A three-layer federated environment.

Fig. 1. Biometric recognition examples.

identification results can be generated through training a complex and effective model (which includes parameters such as layers, network structure, and labels) and biometric services can be provided anytime and anywhere. This is the most classic use of current biometric services. For example, when a suspect escapes, it is very much possible that images of this suspect can be captured by mobile devices because of their ubiquity. Then, the mobile devices transmits the captured data to the remote cloud servers for processing through a mobile Internet. Finally, the results are generated through large-scale computing. The cloud servers transmit the results to the police and tell them where to catch the suspect.

However, there are some challenges faced by providing biometric services based on the collaboration of mobile devices and cloud servers. Because cloud servers are usually far away from the mobile users, transmitting the service request and data to the remote cloud servers incurs long network transmission time. However, assisting the police narrow down potential suspects requires immediate analysis of, or response to, collected sensing data. In addition, mobile devices can capture large amounts of biometric data and frequently transmit the data to the remote cloud, which can cause an enormous amount of network traffic that will increase the load on the core network. The burden of data to upload toward the remote cloud servers leads to inefficient use of bandwidth, and a recent study by Cisco shows that total network traffic will triple by 2019, worsening the situation further [8].

Therefore, there is a pressing need to redesign the MCC architecture to serve biometric services with better efficiency. Mobile edge computing (MEC) [9], [10] is envisioned as offering a hybrid scheme. As shown in Fig. 2, mobile devices, edge computing, and cloud computing form a three-layer service delivery model. In addition, with the development

of 5G, which aims at lower latency than 4G equipment, for better implementation of the Internet of Things, MEC has demonstrated its advantages of reducing latency in many applications (e.g., Nokia and China Mobile now provide real-time services for Shanghai racing events and have reduced the delay to <500 ms). Therefore, it is imperative to use mobile edge computing to assist biometric services.

Our contributions: In this paper, considering the characteristics of mobile edge computing and biometric technologies comprehensively, we propose a mobile edge computing-assisted biometric services framework and implement a prototype system. In addition, based on a developed prototype system, extensive experiments with a real mobile edge computing network environment and a real-world database are conducted. It is demonstrated that the proposed framework can reduce the response time, especially transmission time, and significantly reduce the network traffic within the core network.

II. Related Technologies

A. Deep Learning

In 2006, a breakthrough of deep learning was made by Geoffrey Hinton [3], and this work was quickly followed up by Yann LeCun and Yoshua Bengio [4]. Deep learning, also known as deep structured learning, can improve the accuracy of biometric identifications by enabling the learning of effective feature representations of data.

Deep learning is suitable handling large amounts of data sets. With the help of large amounts of biometric data and the powerful computational resources of cloud servers, it is possible to train more powerful statistical models to reveal the intrinsic biometric data. These models have drastically improved the robustness of biometrics, such as face recognition, iris recognition, and fingerprint identification.

B. Mobile Edge Computing

Mobile edge computing was proposed by the European Telecommunications Standard Institute (ETSI) in 2014 as a means to provide computing capabilities and storage capacity at the edge of the mobile network [10]. Mobile edge computing enables a seamless integration of the cloud computing functionalities into the mobile network. It entails placing a number of small-scale servers at the network edge and dispersing partial data storage, processes, and applications on those small-scale servers rather than transmitting almost the entire sets of data and processes to the remote cloud servers. With the help of MEC, more applications can be run with significant reductions of the load on the core network and communication delay and improvement of the QoS.

III. Mobile Edge Computing-Assisted Biometric Services Framework

In this section, we present a mobile edge computing-assisted biometric services framework by integrating mobile edge computing and biometric services. The framework consists of three components: mobile devices, edge servers, and cloud servers, as shown in Fig. 3.

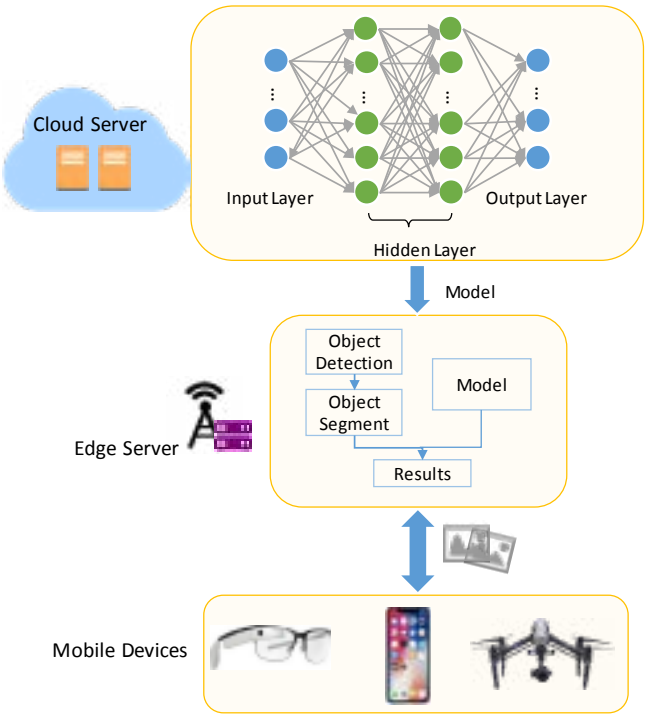


Fig. 3. Mobile edge computing-assisted biometric services framework. The detailed processes are as follows: The model is first trained on the cloud servers based on huge amounts of biometric data and deep learning techniques. Then, the model is transmitted to the edge servers and saved. When a user has a need for requesting biometric services, the user transmits the request and data to the edge servers rather than to the remote cloud servers. When the request and data arrive at the edge servers, the edge servers process the request and generate the results through the model. Finally, the results are returned to the user.

In the proposed framework, mobile device, edge servers, and cloud servers provide biometric services with high QoS in a collaborative manner. Mobile users use their mobile devices to collect biometric data as the source data. After capturing the biometric data, the mobile devices transmit these data to the edge servers for processing through an LTE base station. With the help of cloud computing, large amounts of biometric data, and biometric data analysis techniques based on deep learning, we can generate accurate identification results through training a complex and effective model (which includes parameters such as layers, network structure, and labels). Finally, the model gets transmitted to the edge servers through the Internet and saved. Biometric data arriving at the edge servers are first preprocessed to generate clear segment objects. Then, the results are generated through the model. Finally, the results are returned to the mobile users.

The main characteristics and functions of each component are described in the following.

A. Mobile Devices

For convenience, in our work, we assume the mobile devices are smartphones. Equipped with multiple sensors, smartphones make it easy to capture a large amount of biometric data

but have limited computation capability and storage capacity and restricted power. Operations on smartphone include the following:

- capturing biometric data,
- transmitting the biometric data to the edge servers through an LTE base station in raw form, and
- receiving the results from the edge servers.

B. Edge Servers

In our work, we consider the base station servers as edge servers. To provide real-time biometric service, it is necessary to avoid excessive data transmission to reduce communication delays. Because the edge servers are in proximity to the mobile devices and their resource and computation abilities are intermediate between those of mobile devices and remote cloud servers, operations with small amounts of computation can be conducted on the edge servers; these include the following:

- executing an object detection algorithm to detect the object,
- removing unrelated regions,
- receiving the training model transmitted from the cloud servers and saving it, and
- generating the results through the model and transmitting them to the mobile users.

C. Cloud Servers

Cloud servers refer to resource-rich cloud infrastructure; they are considered to have unlimited computation capability and unlimited storage capacity. Because the cloud servers provide cheaper and virtually unlimited storage capacity and unlimited computing power, large amounts of biometric data can be saved and the most time-consuming operations (such as training the model) can be performed. Operations on the cloud servers include the following:

- storing large amounts of biometric data,
- training the model by using deep learning techniques based on these biometric data, and
- transmitting the model to the edge servers.

The mobile edge computing-assisted biometric services framework makes full use of the power of network edge servers, undertaking partial computation tasks. The comprehensive characteristics of mobile devices, edge servers, and cloud servers enable them to assist biometric services through reasonable collaboration, providing biometric services with high QoS.

IV. Advantage of Mobile Edge Computing

Mobile edge computing can help biometrics provide biometric services with high QoS. In the mobile edge computing-assisted biometric services framework, mobile devices, edge servers, and cloud servers collaborate to complete tasks. To explain why mobile edge computing can assist biometrics in providing biometric services with high QoS, in terms of the service providers and consumers and [11], we first divide the traditional biometric services into cloud-to-mobile and mobile-to-mobile patterns. Then, we compare them with the mobile edge computing pattern.

A. Cloud-to-Mobile Pattern

The cloud-to-mobile (C2M) pattern is commonly used today; it provides biometric services by combining cloud servers and mobile devices. Biometric applications are deployed on the remote cloud servers, while the mobile device acts like a thin client to invoke the biometric applications through a mobile Internet. In this pattern, mobile users make requests and data are delivered to the remote cloud servers through the mobile Internet. Then, the cloud server processes the requests and generates the results. Finally, the results are returned to the mobile users through the mobile Internet.

The C2M pattern offers two advantages:

- *Capability extension.* The C2M pattern supports mobile users to upload large amounts of biometric data to the cloud servers, where the data are processed and analyzed. Thus, the C2M pattern can augment the capabilities of mobile devices for resource-hungry applications.
- *Energy savings.* Mobile devices are recognized as resource-constrained devices. Offloading computation-intensive tasks to the cloud servers while only using the mobile device to perform less computationally intensive tasks—such as acquiring biometric data, transmitting biometric data to the cloud servers, and receiving the results—can significantly reduce energy consumption and keep mobile devices working as long as possible.

With these two advantages, the C2M pattern enables mobile users to invoke services in diverse environments. However, C2M pattern has three disadvantages:

- *High bandwidth consumption.* Uploading large amounts of biometric data to the remote cloud servers consumes significant bandwidth, increases the load on the core network, and causes network congestion.
- *Network transmission latency.* The C2M pattern requires transmissions from mobile users to remote cloud servers with distances ranging from tens to thousands of kilometers. Therefore, transmitting large amounts of biometric data to the remote cloud servers inevitably leads to communication latency, which cannot meet the demand of time-sensitive services, such as finding lost children or searching for terrorists.
- *Poor security and privacy.* Security and privacy are extremely important in the application of biometrics. Multiple mobile users interacting frequently with cloud servers compromises the safety of the data stored on the cloud servers. In addition, directly transmitting individual information to the cloud servers incurs individual information disclosure. For example, face images of an individual can be used for face recognition for banking services. Therefore, protecting the security and privacy of face images becomes critical. However, transmitting such images to the remote cloud servers in their raw form can

subject the images to copying from the cloud servers by an unauthorized person.

B. Mobile-to-Mobile Pattern

The mobile-to-mobile (M2M) pattern indicates entails deployment of biometric applications on mobile devices and delivery over wireless networks (such as Bluetooth and WLAN). In the M2M pattern, mobile devices play the role of service consumers or service providers. Users can consume biometric services deployed on their own mobile devices directly and also can consume biometric services deployed on peer mobile devices through free wireless networks. When users utilize biometric services from peer mobile devices through free wireless networks, they are moving in a limited range, such as within a museum or a classroom.

The M2M pattern offers two potential advantages:

- *Internet independence.* In the M2M pattern, mobile users have two situations in which they consume or provide biometric services. One is the case in which mobile users consume biometric services from their personal mobile devices directly, because the biometric application is deployed on their own mobile devices. The other situation is when mobile users consume biometric services from other peer mobile users through free wireless networks, because the biometric applications are deployed on their peer mobile users' devices. Neither case uses the mobile Internet, so they thus avoid any expenses for accessing the mobile Internet.
- *Service sharing.* Because the communication between mobile users is based on free wireless communication and they move in a limited range, mobile users can conveniently share their own biometric services with each other.

Benefitting from these two advantages, the M2M pattern makes it possible to use mobile devices to provide biometric services anywhere without the impact of the Internet. However, it has the following limitations:

- *Poor computational capability.* Mobile devices are recognized as resource-constrained computing devices, although the computational resources in mobile devices have become more powerful, they continue to remain incapable of running computationally complex operations.
- *Low storage capacity.* Although the storage capacity of mobile devices has improved, it still cannot meet the demand for storing large amounts of biometric data. Storage space of mobile devices is in the gigabyte range, whereas acquired biometric data is on the order of terabytes.
- *Limited power.* Providing biometric services anytime and anywhere causes high battery drain. Regardless of the fact that battery technology for mobile devices has improved to considerably lengthened standby time, this power limitation still poses a significant obstacle for users constantly using their mobile devices for providing services as long as possible.
- *Poor security and privacy.* Mobile devices connected with the physical world are entities that have a large amount of data to be shared with other devices. Owing to their limited computing power, it is hard to run complex encryption techniques on them, thus leading to poor security.

As can be seen from the above discussion, both C2M and M2M patterns have their advantages and limitations. It is worth noting that the mobile edge computing pattern not only inherits their strengths but also makes up for their shortcomings. These advantages are briefly described in the following.

- *Bandwidth savings.* Because the edge servers provide storage capacity and processing capability, after the model is trained on the cloud servers and transmitted to the edge servers, the recognized tasks can be done on the edge servers without needing to transmit all the biometric data and processing to the remote cloud servers. This can significantly reduce the load on the core network and save considerable bandwidth.
- *Low network communication latency.* Because the distance from the mobile users to the edge servers is typically $<$1 km, whereas a C2M pattern requires transmissions from mobile users to cloud servers with distances ranging from tens to thousands of kilometers, communication delays can be significantly reduced.
- *Improved security and privacy.* When we add a new image, we only store feature points trained on the edge servers, not on the cloud servers in the form of raw images; this can ensure privacy and security [12].

V. Experiments

In this section, we implement a prototype system to verify the feasibility of a mobile edge computing-assisted biometric services framework based on a real network environment and database.

A. Experiment Setup

The experiment environment consists of four parts: a mobile device, a base station, an edge server, and a cloud server. Fig. 4 illustrates the structure of the mobile edge computing setup and Fig. 5 illustrates the physical map of the mobile edge computing-assisted biometric services framework prototype. The parameters of each part are as follows:

1) Mobile Device: A Huawei Honor 8 smartphone is used as the mobile device. This smartphone is equipped with four Cortex A72 2.3 GHz processors and four Cortex A53 1.8 GHz processors and runs under Android 7.0.

*2) Base Station:*The base station is based on the Open Air Interface, and the hardware of the base station consists of three components: a radio-frequency signal generator, base station server A, and base station server B, respectively.

3) Edge Server: The edge server is a computer equipped with an Intel i5-4590 3.3 GHz CPU and 12 GB of RAM. The mobile device and edge server are connected through the LTE base station with an upload speed of 1000 kB/s and a downlink speed of 1.36 MB/s.

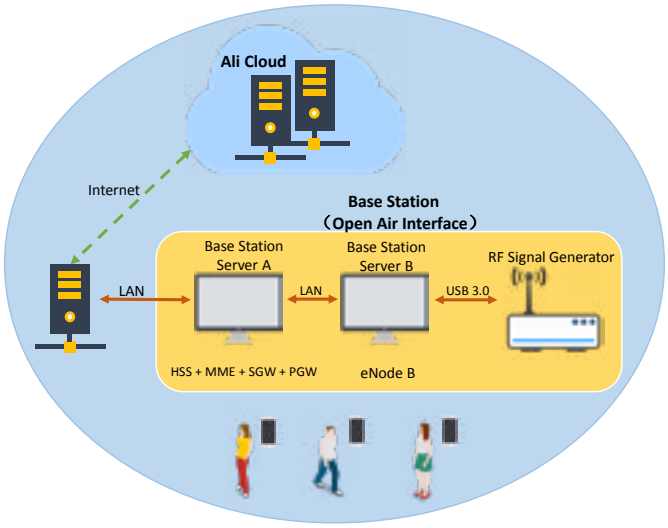


Fig. 4. Structure of mobile edge computing setup.

Fig. 5. Physical map of the mobile edge computing-assisted biometric services framework prototype.

4) Cloud Server: The cloud server is Alibaba Cloud (https://www.alibabacloud.com/). The edge server and cloud server are connected through an Internet backbone.

In our experiments, we used the algorithm proposed by He et al. [13] in 2015 to represent deep learning techniques, and we compared the proposed framework with the conventional MCC framework on a newly collected face database, the Lab_face database, which contains 420 subjects of 21 people with different facial expression and under variable lighting conditions for each individual. Because the subjects are collected by different mobile devices, they have different sizes. We choose five different sizes of images, which are related to the Lab_face database, to evaluate the response time. To avoid network jitter from affecting the response time, the results are averaged over 10 trials in our experiments.

B. Experiment Results

1) Response Time: The response time is defined as the time interval during which the users request to the recognized service and receive the results. Note that training the model by using larger amounts of biometric data and deep learning techniques is performed offline. In addition, periodically transmitting the model to the edge server also is not included in the response time.

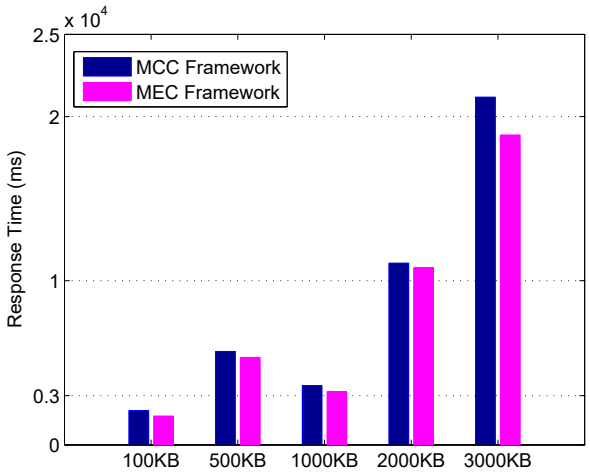


Fig. 6. Response time for different frameworks under different image sizes.

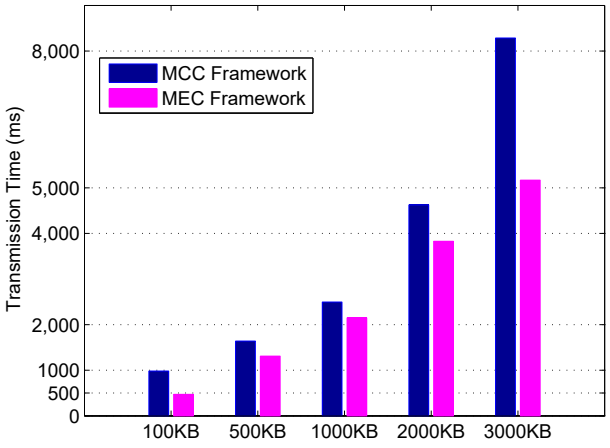


Fig. 7. Transmission time for different frameworks under different image sizes.

Fig. 6 shows that the response time of the MEC framework is shorter than that of the MCC framework. However, the response time of the MEC framework is not significantly reduced compared with that of the MCC framework. To more precisely analyze the response time, we divide the response time into two parts: transmission time and processing time. Fig. 7 shows the transmission time and Fig. 8 shows the processing time for the two frameworks under different image sizes, respectively.

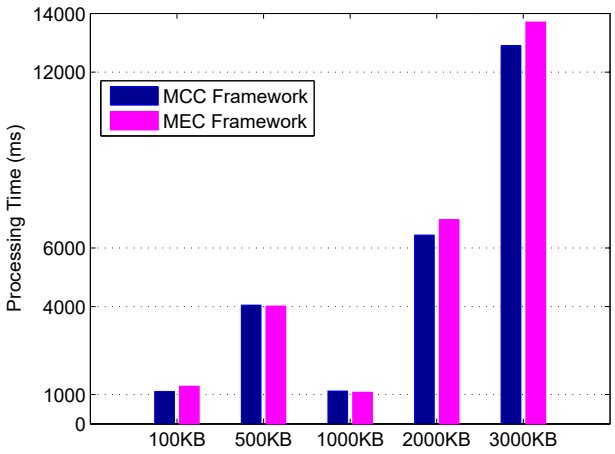


Fig. 8. Processing time for different frameworks under different image sizes.

From Fig. 7, it can be seen that the MEC framework can significantly reduce the average transmission time compared with that of the MCC framework. The main reason for this reduction is that, when we consume biometric services, we only transmit our requests and data to the edge servers rather than to the remote cloud servers. From Fig. 8, it can be seen that there is only a slight gap between the processing times of the two frameworks. However, the processing time occupies a large proportion of the response time, resulting in a slight gap between the response times of the two frameworks.

However, speeding up the processing time is not considered in this paper and can be seen as a later research topic. From Fig. 7, we can see that mobile edge computing can significantly reduce transmission delay.

2) Network Traffic: Fig. 9 shows that the amount of network

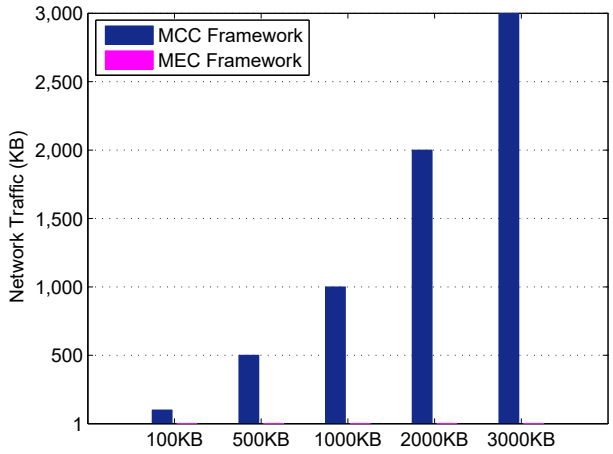


Fig. 9. Network traffic within the core network for different frameworks under different image sizes.

traffic within the core network of the MEC framework is much smaller than that within the MCC framework. In the MEC framework, the network traffic within the core network is 1 KB, whereas the network traffic within the core network of the MCC framework is the size of the transmitted image.

In the MEC framework, the model is trained on the cloud server. Then, the cloud server transmits the model to the edge server. Combined with the motivation of Section I, when the police need to request biometric services, such as to recognize an image, they will transmit the image to the edge server rather than to the remote cloud server. Therefore, the network traffic within the core network is zero. If the police want to add a suspect to the database, they also only need to transmit the suspect's image to the edge server for training based on the model, which will be a new model containing the new suspect. In this case, the network traffic within the core network is also zero. Moreover, if the police want to store the suspect in the cloud server, they only need to transmit the feature points after training on the edge server to the cloud server. In our experiment, the feature points of an image account for only 1 KB, and this situation occurs only when updating the database. At worst, every time a user consumes a biometric service, the user updates the database. In this case, the network traffic within the core network is 1 KB. However, in the MCC framework, because almost all processes are done on the cloud server, mobile users transmit the captured data to the remote cloud servers through the mobile Internet directly. Therefore, the network traffic within the core network is the size of the transmitted image.

It is known that transmitting large amounts of data can increase the load on the core network and even lead to network congestion. Compared with the MCC framework, the advantages of the MEC framework in terms of network traffic within the core network are clear.

VI. Conclusion

In this paper, we proposed a mobile edge computing-assisted biometric services framework that can provide biometric services with high QoS by fully utilizing the advantages of mobile edge computing. Under such framework, mobile devices, edge servers, and cloud servers act as providers of biometric services and collaborate to complete the user's service request. The advantages of the proposed framework have been demonstrated by a prototype system based on a real mobile edge computing environment and database; this framework can significantly improve the QoS of biometric services by reducing the response time and amount of network traffic.

In future work, using mobile edge computing, we plan to optimize the process and increase the speed of the process to further provide biometric services with higher QoS.

References

[1] K. J. Anil, "Technology Biometric Recognition", *Nature*, vol. 449, no. 7158, 2007, pp. 38-40.

[2] A. Kochetkov, "Cloud-based biometric services: just a matter of time", *Biometric Technology Today*, vol. 2013, no. 5, 2013, pp. 8-11.

[3] G. Hinton, S. Osindero, Y. W. Teh, “A Fast Learning Algorithm for Deep Belief Nets”, *Neural Computation*, vol. 18, no. 7, 2006, pp. 1527-1554.
[4] Y. LeCun, Y. Bengio, G. Hinton, “Deep learning”, *nature*, vol. 521, no. 7553, 2015, pp. 436-444.
[5] Y. Sun, D. Liang, X. G. Wang, X. O. Tang, “DeepID3: Face Recognition with Very Deep Neural Networks”, arXiv: 1502.00873v1, 2015.
[6] J. T. Liu, Y. F. Deng, T. Bai, Z. P. Wei, C. Huang, “Targeting ultimate accuracy: face recognition via deep embedding”, arXiv: 1506.07310, 2015.
[7] N. Fernando, S. W. Loke, W. Rahayu, “Mobile cloud computing: A survey”, *Future Generation Computer Systems*, vol. 29, no. 1, 2013, pp. 84-106.
[8] M. Villari, M. Fazio, S. Dustdar, O. Rana, R. Ranjan, “Osmotic Computing: A new paradigm for edge/cloud integration”, *IEEE Cloud Computing*, vol. 3, no. 6, 2016, pp. 76-83.
[9] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, V. young, “Mobile Edge Computing A key technology towards 5G-First edition”, 2015.
[10] P. Mach, Z. Becvar, “Mobile Edge Computing: A survey on Architecture and computation offloading”, *IEEE Communications Survey & Tutorials*, 2017.
[11] S. G. Deng, L. T. Huang, H. Y. Wu, W. Tan, J. Taheri, A. Zomaya, Z. H. Wu, “Toward Mobile Service Computing: Opportunities and Challenges”, *IEEE Cloud Computing*, vol. 3, no. 4, 2016, pp. 32-41.
[12] W. Abdul, Z. Ali, S. Ghouzali, B. Alfawaz, G. Muhammad, M. S. Hossain, “Biometric security through visual encryption for fog edge computing”, *IEEE Access*, vol. 5, 2017, pp. 5531-5538.
[13] K. M. He, X. Y. Zhang, S. Q. Ren, J. Sun, “Deep residual learning for image recognition”, in *Processdings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770-778.

An Approach to Discovering Event Correlations among Edge Sensor Services

Chen Liu[1,2], Yunmeng Cao[1,2], Yanbo Han[1,2]

[1] Beijing Key Laboratory on Integration and Analysis of Large-Scale Stream Data, North China University of Technology, Beijing, China

[2] Institute of Data Engineering, School of Computer Science and Technology, North China University of Technology, Beijing, China

Abstract: IoT (Internet of Things) allows industry devices to be sensed and controlled remotely. A surge in sensor data volume has exposed the shortcomings of cloud computing, particularly the limitation of network transmission capability and centralized computing resources. Fog computing was proposed to brings back partial computation load from the cloud to the edge sensors. However, related researches on fog computing just start and far from being mature. To our best knowledge, few works attach enough importance on what software abstractions should be fit to put on an edge node and how to program them.

This paper proposes a service-oriented framework, called as INFOG, to support the dynamic cooperation among sensors with the fog computing paradigm. Proactive data services and service hyperlinks, which are our previous work, are two key abstractions for the INFOG framework. The services are software-defined abstraction of physical sensors. They are deployed in edge nodes in INFOG. And service hyperlinks, encapsulation of service correlations, enable the cooperation of sensors at the software layer. To mine service hyperlinks, we propose an effective algorithm, which can transform event correlation mining problem into a frequent sequence mining problem. Based on the dataset from a real power plant, we do experiments to verify the effectiveness of our algorithm.

Keywords: fog computing; sensor data; proactive data service; service hyperlinks.

1 Introduction

Today, with the new wave of the forth industry revolution, the IoT-based integration of physical systems and information systems is gaining momentum [1]. The big real-time IoT data leads to the emergency of a new computing paradigm i.e., fog computing. Compared with cloud computing, the fog computing doesn't require data to be stored and processed in a centralized way. It stresses data and its processing should be put close to the edge equipment, which can greatly lower the computation and storage costs of the cloud server.

Although fog computing has lots of advantages, its related researches just start and far from being mature. Most of current researches still put their focuses on the architecture, wireless sensor networks, hardware and underlying applications [2]. Few works attach enough importance on what software abstractions should be fit to put on a fog/edge node and how to program them. Actually, it is very important for an IoT application built with the fog computing paradigm.

The real problem is the detection of some complicated anomaly heavily depends on the cooperation of different sensors. From the perspective of applications, sensor cooperation will be finally mapped to the cooperation of software abstraction on the edge nodes. Discovery and analyses of sensor cooperation process becomes more difficult as it

usually spans over a set of autonomous and distributed sensors. A complete process is hard to be clearly described since process knowledge is scattered across events derived from multiple sensors as well as their correlations. Hence, the paper aims to mine partial knowledge, i.e. event correlations, to support the cooperation of sensors on edge nodes.

In our previous work, we try to map physical sensors into a software-defined abstraction, called as proactive stream data services [3, 4]. Based on this abstraction, this paper proposes a service-oriented framework, called as INFOG (Dynamic Sensor Cooperation in the Fog Computing Paradigm), to support the dynamic cooperation among sensors with the fog computing paradigm. The key of INFOG framework is the service hyperlink model, which is proposed to be capable of encapsulating correlations among events. To mine service hyperlinks, we propose an effective algorithm, which can transform event correlation mining problem into a frequent sequence mining problem. Furthermore, experiments are done to show efficiency and effectiveness of our algorithm based on a dataset from a real power plant.

2 Preliminaries

Proactive data services and service hyperlinks are two key abstractions for the INFOG framework. Firstly, we rely on our previously proposed proactive data service model to encapsulate physical sensors into flexible and deployable software abstractions [3-4]. To facilitate the cooperation with other sensors/services, our service model is deeply blended with the event model. A data service regards each status change of sensor data as an event, multiple event streams can generate a new event stream with richer semantics through operations such as filtering, aggregation, and correlation.

Definition 1. (Event): an event can be denoted by $e = \langle id, source, type, val, time \rangle$, in which *id* is the unique identifier of *e*; *source* is the service which generates *e* and herein we identify a *source* by the corresponding serivce ID; *type* is the event type of *e*; *val* is the value of *e*; *time* is generation time of *e*.

Based on the Definition 1, an **event sequence** is a finite series of events ordered in time. And an **event stream** is an infinite event sequence.

Definition 2. (Proactive Data Service): A proactive data service is defined as a 7-tuple: $S = <uri, in\_channels, out\_channels, APIs, event\ handler, KPI, hyperLinks>$. *uri* is the unique identifier of *S*; *in_channels* represents input channels receiving all streams arriving at the service; *out_channels* represents output channels, through which output event streams are distributed to the other services; *APIs* is a set of RESTful-like APIs; *event handler* represents the composition of several handling operations for processing received streams and generating output event streams; *KPIs* corresponds to key attributes to be exposed with RESTful-like APIs, which are generated from the attributes of multiple physical sensors; and *hyperLinks* is essentially a routing table, which can point out the target services that an output event stream is sent to.

Service hyperlink is the other important abstraction for the INFOG framework. Service hyperlinks encapsulate event correlations among multiple event streams.

Definition 3. (Service Hyperlink): Given two proactive stream data services S_i and S_j, let ES_i be an event stream of S_i, and ES_j be an event stream of S_j. If ES_j correlated with ES_i, we define a service hyperlink as $SHL(S_i, S_j) = \{\langle id, ES_i \Rightarrow ES_j \rangle\}$.

In INFOG framework, discovered service hyperlinks are used to guide event routing in the runtime. Service hyperlinks are recorded into its corresponding service as the

preset process knowledge fragments. They can help us to find out the next services where the events should be most possibly sent to. When a service generates an event, it first checks the hyperlinks to find if there are routes for this event, if yes it then sends this event to next service according hyperlinks; if no it broadcasts this event to other services.

4 Service Hyperlink Discovery

4.1 The Rationale of Service Hyperlink Discovery

Based on the above description, the inputs of service hyperlink algorithm are numerical event streams from various sources. k event streams can generate $k*k$ scanning times, which will bring heavy latency in a streaming algorithm. The input event streams have a high speed of one event per second. It is hard to analyse a large volume of numerical event streams with such high speed instantly [5]. To facilitate the analysis, many researches often pre-process a numerical event stream by transforming it into a simplest form like a symbol event stream [5, 6].

Before discussing the rationale of our algorithm, we introduce the pre-processing of numerical event streams, i.e. symbolization. Let $E = \langle e_1, e_2, \ldots, e_n \rangle$ be a numerical event sequence. **Symbolic Aggregate approXimation (SAX)** [5] is a classic symbolic representation algorithm, which allows an event sequence of length n to be reduced to a string of length m ($m \ll n$) composed of k different symbols ($k > 2$). Besides reducing the speed of event streams, another advantage of symbolization is that we can measure the correlation of event sequences in symbolization space [5-7]. A symbolized event sequence is an abstract of the original one. If the values of an event sequence correlate with another one, there will exist frequent sequence between them. Obviously, longer closed frequent sequence indicates higher correlation of them.

Based on the analysis above as well as definition 2, we can measure the event correlation between two numerical event streams via counting the event sequence pairs which have a long enough closed frequent sequence with above temporal constraint.

We propose an algorithm with three steps to instantly discover service hyperlinks. Firstly, it symbolizes the received numerical event streams. Secondly, our algorithm mines frequent sequences with temporal constraint in the current sliding window. Thirdly, it counts the correlated event sequence pairs to discover service hyperlinks.

4.2 The Service Hyperlink Discovery Algorithm

Generally, minimum support threshold in this step is set to be 2 as an event correlation should occur between at least two event streams.

Before discussing the algorithm, we introduce the concept of **event cluster**, which is a set of events with same symbol (including the given one) which arrives before a given event in a short time Δt. In a sliding window, for each symbolized event sequence, we compute the event cluster for each event in it. If of the cluster size is no less than minimum support threshold, the symbol in the cluster is a frequent sequence with length 1. If k event clusters for a given event sequence corresponding to same sequence id set, there is a frequent sequence with length k. Note that, if an event can only be involved by one event cluster. Herein, an event is included by the newest event cluster preferentially.

Our algorithm tries to instantly discover service hyperlinks. To enhance the instantaneity, only the event sequence with a new arriving element in current sliding window needs to compute event clusters for its each event. Besides, only the event sequences related to the events in the event cluster of the new arriving event may generate new frequent sequences.

From now on, our algorithm finds all closed frequent sequences in current window. It can identify correlated event sequences. To discover a service hyperlink, the algorithm will count the correlated event sequence pairs. Assume that a new closed frequent sequence among a set of event streams in current sliding window is discovered. If there is no closed frequent sequence among same event streams in the last window, the counter is increased by one. Otherwise, it remains the same. If a newly updated counter reaches the threshold *w*, a valid service hyperlink is discovered. To handle event streams, we borrow ideas from Lossy Counting algorithm [9] to make approximate counting in our algorithm. By the limitation of space, we omit the details of Lossy Counting algorithm, which is a classic streaming algorithm guaranteeing the accuracy by user-defined error.

5 Experiments

Datasets: The following experiments use real plant power dataset in Tianjin, China. The sets contain sensor data from 2016-03-01 00:00:00 to 2016-03-30 23:59:59. Totally 440 sensors are involved and each sensor generates one data record per second. Our services generate descending event streams and ascending event streams for each sensor.

Values of parameter Δt and *w* will significantly impact the results of our algorithm. According to our previous works, Δt can be set to be 5 minutes at most. Thus, in this part, we verify the variation of discovered service hyperlinks number under different *w* values.

We run our algorithm for w = 2, 4, …, 20 under Δt = 5 minutes, and record the number of discovered service hyperlinks on the real event set. Figure 1 shows the final results. The figure shows that total number of service hyperlinks falls exponentially with the increase of *w* during w =2, 4, …, 10. When *w* reaches 10, the number falls much more slowly than before. As the figure shows, it falls linearly with the growth of *w*.

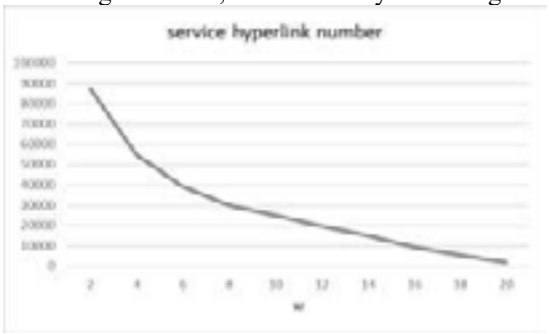


Fig. 1. Variation of Service Hyperlink Number under Different Values of w

6 Related Works

Reguieg regarded event correlation as correlation condition mentioned above [10]. It presented a framework and techniques with multi-pass algorithms to discover correlation

conditions in process discovery and analysis tasks over big event datasets using MapReduce. Motahari-Nezhad focused on event correlations in service-based processes [11]. It proposed the notion of correlation condition as a predicate over the attributes of events that can verify which sets of events belong to the same instance of a process. Liu presented an event correlation service for distributed middleware-based applications [12]. It enables complex event properties and dependencies to be explicitly expressed in correlation rules.

Recently, some researchers focus on event dependencies. Song mined activity dependencies to discover process instances when event logs cannot meet the completeness criteria [13]. In this paper, the control dependency indicates the execution order and the data dependency indicates the input/output dependency in service dependency. A dependency graph is utilized to mine process instances. In fact, the authors do not consider the dependency among events. Plantevit presented a new approach to mine temporal dependencies between streams of interval-based events. [14]. Two events have a temporal dependency if the intervals of one are repeatedly followed by the appearance of the intervals of the other one, in a certain time delay.

Funding

This work was supported by National Natural Science Foundation of China (No. 61672042).

References

[1] S Zhao, B Cheng, L Yu, S Hou, Y Zhang, J Chen, “Internet of Things Service Provisioning Platform for Cross-Application Cooperation,” International Journal of Web Services Research, 2016, 13(1), pp: 1-22.
[2] S Yi, C Li, Q Li, A survey of fog computing:concepts, applications and issues,” Proceedings of the 2nd Workshop for Mobile Sensing, Computing and Communication, 2015, pp: 37-42.
[3] Y Han, G Wang, J Yu, C Liu, Z Zhang, M Zhu, “A Service-Based Approach to Traffic Sensor Data Integration and Analysis to Support Community-Wide Green Commute in China,” IEEE Transactions on Intelligent Transportation Systems, 2016, 17(9), pp: 2648-2657.
[4] Y Han, C Liu, S Su, M Zhu, Z Zhang, “A proactive service model facilitating stream data fusion and correlation,” International Journal of Web Services Research, 2017, 14(3), pp: 1-16.
[5] J Lin, E Keogh, S Lonardi, B Chiu, “A symbolic representation of time series, with implications for streaming algorithms,” Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, 2003, pp: 2-11.
[6] M Van Hoan, M Exbrayat, “Time series symbolization and search for frequent patterns,” Proceedings of the 4th Symposium on Information and Communication Technology, 2013, pp: 108-117.
[7] G Smith, J Goulding, “A novel symbolization technique for time-series outlier detection,” Proceedings of 2015 IEEE International Conference on Big Data, 2015, pp: 2428-2436.
[8] J Pei, J Han, W Wang, “Constraint-based sequential pattern mining: the pattern-growth methods,” Journal of Intelligent Information Systems, 2007, 28(2), pp: 133-160.
[9] G S Manku, R Motwani, “Approximate frequency counts over data streams,” Proceedings of the VLDB Endowment, 2002, 5(12), pp: 1699-1699.
[10] H Reguieg, B Benatallah, H R M Nezhad, F Toumani, “Event correlation analytics: scaling process mining using mapreduce-aware event correlation discovery techniques,” IEEE Transactions on Services Computing, 2015, 8(6), pp: 847-860.
[11] H R Motahari-Nezhad, R Saint-Paul, F Casati, B Benatallah, “Event correlation for process discovery from web service interaction logs,” VLDB Journal, 2011, 20(3), pp: 417-444.
[12] Y Liu, I Gorton, V K Lee, “The architecture of an event correlation service for adaptive middleware-based applications,” Journal of Systems and Software, 2008, 81(12), pp: 2134-2145.
[13] W Song, H A Jacobsen, C Ye, X Ma, “Process Discovery from Dependence-Complete Event Logs,” IEEE Transactions on Services Computing, 2016, 9(5), pp: 714-727.
[14] M Plantevit, C Robardet, V M. Scuturici, “Graph dependency construction based on interval-event dependencies detection in data streams,” Intelligent Data Analysis, 2016, 20(2), pp: 223-256.

Fog-Cloud task scheduling of Energy consumption Optimization with deadline consideration

Abstract: The emerging IoT introduces many new challenges that cannot be adequately addressed by the current "cloud-only" architectures. The cooperation of the fog and cloud is considered to be a promising architecture, which efficiently handles IoT's data processing and communications requirements. However, how to schedule tasks to better adapt to IoT real-time needs and reduce the energy in the fog-cloud system is not well addressed. In this paper, we formulate a task scheduling problem into a constrained optimization problem in Fog-Cloud Computing System. Then, an efficient deadline-energy scheduling algorithm based on ant colony optimization(DEACO) is proposed, which achieves to reduce energy consumption on the condition of satisfying tasks deadline. Finally, the experimental result shows that our scheduling approach reduces energy more effective.

Keywords: IoT; Cloud Computing; Fog Computing; Energy consumption; Task scheduling; optimal ant colony algorithm.

1 Introduction

Fog computing is proposed by Cisco in 2012 (Varshney and Simmhan, 2017), which provide computing, storage and networking services between traditional cloud computing data centers and smart terminal devices. In the viewpoint of Cisco, fog computing as the complement of the cloud computing extends the cloud computing to the edge of the network, deploying fog devices closer to where data is generated, analyzing data and processing user requests directly at the edges rather than enforcing all requests and data are transmitted to the cloud server. Fog-Cloud architecture can achieve real-time application handled on fog nodes and sophisticated application delivered to the cloud nodes.

In current IoT environment, one of the typical distributed computing, the management of the task and resource is a big challenge especially in consideration of energy consumption with business application QoS demands. In order to better meet user needs and ensure that tasks are handled smoothly, an effective task scheduling strategy is very important, which greatly affects the energy consumption of the resources. If the task can not be properly scheduled will increase energy consumption, so the energy-aware scheduling strategy can save a large amount of energy.

Xiuli et al. (2016) introduced the cloud and fog network architecture into the field of car networking to solve the problem of high latency and not to support for mobility and location awareness in today's car networking. An improved particle swarm optimization algorithm is proposed to reduce delay and improve the quality of service QoS. Hoang and Dang (2017) proposed an architecture based on fog regions and clouds, and designed an efficient t ask s cheduling m echanism f or h euristic s cheduling a lgorithm t o m inimize the task completion time and improve user experience. However, they don't take into account the needs of latency-sensitive applications that require task processing to complete within a certain delay. Sharma and Verma (2017) proposed a new energy aware task scheduling algorithm to reduce energy consumption in a cloud environment. Kaur et al. (2017) proposed

approach that can achieve to minimize the overall energy consumption of the cloud system without missing their deadlines for scheduling real time tasks. However, they only focus on cloud environment.

In summary, it is now an emerging and critical issue to design a scheduling algorithm to support the satisfaction of QoS requirement and reduce the energy consumption in a hybrid fog-cloud environment. With this initial motivation, this paper addresses the task scheduling issue in fog-cloud system, to meet the requirement of the task deadline and optimize energy consumption. The experimental results show the outstanding performance of our method compared with some other works.

The remainder of the paper is organized as follows. In section 2, we firstly make the model of tasks, cloud nodes and fog nodes. Next, we formulate the task scheduling problem. Finally, we present our proposed algorithm. Then we describe some experimental results in section 3. Section 4 presents our conclusions.

2 Task Scheduling in Fog-Cloud Environment

In this paper, we focus on task scheduling problem in fog - cloud system, which aims to assign a set of tasks to every fog nodes and every cloud nodes optimally. More specifically, the goal is to achieve the task scheduling that minimizes energy consumption under the condition of meeting the task delay constraint, according to the length of the task, delay constraints and the processing capacity of the cloud-fog nodes and the network bandwidth.

2.1 Tasks, Fog and Cloud Model

The request information is submitted by users to the fog devices, then, which is converted into a series of tasks in the fog-cloud broker. Task sets $Tasks$ =$\{task_1, task_2, task_3, \ldots, task_i, \ldots, task_l\}$. in which $task_i$ $(1 \leq i \leq l)$ denotes the i^{th} task in the set. For $\forall\ task_i \in Tasks$ has four characteristics: task length, input data size, output data size, deadline. They are denoted as T_{le}, T_{is}, T_{os}, T_{de}, respectively.

In the fog and cloud system, there are heterogeneous fog and cloud nodes that virtualize these physical resources with computing and storage as a series of resource collections. The cloud resource set has j fog nodes and k cloud nodes. The attributes of the fog nodes and cloud nodes are defined as follows.

$Fogs$=$\{fog_1, fog_2, fog_3, fog_4, \ldots, fog_j, \ldots, fog_m\}$. FN_{mips} and FN_{bw} represent the computing power and network bandwidth of fog node fog_j, respectively.

$Clouds$=$\{cloud_1, cloud_2, cloud_3, \ldots, cloud_k, \ldots, cloud_n\}$. CN_{mips} and CN_{bw} represent the computing power and network bandwidth of cloud node $cloud_k$, respectively.

2.2 Optimization Function

According to the scheduling strategy, if the task is allocated to the cloud nodes, energy consumption includes transmission energy consumption and computing energy consumption in the cloud. If the task is assigned to the fog node, energy consumption is only the computing energy consumption in the fog node. Our ultimate goal is to minimize the energy consumption under tasks deadline constraint. As a result, objective function and the constraint condition are as follows.

$$\min \quad F = \sum_i \sum_j E_{ij}^{fog} + \sum_i \sum_k E_{ik}^{cloud} \tag{1}$$

$$\text{s.t.} \quad \begin{cases} (Tran_{ik} + Com_{ik}) \leq T_{de}^{i} & (2) \\ (Tran_{ij} + Com_{ij}) \leq T_{de}^{i} & (3) \\ E_{ik}^{cloud} = P_{vmh} * Com_{kh}^{i} + r\left\{P_{idle}^{router} / (U * C_{max}) + E_b\right\} * N_{bit} & (4) \\ E_{ij}^{fog} = P_{idle}^{fog}(a+1) Com_{ij} + \int_{Com_{ij}} \left\{P(t) - P_{idle}^{fog}\right\} dt & (5) \end{cases}$$

where $Com_{ij} = T_{le}^{i}/FN_{mips}^{j}$, $Com_{ik} = T_{le}^{i}/FN_{mips}^{k}$, $Tran_{ij} = \left(T_{is}^{i} + T_{os}^{i}\right)/FN_{bw}^{j}$ and $Tran_{ik} = \left(T_{is}^{i} + T_{os}^{i}\right)/Ru_{bw} + \left(T_{is}^{i} + T_{os}^{i}/CN_{bw}^{k}\right)$ denote the computing time of $task_i$ on fog node fog_j, the computing time of $task_i$ on cloud node $cloud_k$, the transmission time of $task_i$ on fog node fog_j, the transmission time of $task_i$ on cloud node $cloud_k$, respectively. P_{vmh} represents the virtual machine's power consumption, P_{max} denotes the router's maximum power consumption, P_{idle}^{router} represents the router's free power consumption, C_{max} denotes the maximum load on the router and U represents the router's utilization, Let r denote the average number of routers in the path of a task. $E_b = (P_{max} - P_{idle}) / (C_{max}U)$, N_{bit} denotes the number of bits of the task $task_i$ transmitting passed to the router. $a = t_{idle}/t_{act}$ denotes the ratio of free time to active time of the fog nodes. P_{idle}^{fog} represents the free power of the fog nodes. $P(t)$ denotes the power of the fog node at the moment t of executing the task.

2.3 ACO-based Algorithm (DEACO)

The ACO is a bionic class inspired optimization algorithm(Dorigo et al., 2005). In this paper, the ant colony algorithm is used to solve the optimization problem with constraints. The scheduling problem in fog-cloud system combined with an ant colony optimization algorithm is represented as a Bipartite Graph $Z = (T; V, E)$, where T and V are a set of nodes which denote tasks T_i $(i = 1, 2, \ldots, l)$ and vms V_j $(j = 1, 2, \ldots, m + n)$ in the fog nodes and cloud nodes respectively, and E is the set of paths for a task of T_i to a Vm of V_j, where $E = \{e_{ij} | (i = 1, 2, \ldots, l; j = 1, 2, \ldots, m + n)\}$, an ant from a set of ants k $(0 \leq k \leq K)$ selects path e_{ij} to express tasks T_i assigned to vms in the fog nodes and cloud nodes. Ants completing a trip means that all tasks are assigned to virtual nodes in the fog nodes and cloud nodes for processing. The DEACO algorithm is presented in algorithm 1.

3 Experiment and Analysis

In this experiment, in order to evaluate the performance of the proposed scheduling mechanism, we compare our algorithm with two others: Greedy for Energy(GfE), and DACO(standard ant colony algorithm optimization time). We use Cloudsim and extend the modular of task scheduler with fog for modeling and simulation. Our experiment cover a set of random tasks with the increase of sizes from 20 to 100 and a set of heterogeneous vms that are from one cloud node and 6 fog nodes. The I/O data of a task have a size from 5 to 6 MB. The length of the task is from 9000 MI to 20000 MI. The value of the parameters of the tasks is length ∈ [9000,15000]MI, deadline ∈ [10,20]s. The value of the parameters of the cloud nodes is processing rate ∈ [1000,1500]MIPS, bandwith ∈ [1.7,2]Mbps, respectively. The value of the parameters of the fog nodes is processing rate ∈ [700,900]MIPS, bandwith ∈ [90,100]Mbps, Idle=10w, Active ∈ [18,25]w, a=5,respectively. The value of the parameters of the routers is Idle=11070w, Max=12300w, Maxload=4480w,

Algorithm 1: DEACO

Input: List of Tasks T_i and List of VM V_j, K, N_{max}
Output: $solution = \{e_{ij} | (i = 1, 2, \ldots, l; j = 1, 2, \ldots, m + n)\}$, minEnergy

for $Iter \leftarrow 1$ **to** N_{max} **do**
  for $k \leftarrow 1$ **to** K **do**
    Initialize the heuristic information η_{ij}, the pheromone concentration τ_{ij}, each ant select a path e_{ij} randomly ;
    while $i \leq l$ **do**
      for $j \leftarrow 1$ **to** n **do**
        $D_{ij} \leftarrow Tran_{ij} + Com_{ij}$;
        $DE_{ij} \leftarrow (D_{ij}/D_{max}) * (E_{ij}/E_{max})$;
        $\eta_{ij} \leftarrow 1/DE_{ij}$;
        $p_{ij}(t) \leftarrow [\tau_{ij}(t)]^{\alpha} (\eta_{ij})^{\beta} / \sum_{k \in allowed_k} [\tau_{ij}(t)]^{\alpha} (\eta_{ij})^{\beta}$;
        Compute E_{ij} (refers to Eq. (4), (5)), D_{ij}, DE_{ij}, η_{ij}, p_{ij} ;
        Choose the Vm_j for T_i ;
        if $D_{ij} \leq T^i_{de}$ **then**
          $Table_k \leftarrow \{e_{ij}\}$;
    Update local pheromone $\tau_{ij}(t+1) \leftarrow (1-\rho) \times \tau_{ij}(t) + \sum_{k=1}^{m} Q/E^k_{ij}(t)$;
  Calculate best energy F_{best} (refers to Eq. (1));
  Record the best path e_{ij} ;
  Update global pheromone $\tau_{ij}(t+1) \leftarrow (1-\rho) \times \tau_{ij}(t) + Q/F_{best}$;
$minEnergy \leftarrow F_{best}$;
return $solution \leftarrow$ best path e_{ij};

utilization=0.6, respectively. The value of the parameters of the DEACO algorithm is α =1, β=1, Q=100, m=10, t_{max}=100, ρ=0.2, respectively.

In the Fog-Cloud architecture, to show the results of our algorithm and comparison experiments in energy comparison, we define a calculation model of energy consumption measurement(EMC), which is the ratio between the sum energy consumption of the tasks processed based on the scheduling algorithm and the sum energy consumption of the tasks processed on the the worst node. The EMC equal to 1 represents the situation that the scheduler makes the highest energy consumption value, the smaller the value of the EMC, the better the scheduling strategy optimizes energy consumption.

In the experiment, we evaluate the EMC for the GfE, DACO and the DEACO, given 20 to 100 tasks. Figure 1 shows the result, in which the DEACO has the best performance, followed by GfE, while the performance of DACO was the worst. It was observed that the proposed algorithm achieved an improvement of about 12% when compared to the DACO strategy, was approximately 8% better than GfE algorithm in average.

4 Conclusion

In this paper, this paper address task scheduling in hybrid cloud-fog computing. Firstly, we present a modeling of the energy of the fog and cloud. Then, we propose the algorithm to optimize energy consumption with considerations of task deadline. Finally, simulations

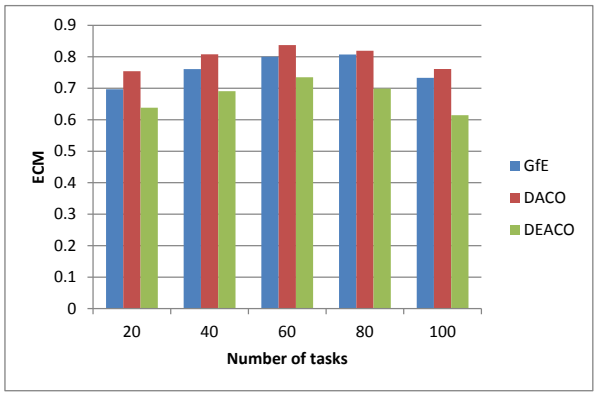


Figure 1: Energy comparison of DEACO vs. DACO and GfE

and numerical results have shown that our work can show a better performance than other existing methods.

In future work, the effect of load balancing will be considered. Also the comparison between our approach and other meta-heuristics approaches will be performed.

References

Dorigo, Marco, Blum, Christian, 2005. Ant colony optimization theory: a survey. Theoretical Computer Science 344 (2-3), 243–278.

Hoang, D., Dang, T. D., Aug 2017. Fbrc: Optimization of task scheduling in fog-based region and cloud. In: 2017 IEEE Trustcom/BigDataSE/ICESS. pp. 1109–1114.

Kaur, S., Ghose, M., Sahu, A., Dec 2017. Energy efficient scheduling of real-time tasks in cloud environment. In: the 15 international Conference on Data Science and Systems (HPCC/SmartCity/DSS). pp. 178–185.

Sharma, M., Verma, A., Feb 2017. Energy-aware discrete symbiotic organism search optimization algorithm for task scheduling in a cloud environment. In: 4th International Conference on Signal Processing and Integrated Networks (SPIN). pp. 513–518.

Varshney, P., Simmhan, Y., May 2017. Demystifying fog computing: Characterizing architectures, applications and abstractions. In: 2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC). pp. 115–124.

Xiuli, Zhiyuan, Chenhua, Jian, Fang, 2016. A novel load balancing strategy of software-defined cloud/fog networking in the internet of vehicles. China Communications 13 (S2), 140–149.

Execution Cost and Fairness Optimization for Multi-Server Mobile-Edge Computing Systems with Energy Harvesting Devices

Hailiang Zhao, Wei Du, Wei Liu, Tao Lei, Qiwang Lei

School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China

Abstract: Mobile edge computing systems with energy harvesting devices pave the way for quality of user experience (QoE) improvement by computation offloading and green computing by utilizing renewable energy. Nevertheless, because battery energy is obtained free and time-varying, computation offloading strategies should have different design objective and is temporally correlated. Moreover, considering multi-user and multi-server systems where mobile devices can move arbitrarily, resource competition and MEC server selection should be incorporated in the strategies. In this paper, we will develop two effective computation offloading strategies. The execution cost (i.e., execution latency and penalty for dropped tasks) and the fairness (in terms of ratio of computation tasks offloaded) are adopted as the performance metrics. Two online algorithms, namely, the LODCO-Based Greedy Algorithm and LODCO-Based ϵ-Greedy Algorithm, are proposed. Both of them are based on Lyapunov Optimization technologies and the LODCO Algorithm[1]. By choosing the execution mode among local execution, offloading execution and task dropping for each mobile device, our algorithms can asymptotically obtain the optimal results for the whole system. Two algorithms proposed are low-complexity and could be operated without too much priori knowledge. Moreover, the algorithms will inherit every advantage from LODCO Algorithm and adapt perfectly to the more complex environment. Simulation results illustrate that compared with the LODCO Algorithm, our algorithms could improve the ratio of computation tasks offloaded by 5% and 10%, respectively.

Keywords: Mobile edge computing, Energy harvesting, Computation offloading, Greedy strategy, Lyapunov Optimization.

1 Introduction

Mobile edge computing (MEC) is a new paradigm which provides IT environment and cloud computing capability within radio access networks [1]. By offloading computation intensive tasks to MEC servers, users could experience low latency and ultra-high bandwidth services in a MEC system [2]. To overcome the limitations of battery-powered mobile devices, energy harvesting (EH) is introduced to MEC systems [3], where mobile devices could be charged by renewable energy sources such as solar radiation, wind and human motion energy [4]. Recently, efficient computation offloading strategies have been

[1] LODCO is the abbreviation of the Lyapunov optimization-based dynamic computation offloading, which is an algorithm designed in Reference [3].

developed for single-user and single-server MEC systems with EH devices [3]. Unfortunately, these strategies are not suitable for multi-user and multi-server systems, which are more typical scenarios in a real world [5] because of failing to address resource competition and server selection.

To exploit in full the benefits of computation offloading in the considered multi-user and multi-server MEC systems with EH devices, there are several key challenges that need to be addressed. Firstly, the computational resource and the radio resource are shared by multiple mobile devices. As a result, interference and competition could not be ignored. How to allocate limited resources fairly among users should be investigated. Secondly, a user probably could connect to more than one MEC server. Then, the user could choose one server to offload its task. So, how to choose a proper server according to the system optimization metrics or the user's preference has to be investigated.

In this paper, we design two online computation offloading strategies for multi-user and multi-server MEC systems with EH devices. Our major contributions are summarized as follows:

- We consider a general MEC system with multiple EH mobile devices and multiple resource-constrained MEC servers where every mobile device can move arbitrarily within certain areas.
- User mobility and its effect on resources contention and server selection are modelled as one part of a non-convex optimization problem.
- The execution cost (i.e. execution latency and penalty for dropped tasks) and fairness (in terms of the ratio of offloading computation tasks) are optimized simultaneously.
- After utilizing Lyapunov Optimization, the original problem can be converted to a deterministic optimization problem at each time slot, which is a cornerstone of two algorithms proposed in this paper.
- Both two proposed algorithms can handle the correlation between any two mobile devices when choosing the computation modes, especially the offloading computation mode, which could not be solved by the LODCO Algorithm.
- In addition, both two proposed algorithms can obtain the optimal results after several iterations. By comparing with the LODCO Algorithm, the LODCO-Based Greedy Algorithm and the LODCO-Based ϵ-Greedy Algorithm not only keeps perfectly the advantages of LODCO Algorithm but also promotes the ratio of computation tasks offloaded notably.

The organization of this paper is as follows. We survey state of the art in Section 2. In Section 3, the system model is introduced. In Section 4, the LODCO-Based Greedy Algorithm and the LODCO-Based ϵ-Greedy Algorithm are proposed based on the formulated problem. Simulation results and Conclusions of this paper are demonstrated in Section 5 and Section 6, respectively.

2 Related works

Computation offloading for multi-user multi-server mobile systems is a very challenging problem because of complexity of the scenario and interdependence among users and servers. A few strategies have proposed in recent years. In [6], the power-delay trade-off in the context of task offloading was studied. The problem was formulated as a computation and transmits power minimization subject to latency and reliability constraints. In [7], the

power minimization for the mobile devices by data offloading was investigated. Centralized and distributed algorithms for joint power allocation and channel assignment together with decision making were proposed. In [8], the problem of joint task offloading and resource allocation was formulated as a mix integer non-linear program. The task offloading decision, uplink transmission power of mobile users and computational resource allocation at the MEC servers were jointly optimized. The users' task offloading gains, which are measured by the reduction in task completion time and energy consumption, were maximized. Obviously, energy consumption is always optimized in all the works mentioned above. However, the optimization objective of MEC systems with EH devices is shifted from minimizing the battery energy consumption as the harvested energy is ample and free. As a result, those computation offloading strategies dedicated to the energy conservation cannot be utilized without modification.

In [9], a device-edge-cloud MEC system was investigated. A network aware multi-user multi-edge computation partitioning problem was formulated. Computation and radio resources were allocated such that the average throughput of the users was maximized. The system considered in [9] is similar to this paper. Nevertheless, a partial offloading model was utilized in [9] while a binary offloading is exploited in this paper.

The work most similar to this paper is [3]. In [3], a green MEC system with EH devices was analysed, and an effective computation offloading strategy was developed. Moreover, the execution cost, which includes both the execution latency and task failure, was adopted as the performance metric. Finally, a low-complexity online algorithm was proposed. The key differences between our work and [3] are summarized as follows. Firstly, while the system discussed in [3] focused on a single user single MEC server, we dedicate on multi-user multi-server scenario. Secondly, user mobility, which was ignored in [3], is considered in this paper. Thirdly, the MEC server has limited computation capability, which generalizes the work in [3], where the MEC server was assumed to be with unlimited computational resources. Fourthly, compared to the single-objective optimization problem formulated in [3], we are interested in optimizing two objectives simultaneously. Due to the differences mentioned above, our work is more complicated and difficult than that in [3].

3 System model

3.1 System description

We consider a MEC system consisting of N mobile devices equipped with EH component and M MEC servers, where each mobile device and each MEC server share the same property, respectively. We assume that time is slotted, and denote the time slot length and the time slot index set by τ and $\mathcal{T} \triangleq \{0,1, \dots\}$, separately. We use $\mathcal{N} \triangleq \{1,2, \dots, N\}$ to denote the set of mobile devices and $\mathcal{M} \triangleq \{1,2, \dots, M\}$ to denote the set of MEC servers.

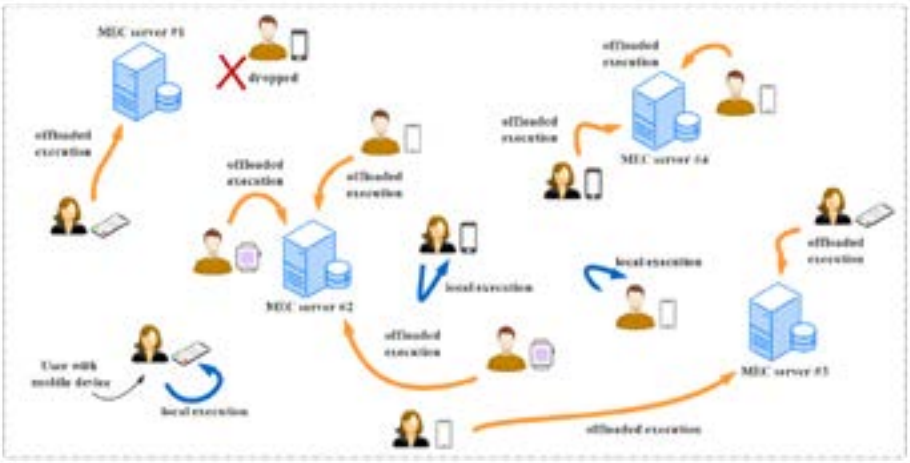


Fig. 1. A system with multiple mobile device and multiple MEC server.

As shown in Fig. 1, all mobile devices and MEC servers are limited to a specific area, where each MEC server is located at a particular position without moving and each mobile device can move around arbitrarily. Each mobile device's location is assumed to be independent and identically distributed (i.i.d.), i.e., ith mobile device's location remains static within each time slot but varies among different time slot. Denote the location of ith mobile device at tth time slot as (x_i^t, y_i^t), and $x_i^t \sim U(0, \mathcal{W})$, $y_i^t \sim U(0, \mathcal{L})$, $t \in \mathcal{T}$, $i \in \mathcal{N}$, where $\mathcal{W}$ and $\mathcal{L}$ denote the width and length of the specific space, respectively.

Denote the distance between ith mobile device and jth MEC server at tth time slot as $d_{i,j}^t$, where $d_{i,j}^t \triangleq \sqrt[2]{(x_i^t - x_j^t)^2 + (y_i^t - y_j^t)^2}$, i.e., the distance matrix at tth time slot $D^t \triangleq (d_{i,j}^t)_{N \times M}$ is determined, while time-varying in different time slots.

3.2 Computation tasks model

We focus on delay-sensitive computation tasks with the execution deadline no greater than the length of time slot [6]. Denote the computation task generated by ith mobile device at tth time slot as CT_i^t, who has the fixed size L (in bits). Meanwhile, we assume the computation tasks are modeled as an i.i.d. Bernoulli process [3], i.e., at the beginning of each time slot, for each mobile device one computation task CT_i^t is requested with probability ρ where $0 < \rho < 1$. Denote $\zeta_i^t = 1$ if ith mobile device get computation task request at tth time slot.

In our system, there exists no cache queue either in mobile devices or MEC servers. As a result, computation task CT_i^t can either be executed locally at ith mobile device, or be offloaded to jth MEC server and then be executed, where j is the chosen MEC server for ith mobile device by system operation. Besides, if the energy is insufficient of ith mobile device or $\zeta_i^t = 0$, the computation task at tth time slot will be dropped. Denote $I_{i,c}^t \in \{0,1\}$ with $c = \{l, r, d\}$ as the computation mode indicators [3] for ith mobile device at tth time slot, where $I_{i,l}^t = 1$, $I_{i,d}^t = 1$ and $I_{i,r}^t = 1$ indicate that the computation task is executed locally, executed remotely by MEC server and dropped, independently. Because there

exists only 3 modes for ith mobile device to choose at tth time slot, hence those indicators should follow the equation below:

$$I_{i,l}^t + I_{i,r}^t + I_{i,d}^t = 1, t \in \mathcal{T}, i \in \mathcal{N}. \quad (1)$$

3.3 Offloaded computation model

Denote $c_{i,j}^t = 1$ as the indicator of ith mobile device choosing jth MEC server to offload, where $c_{i,j}^t \in \{0,1\}$. Thus, the connection matrix $C^t \triangleq (c_{i,j}^t)_{N\times M}$ is time-varying, which is the same as the distance matrix. We each computation task is assigned to only one server when choosing the offloading computation mode [6], i.e.,

$$\sum_{j=1}^{M} c_{i,j}^t = 1, t \in \mathcal{T}, i \in \mathcal{N}, j\epsilon\mathcal{M}. \quad (2)$$

Denote $\gamma_{i,j}^t$ as the small-scale fading channel power gains, which are assumed to be exponentially distributed with unit mean. Besides, each mobile device shares the same $\gamma_{i,j}^t$ at tth time slot. According to communication theory, the channel power gain from ith mobile device to jth MEC server can be expressed by $h_{i,j}^t = \gamma_{i,j}^t g_0 (\, d_0/d_{i,j}^t)^\theta$, where d_0 denotes the reference distance, θ denotes the pass-loss exponent and g_0 denotes the path-loss constant. As a result, we can obtain the achievable rate $\Gamma(h_{i,j}^t, p_i^t)$ by $\Gamma(h_{i,j}^t, p_i^t) = \omega \log_2(1 + h_{i,j}^t p_i^t/\sigma)$ according to *Shannon Theorem*, where ω represents the system bandwidth, σ is the noise power at each MEC server and p_i^t is the transmit power who cannot exceed the maximum value $p^{\max}$. We assume that every connected mobile device of jth MEC server shares the same bandwidth and each MEC server shares the same noise power.

Denote X_{mobile} and X_{server} as the numbers of CPU cycles required to process one bit task of mobile device and MEC server, respectively. We assume that the computational abilities of MEC servers are constrained, i.e.,

$$\sum_{i\in\mathcal{N}} \mathrm{I}(\, I_{i,r}^t) \cdot c_{i,j}^t L X_{\text{server}} \le f_{\text{server}}^{\max} \tau, t \in \mathcal{T}, j\epsilon\mathcal{M}, \quad (3)$$

where $\mathrm{I}(\cdot)$ is the indicator function and $f_{\text{server}}^{\max}$ denotes the upper bound of each MEC server's CPU-cycle frequency.

We take no consideration of execution latency consumed by the MEC server's execution process [3], i.e., for ith mobile device, the total execution latency of this mode equals the transmission delay for the input task. Thus,

$$D_{i,\text{remote}}^t = \frac{L}{\Gamma(h_{i,j}^t, p_i^t)}, t \in \mathcal{T}, i \in \mathcal{N}, j\epsilon\mathcal{M}. \quad (4)$$

Similarly, we take no consideration of MEC server's energy consumption. Thus, the energy consumed by ith mobile device can obtained by

$$E_{i,\text{remote}}^t = p_i^t \frac{L}{\Gamma(h_{i,j}^t, p_i^t)}, t \in \mathcal{T}, i \in \mathcal{N}, j\epsilon\mathcal{M}. \quad (5)$$

3.4 Local computation model

In order to execute L bits computation task successfully, LX_{mobile} CPU cycles are required. By applying the dynamic voltage and frequency scaling technologies (DVFS) [10], mobile device can control the energy consumed and the execution latency. Thus, the total execution latency of this mode can be obtained by

$$D_{i,\text{local}}^t = \sum_{v=1}^{LX_{\text{mobile}}} (f_{i,v}^t)^{-1}, t \in \mathcal{T}, i \in \mathcal{N}. \quad (6)$$

thus, the energy consumed by ith mobile device can obtained by [3]

$$E_{i,\text{local}}^t = \sum_{v=1}^{LX_{\text{mobile}}} \mathcal{s}(f_{i,v}^t)^2, t \in \mathcal{T}, i \in \mathcal{N}, \quad (7)$$

where $\mathcal{s}$ is the effective capacitance coefficient.

Moreover, we denote the upper bound of CPU-cycle frequency for each mobile device as $f_{\text{mobile}}^{\max}$, i.e., $\forall v \in \{1,2,\dots,LX_{\text{mobile}}\}$, $f_{i,v}^t \le f_{\text{mobile}}^{\max}, t \in \mathcal{T}, i \in \mathcal{N}$.

3.5 Energy harvesting model

In order to embody the stochastic and intermitted nature of the renewable energy process [11], we assume that the harvestable energy E_H^t for each mobile device at the beginning of tth time slot is uniformly distributed with the maximum value of $E_H^{\max}$.

The system need to decide the size of energy who will be stored in the battery of ith mobile device [3]. Denote this part of energy as e_i^t, then we have:

$$0 \le e_i^t \le E_H^t, t \in \mathcal{T}, i \in \mathcal{N}. \tag{8}$$

We assume that other kinds of energy consumption besides local-execution and remote-execution is sufficient small. Denote the energy consumed by ith mobile device as $\mathcal{E}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t)$, where $\boldsymbol{I}_i^t \triangleq [I_{i,l}^t, I_{i,r}^t, I_{i,d}^t]$, $\boldsymbol{f}_i^t \triangleq [f_{i,1}^t, \dots, f_{i,LX_{\text{mobile}}}^t]$. Then We can obtain $\mathcal{E}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t)$ by the following equation:

$$\mathcal{E}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t) = I_{i,l}^t E_{i,\text{local}}^t + I_{i,r}^t E_{i,\text{remote}}^t, t \in \mathcal{T}, i \in \mathcal{N}. \tag{9}$$

Denote the battery level of ith mobile device at tth time slot as b_i^t. Obviously, the energy consumption at each time slot cannot surpass the battery level, i.e.,

$$\mathcal{E}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t) \le b_i^t, t \in \mathcal{T}, i \in \mathcal{N}. \tag{10}$$

Besides, b_i^t evolves according to the following equation:

$$b_i^{t+1} = b_i^t - \mathcal{E}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t) + e_i^t, t \in \mathcal{T}, i \in \mathcal{N}. \tag{11}$$

3.6 QoE-Cost function

Users' QoE [12] consists of execution delay and the penalty for dropping the task. Denote the QoE-cost of the whole system at tth time slot as $\text{cost}_{\text{sum}}^t$, which is the sum of QoE-cost of each mobile device (denoted as cost_i^t). Therefore, we can obtain the following equation:

$$\text{cost}_{\text{sum}}^t \triangleq \sum_{i \epsilon \mathcal{N}} \text{cost}_i^t = \sum_{i \epsilon \mathcal{N}} \left[\mathcal{D}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t) + \emptyset \cdot \text{I}\left(\zeta_i^t \cap I_{i,d}^t\right)\right], t \in \mathcal{T}, i \in \mathcal{N}, \tag{12}$$

where $\emptyset$ denotes as the weight of task dropping cost and $\mathcal{D}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t)$ is given by

$$\mathcal{D}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t) = \text{I}(\zeta_i^t = 1) \cdot (I_{i,l}^t D_{i,\text{local}}^t + I_{i,r}^t D_{i,\text{remote}}^t), t \in \mathcal{T}, i \in \mathcal{N}. \tag{13}$$

3.7 Optimization model

In our system environment, each MEC server has more computing power than each mobile device, i.e., while the cost of offloading the task is still lower than the cost of local execution, the more computation tasks executed remotely, the better user's quality of experience. We can obtain the number of offloading computation tasks at tth time slot by

$$\sum_{i \in \mathcal{N}} \text{I}\left(\zeta_i^t \cap I_{i,r}^t\right), t \in \mathcal{T}, \tag{14}$$

where $\text{I}(\cdot)$ is the indicator function. Therefore, we have two optimization goals, i.e., the minimization of the average weighted sum QoE-cost and the maximization of the number of offloading computation tasks.

Denote the system operation at the tth time slot as

$$\boldsymbol{SO}^t \triangleq [\boldsymbol{I}^t, \boldsymbol{f}^t, \boldsymbol{p}^t, C^t, \boldsymbol{e}^t], t \in \mathcal{T}, \tag{15}$$

where $\boldsymbol{I}^t \triangleq [I_1^t, \dots, I_N^t]$, $\boldsymbol{f}^t \triangleq [\boldsymbol{f}_1^t, \dots, \boldsymbol{f}_N^t]$, $\boldsymbol{p}^t \triangleq [p_1^t, \dots, p_N^t]$, $\boldsymbol{e}^t \triangleq [e_1^t, \dots, e_N^t]$. Consequently, the optimization problem can be expressed as:

$$\mathcal{P}_1: \min_{\boldsymbol{SO}^t} \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \text{cost}_{\text{sum}}^t - \psi \cdot \sum_{i \in \mathcal{N}} \text{I}\left(\zeta_i^t \cap I_{i,r}^t\right)$$

$$s.t.\ (1),(2),(3),(8),(10)$$
$$0 \le f_{i,v}^t \le f_{\text{mobile}}^{\max}, t \in \mathcal{T}, i \in \mathcal{N}, v \in \{1, \dots, LX_{\text{mobile}}\} \quad (16)$$
$$\mathcal{E}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t) \le E_{\max}, t \in \mathcal{T}, i \in \mathcal{N} \quad (17)$$
$$0 \le p_i^t \le p^{\max}, t \in \mathcal{T}, i \in \mathcal{N} \quad (18)$$
$$I_{i,c}^t \in \{0,1\}, c \in \{l, r, d\}, t \in \mathcal{T}, \quad (19)$$

where ψ defined as the weight of the second optimization goal. (16) and (18) incarnate the constraints of mobile devices' CPU-cycle frequency and maximum transmit power, respectively. (17) incarnates the upper bound of battery discharging for security reasons, i.e., the amount of energy output energy cannot exceed $E_{\max}$ at each time slot. (19) represents the 0-1 indicator constraint which has been described in subsection 3.2.

4 Online Algorithms for Execution Cost and Fairness Optimization

In this section, we will develop LODCO-Based Greedy Algorithm and LODCO-Based ϵ-Greedy Algorithm to solve $\mathcal{P}_1$ on account of LODCO algorithm [3]. First of all, we will convert the original problem which is time-dependent to a deterministic problem $\mathcal{P}_2$ by taking advantages of Lyapunov Optimization. Then the LODCO algorithm will be upgraded and reconstructed for the multi-user and multi-server MEC system by virtue of Greedy Strategy. We will not demonstrate the details of LODCO algorithm, but we will explain closely about why we can apply it to our model.

4.1 Drift plus penalty formula

Lyapunov optimization technologies demands that the allowable action sets are i.i.d., which cannot be satisfied by the time-dependent battery queues of mobile devices. Therefore, for each mobile device we use the perturbation parameter θ [1](which is lower bounded by $\tilde{E}_{\max} + V\emptyset$ in [3]) to define to virtual battery queue $\tilde{b}_i^t$ by

$$\tilde{b}_i^t \triangleq b_i^t - \theta, t \in \mathcal{T}, i \in \mathcal{N}, \quad (20)$$

where $\tilde{E}_{\max} \triangleq \min\{\max\{kW(f_{\text{mobile}}^{\max})^2, p^{\max}\tau_d\}, E_{\max}\}$.

Besides, if a task requested at the tth time slot is being executed locally, the optimal frequencies of the LX_{mobile} CPU cycles should be the same, i.e., $f_{i,v}^t = f_i^t, i \in \{1, \dots, LX_{\text{mobile}}\}$, which can be obtained by *Inequality of arithmetic and geometric means*.

According to the analysis above, we define the Lyapunov function as

$$L(t) \triangleq \frac{1}{2}\sum_{i\in\mathcal{N}}\left(\tilde{b}_i^t\right)^2 = \frac{1}{2}\sum_{i\in\mathcal{N}}(b_i^t - \theta)^2, t \in \mathcal{T}. \quad (21)$$

Thus, the conditional Lyapunov drift can be written as

$$\Delta(t) \triangleq \mathbb{E}\left[L(t+1) - L(t)\middle|\tilde{b}^t\right], t \in \mathcal{T}, \quad (22)$$

where $\tilde{b}^t \triangleq [\tilde{b}_1^t, \dots, \tilde{b}_N^t]$. Then the Lyapunov drift-plus-penalty function can be written as

$$\Delta_V(t) \triangleq \Delta(t) + V \cdot \mathbb{E}\left[\text{cost}_{\text{sum}}^t\middle|\tilde{b}^t\right], t \in \mathcal{T}. \quad (23)$$

Because of the energy evolution equation (11), we can obtain that

$$\left(\tilde{b}_{i+1}^t\right)^2 \le \left(\tilde{b}_i^t\right)^2 + 2\tilde{b}_i^t\left(e_i^t - \mathcal{E}(\boldsymbol{I}_i^t, f_i^t, p_i^t)\right) + (e_i^t)^2 + \mathcal{E}^2(\boldsymbol{I}_i^t, f_i^t, p_i^t), t \in \mathcal{T}, i \in \mathcal{N}. \quad (24)$$

[1] the perturbation parameter θ is proposed in [3] to circumvent the above issue, which is that the vanilla version of Lyapunov Optimization technologies cannot be applied directly. The process to obtain the value of θ is an important component of LODCO Algorithm.

Because $\tilde{E}_{\max}$ denotes as the real energy consumption's upper bound consumed by ith mobile device at tth time slot, we have $\mathcal{E}(\boldsymbol{I}_i^t, f_i^t, p_i^t) \leq \tilde{E}_{\max}$. Since $e_i^t \leq E_{i,H}^t$ and $E_{i,H}^t$'s are i.i.d. among different time slots with the maximum value of $E_H^{\max}$, so we can get $e_i^t \leq E_H^{\max}$. As a result, with some manipulations, we have

$$\Delta(t) \leq \sum_{i\epsilon\mathcal{N}} \mathbb{E}\left[\tilde{b}_i^t\left(e_i^t - \mathcal{E}(\boldsymbol{I}_i^t, f_i^t, p_i^t)\right)\middle|\tilde{b}^t\right] + C\,, t \in \mathcal{T}, \tag{25}$$

where $C = \frac{N}{2}(\tilde{E}_{\max} + E_H^{\max})^2$. Since (24), we can obtain that

$$\Delta_V(t) \leq \sum_{i\epsilon\mathcal{N}} \tilde{b}_i^t\left(e_i^t - \mathcal{E}(\boldsymbol{I}_i^t, f_i^t, p_i^t)\right) + V \cdot \mathbb{E}\left[\text{cost}_{\text{sum}}^t\middle|\tilde{b}^t\right] + C\,, t \in \mathcal{T}, \tag{26}$$

which means $\Delta_V(t)$ is upper bounded.

Obviously, the upper bound of $\Delta_V(t)$ has the same structure with the problem defined in [3]. Thus, the LODCO Algorithm can hopefully be applied to deicide $\boldsymbol{SO}^t$ by solving the following deterministic problem

$$\mathcal{P}_2\colon \min_{\boldsymbol{SO}^t} \sum_{i\epsilon\mathcal{N}} \tilde{b}_i^t\left(e_i^t - \mathcal{E}(\boldsymbol{I}_i^t, f_i^t, p_i^t)\right) + V \cdot \mathbb{E}\left[\text{cost}_{\text{sum}}^t\middle|\tilde{b}^t\right], t \in \mathcal{T}, i \in \mathcal{N},$$

which subjects to every constraint conditions of $\mathcal{P}_1$.

4.2 Application of LODCO Algorithm

$\mathcal{P}_2$ is an optimized format of $\mathcal{P}_1$ by Lyapunov Optimization except maximizing the number of offloading computation tasks. According to LODCO Algorithm, the problem can be decomposed to two sub problems. The first one is to find the optimal energy harvesting, and the second one is to decide the optimal computation modes.

Optimal energy harvesting: the optimal amount of harvested energy e_i^{t*} for ith mobile device can be obtained by solving the following problem:

$$\min_{0\leq e_i^t \leq E_{i,H}^t, i\epsilon\mathcal{N}, t\epsilon T} \sum_{i\epsilon\mathcal{N}} \tilde{b}_i^t e_i^t.$$

Because each mobile device's decision on optimal energy harvesting is mutually independent, the optimal e_i^{t*} can be obtained separately for each mobile device. Thus, we can utilize the conclusion of [3] directly, i.e., we obtain e_i^{t*} for each mobile device by (21) in [3].

Decide the computation modes: we need to obtain the mode with the minimum value of $J_{CO}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t)$ for each mobile device, where

$$J_{CO}(\boldsymbol{I}_i^t, \boldsymbol{f}_i^t, p_i^t) \triangleq \mathrm{I}\left(I_{i,l}^t\right) \cdot J_{\text{mobile}}^t(f_i^t) + \mathrm{I}\left(I_{i,r}^t\right) \cdot J_{\text{server}}^t(p_i^t) + \mathrm{I}\left(I_{i,d}^t\right) \cdot V\emptyset, t \in \mathcal{T}, i \in \mathcal{N}, \tag{27}$$

where $J_{\text{mobile}}^t(f_i^t)$ and $J_{\text{server}}^t(p_i^t)$ denote as the sub problems of local-execution mode and remote-execution mode, respectively. Both of them are constituent parts of the LODCO Algorithm.

We have the same Lyapunov structure with [3], thus we can obtain the optimal $\boldsymbol{SO}^t$ when each mobile device can make decision separately. However, there exists correlation between any two mobile devices when choosing the computation modes, especially the offloading computation mode. In order to solve the problem, we propose LODCO-Based Greedy Algorithm for the multi-user and multi-server system. Beyond that, we propose the modified version, i.e., LODCO-Based ϵ-Greedy Algorithm to maximize the number of offloading computation tasks by virtue of theoretical knowledge of Reinforcement Learning.

4.3 LODCO-Based Greedy Algorithm

In this section, we demonstrate the details of proposed algorithm.

Algorithm 1: LODCO-Based Greedy Algorithm

At the beginning of time slot t, initialize flag[M] with 0 and establish a map to store the indexes of mobile device and corresponding chosen MEC server.
for each mobile device i **do**
  Obtain the task request indicator ζ_i^t, virtual energy queue $\tilde{b}_i^t$ and harvestable energy $E_{i,H}^t$.
  Generate the location of each mobile device and compute the distance $d_{i,j}^t$ between the ith mobile devices and the jth MEC server.
  Obtain the optimal harvested energy e_i^{t*} by the LODCO Algorithm.
  Obtain the optimal f_i^{t*} for local execution by the LODCO Algorithm, then record the optimal value $J_{\text{mobile}}^t(f_i^t)$. If the battery energy level is insufficient for local execution, set $J_{\text{mobile}}^t(f_i^t)$ as **inf**.
  Obtain the optimal f_i^{t*} for local execution by the LODCO Algorithm, then record the optimal value $J_{\text{mobile}}^t(f_i^t)$ (assume that mobile devices have no correlation now).
  for each MEC server j **do**
    Obtain the channel power gain $h_{i,j}^t$ from ith mobile device to jth MEC server by $h_{i,j}^t = \gamma_{i,j}^t g_0 (\, d_0/d_{i,j}^t)^\theta$.
    Obtain the optimal $p_{i,j}^{t*}$ from ith mobile device to jth MEC server by the LODCO Algorithm, then record the optimal value $J_{\text{server}}^t(p_{i,j}^t)$. If the battery energy level is insufficient for offloaded execution from ith mobile device to jth MEC server, set $J_{\text{server}}^t(p_{i,j}^t)$ as **inf**.
    Choose the optimal p_i^{t*} by selecting the one with minimum $J_{\text{server}}^t(p_{i,j}^t)$, denote as $J_{\text{server}}^t(p_i^t)$ and then record j.
  end for
  Compare $J_{\text{mobile}}^t(f_i^t)$, $J_{\text{server}}^t(f_i^t)$ and $V\emptyset$, choose the mode with the minimum value and set the corresponding indicator variable $I_{i,c}^t$ as 1 (within the power limit).
  if $I_{i,r}^t = 1$ **then**
    obtain the ith mobile device and the corresponding jth MEC server, then insert them into the map with key i and value j.
  end if
end for
Calculate the upper bound of the MEC server S_{UB} by $S_{\text{UB}} \triangleq \lfloor f_{\text{server}}^{\max} \tau / \, LX_{\text{server}} \rfloor$.
while the map is not **NULL do**
  Obtain the key-value pair "i-j" with the minimum $J_{\text{server}}^t(p_i^t)$.
  Compare $J_{\text{mobile}}^t(f_i^t)$, $J_{\text{server}}^t(p_i^t)$ and $V\emptyset$, choose the mode with the minimum value and set the corresponding indicator variable $I_{i,c}^t$ as 1 (we have to do the search again because it is possible that $J_{\text{server}}^t(p_i^t)$ has been modified).
  if $I_{i,r}^t = 1$ **then**
    if flag[j] $\leq S_{\text{UB}}$ **then**
      Remove the key-value pair "i-j" from the map and flag[j]++. Then set $J_{\text{server}}^t(p_{i,j}^t)$ as **inf**.
    else

26: **if** min$\{J^t_{\text{server}}(p^t_{i,:})\}$!= **inf then**[1]
27: Find the optimal j by min$\{J^t_{\text{server}}(p^t_{i,:})\}$ and the insert them to the map. Then **continue**.
28: **else**
29: Select the optimal mode from other 2 modes: local execution and dropping the task. Then remove the corresponding key-value pair from map.
30: **end if**
31: **end if**
32: **else**
33: Keep the corresponding $I^t_{i,c} = 1$ without change and then remove the corresponding key-value pair from map.
34: **end if**
35: **end while**
36: Calculate the value of $\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta^t_i \cap I^t_{i,r})/\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta^t_i)$, $\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta^t_i \cap I^t_{i,d})/\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta^t_i)$ and $\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta^t_i \cap I^t_{i,l})/\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta^t_i)$.
37: Obtain each mobile device's execution cost and energy consumption.
38: Update the battery level for each mobile device.
39: Update t to $t+1$.

As shown in Algorithm 1, we can use the LODCO Algorithm to obtain the optimal e^{t*}_i and f^{t*}_i because they are independent to each other. We can still use the LODCO Algorithm to obtain $p^{t*}_{i,j}$ for ith mobile device no matter which MEC server is chosen, then we use the Greedy Policy choose the best p^{t*}_i among those optimal $p^{t*}_{i,j}$.

4.4 LODCO-Based ϵ-Greedy Algorithm

In this section, we will demonstrate the details of LODCO-Based ϵ-Greedy Algorithm. There exists the Exploration-Exploitation dilemma in the theoretical model of K-armed bandit [13], which is a typical *Single-Step Reinforcement Learning Task.* Thus, *ϵ-Greedy Strategy* was proposed to achieve a trade-off. In the same way, we view the optimization target of $\mathcal{P}_2$ and the number of computation offloading tasks as "exploitation" and "exploration", respectively. Then, we can obtain the following algorithm based on LODCO-Based Greedy Algorithm.

Algorithm 2: LODCO-Based ϵ-Greedy Algorithm

1: Run step. 1 ~ step. 18 in Algorithm 1.
2: **while** the map is not **NULL do**
3: Obtain the key-value pair "i-j" with the minimum $J^t_{\text{server}}(p^t_i)$.
4: **if** rand() $< \epsilon$ **then**
5: **if** flag[j] $\leq S_{\text{UB}}$ **then**
6: Remove the key-value pair "i-j" from the map and then flag[j]++ no matter whether $J^t_{\text{server}}(p^t_i)$ is the minimum among $J^t_{\text{mobile}}(f^t_i)$, $J^t_{\text{server}}(p^t_i)$ and $V\emptyset$. Then set $J^t_{\text{server}}(p^t_{i,j})$ as **inf**.
7: **else**
8: **if** min$\{J^t_{\text{server}}(p^t_{i,:})\}$!= **inf then**

[1] $J^t_{\text{server}}(p^t_{i,:})$ is defined as $[J^t_{\text{server}}(p^t_{i,1}), J^t_{\text{server}}(p^t_{i,2}), \cdots, J^t_{\text{server}}(p^t_{i,M})]$.

Find the optimal j by $\min\{J^t_{\text{server}}(p^t_{i,:})\}$ and the insert them to the map. Then **continue**.
else
Select the optimal mode from other 2 modes: local execution and dropping the task. Then remove the corresponding key-value pair from map.
end if
end if
else if rand()$\geq \epsilon$ **then**
Run step. 20 ~ step. 33 in Algorithm 1.
end if
end while
Calculate the value of $\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta_i^t \cap I_{i,r}^t)/\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta_i^t)$, $\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta_i^t \cap I_{i,d}^t)/\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta_i^t)$ and $\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta_i^t \cap I_{i,l}^t)/\sum_{i\in\mathcal{N}} \mathrm{I}(\zeta_i^t)$.
Obtain each mobile device's execution cost and energy consumption.
Update the battery level for each mobile device.
Update t to $t+1$.

5 Simulation results

In this section, we will demonstrate the results of the proposed algorithms and verify their effectiveness. Then, we will show the impacts of the system parameters by control variable method. As mentioned in section 4, we will not elaborate upon the details about the verification of LODCO Algorithm.

The simulation was run on a machine with an Intel Core 2.5 GHz i7-4710MQ CPU. The algorithm was implemented in MATLAB R2015b and was given up to 8 GB of memory if needed.

In our system, the harvestable energy $E_{i,H}^t$ is uniformly distributed with the maximum value of $E_H^{\max}$, where $E_H^{\max}$ can be obtained by average EH power P_H, i.e.,

$$E_H^{\max} = P_H \cdot 2\tau, t \in \mathcal{T}. \tag{28}$$

We assume that $P_H = 12$ mW, $g_0 = -40$ dB (path-loss constant), $\mathcal{s} = 10^{-28}$ (effective capacitance coefficient), $\tau = \emptyset = 2$ ms (time slot length and the weight of the task dropping cost, respectively). In addition, $\omega = 1$ MHz (system bandwidth), $\sigma = 10^{-13}$ W (noise power at each MEC server), $f_{\text{mobile}}^{\max} = f_{\text{server}}^{\max} = 1.5$ GHz (upper bounds of each mobile device and each MEC server's CPU-cycle frequency, respectively), $E_{\max} = 2$ mJ (the real energy consumption's upper bound consumed by mobile device at each time slot), $L = 1000$ bits (size of each computation task), $V = 10^{-5}$ (coefficient of the penalty in Lyapunov Optimization), and $X_{\text{server}} = X_{\text{mobile}} = 5900$ cycles per byte (numbers of CPU cycles required by each mobile device and each MEC server). Besides, we assume there are 10 mobile devices and 5 MEC servers, i.e., $N = 10$, $M = 5$, and $E_{\min} = 0.02$ mJ [1].

According to the above parameter values, the upper bound S_{UB} is 4 by $S_{\text{UB}} \triangleq \lfloor f_{\text{server}}^{\max}\tau / LX_{\text{server}} \rfloor$, which describes the maximum number of mobile devices that can be connected. In the following part, the default value of ρ (task request probability) and ϵ in

[1] $E_{\min}$ is the non-zero lower bound of mobile devices defined in [3].

Algorithm 2 are 0.6 and 0.25 unless stated, respectively. $\gamma_{i,j}^{t}$ (the small-scale fading channel power gains at tth time slot) is exponentially distributed with mean 1. Besides, we assume the maximum distance between random mobile and random MEC server is 100 m unless stated, which is the upper limit for the uniform distribution.

5.1 Validation of Effectiveness

In this subsection, we will verify the effectiveness of LODCO-Based Greedy Algorithm and LODCO-Based ϵ-Greedy Algorithm compared with LODCO Algorithm. As shown in Fig. 2, the battery energy level of each mobile device, which is the mean value of the proposed two algorithms, demonstrate the feasibility of our improving on LODCO Algorithm. The energy level of each mobile device keeps accumulating at earlier stage, and finally stabilizes around the perturbed energy level at about 180th time slot, which exactly keeps the advantages of LODCO Algorithm.

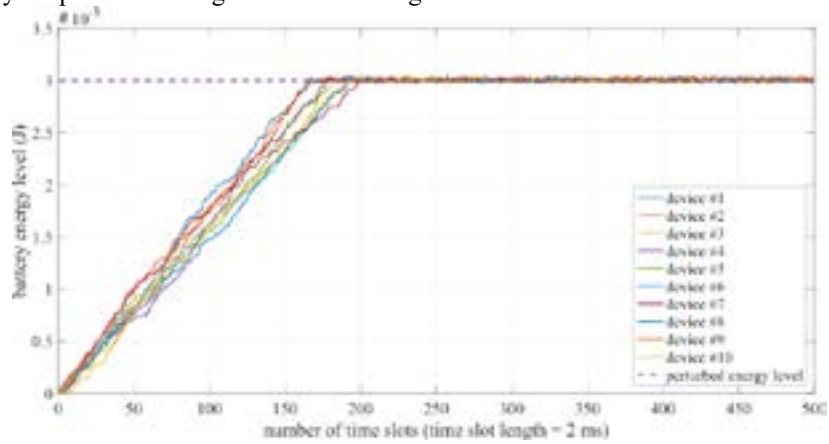


Fig. 2. Battery energy level of each mobile device vs. time.

As depicted in Fig. 3, the Y-axis describes the average value of 1500 time slots of each mobile device's energy level. As seen, each energy level is confined within [0,0.005][1] J, which conforms to the theoretical results derived in LODCO Algorithm.

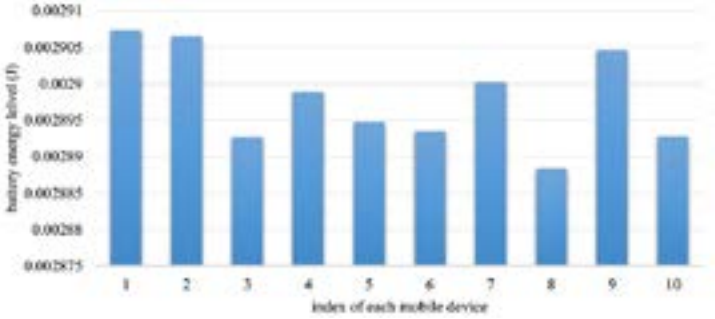


Fig. 3. Average energy level of each mobile device.

Fig. 4 demonstrates the ratio of each chosen modes in our multi-user and multi-server system by LODCO-Based ϵ-Greedy Algorithm. At the very earlier stage, plenty of

[1] 0.005 is the specific value of $\theta + E_H^{\max}$.

computation tasks are dropped due to the insufficient energy level. Then the ratio of dropped tasks significant decreases to 0 along with the ascending battery energy level of each mobile device. Meanwhile, the ratio of offloading tasks clearly greater than the ratio of locally-executed tasks and the average ratio of offloading tasks obtained by LODCO Algorithm [1], which means that LODCO-Based ϵ-Greedy Algorithm is able to obtain better performance than LODCO Algorithm in terms of getting the most of offloading computation number.

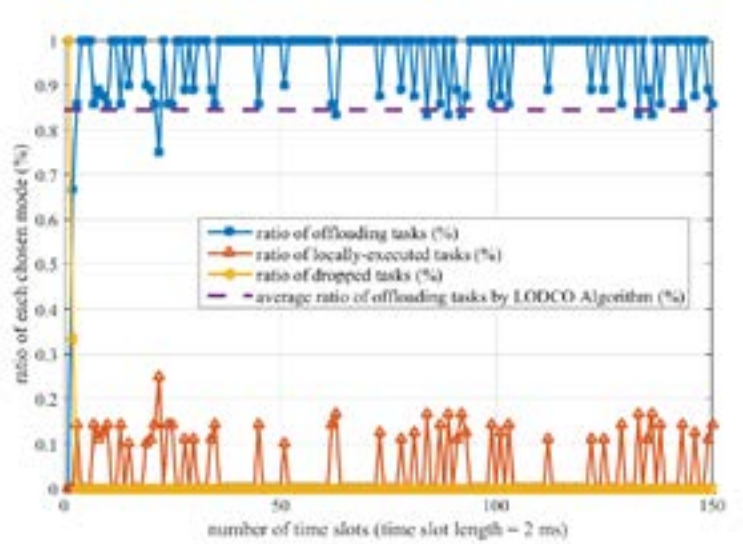


Fig. 4. The ratio of each chosen modes vs. time.

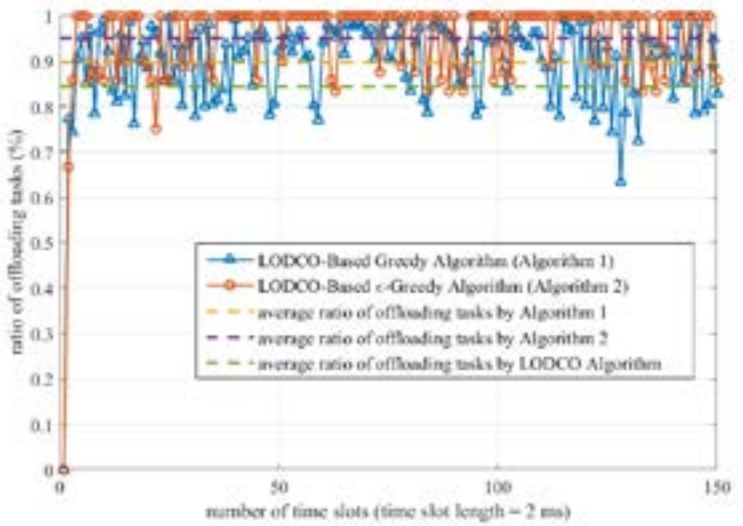


Fig. 5. Comparison between 2 proposed algorithms.

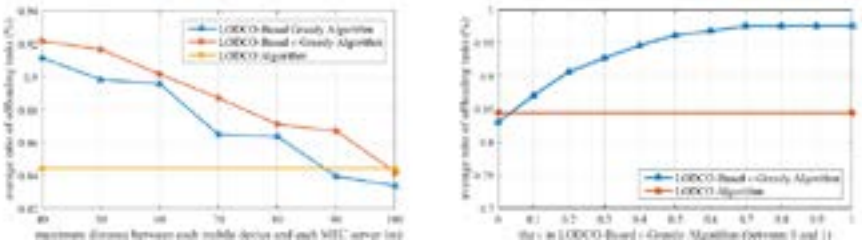
[1] The difference values are approximately 5% and 10%, respectively.

We compare the performance of LODCO-Based ϵ-Greedy Algorithm with that of LODCO-Based Greedy Algorithm on the second optimization goal, i.e., the number of offloading computation tasks.

As depicted in Fig. 5, the average ratio of offloading tasks obtained by LODCO-Based ϵ-Greedy Algorithm (which is 96.0341%) is greater than obtained by LODCO-Based Greedy Algorithm (which is 87.6544%), where both of them are larger than the average ratio of offloading tasks obtained by LODCO Algorithm (which is 84.4401%). The results verify that our second algorithm can obtain better offloading computation tasks number than the first algorithm by ε-Greedy Strategy can do.

5.2 Effects of system parameters

In this subsection, we will demonstrate the impacts of system parameters on the performance of proposed algorithms.



(a) (b)

Fig. 6. Average ration of offloading tasks vs. maximum distance and ϵ.

Fig. 6(a) depicts the impact of the maximum distance between random mobile device and random MEC server. As seen, along with the increase of the maximum distance, the average ratio of tasks offloaded gradually decrease. When the distance is arbitrarily far, the number of offloading computation tasks will be zero. The reason is that the channel power gain grows with the distance between each mobile device and each MEC server, which will lead to larger energy consumption and longer execution delay. As a result, more and more mobile devices will choose to execute the computation tasks locally.

As depicted in Fig. 6(b), along with the increase of ϵ, which belong to $[0,1]$, the average ratio of offloading tasks gradually increase with a slowdown rate and finally converge to the specific value 97.0981%. The reason is that LODCO-Based ϵ-Greedy is based on the ϵ-Greedy Strategy, i.e., a greater ϵ will bring a lager probability to choose the offloading mode.

6 Conclusions

In this paper, we investigated a mobile-edge computing systems with multi-user and multi-server. Then we proposed two algorithms to obtain the lowest execution cost and largest number of offloading computation modes based on LODCO Algorithm, i.e., LODCO-Based Greedy Algorithm and LODCO-Based ϵ-Greedy Algorithm. Those two algorithms are online algorithms with low-complexity. Most importantly, they have no need of too much priori knowledge. By extensive simulation and performance analysis, we can see that those two algorithms inherit every advantage from LODCO Algorithm and adapt to the more complex environment perfectly and offer more than 5% and 10% ratio of offloading

computation tasks, respectively. LODCO-Based ϵ -Greedy Algorithm can choose the offloading mode as far as possible, which can bring resource-limited MEC servers' superiority into full play. In conclusion, our study provides a viable suggestion to design a complex system which is much more approachable to reality.

References

[1] Y.C. Hu, M. Patel, D. Sabella, et al., 'Mobile edge computing – A key technology towards 5G', ETSI White Paper, 2015.

[2] T.X. Tran, H. Abolfazl, P. Pandey, et al., 'Collaborative mobile edge computing in 5G networks: New paradigms, scenarios, and challenges', IEEE Communication Magazine, 2017, 55(4): 54-61.

[3] Y.Y. Mao, J. Zhang, K. B. Letaief, 'Dynamic computation offloading for mobile-edge computing with energy harvesting devices', IEEE Journal of Selected Areas Communications, 2016, 34(12): 3590-3605.

[4] S. Sudevalayam, P. Kulkarni, 'Energy harvesting sensor nodes: Survey and implications', IEEE Communications Surveys & Tutorials, 2011, 13(3): 443-461.

[5] X. Ge, S. Tu, G. Mao, et al., '5G ultra-dense celluar networks', IEEE Wireless Communications, 2016, 23(1): 71-79.

[6] C.F. Liu, M. Bennis, H.V. Poor, 'Latency and reliability-aware task offloading and resource allocation for mobile edge computing', IEEE Global Communications Conference Workshops (GLOBECOM Workshops), Singapore, 2017.

[7] M. Masoudi, B. khamidehi, C. Cavdar, 'Green cloud computing for multi cell networks', IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, USA, 2017.

[8] T.X. Tran, D. Pompili, 'Joint task offloading and resource allocation for multi-server mobile-edge computing networks', arXiv: 1705.00704, 2017.

[9] L. Yang, J.N. Cao, Z.Y. Wang, et al., 'Network aware multi-user computation partitioning in mobile edge clouds', International Conference on Parallel Processing (ICPP), Bristol, United Kingdom, 2017.

[10] Y.Y. Mao, J. Zhang, K. B. Letaief, et al., 'A survey on mobile edge computing: The communication perspective', IEEE Communication Surveys & Tutorials, 2017, 19(4): 2322-2358.

[11] L.B. Huang, M.J. Neely, 'Utility optimal scheduling in energy-harvesting networks', IEEE/ACM Transactions on Networking, 2013, 21(4): 1117-1130.

[12] W.W. Zhang, Y.G. Wen, K. Guan, et al., 'Energy-optimal mobile cloud computing under stochastic wireless channel', IEEE Transactions on Wireless Communications, 2013, 12(9): 4569-4581.

[13] S. Bubeck, N. Cesa-Bianchi, 'Regret analysis of stochastic and nonstochastic multi-armed bandit problems', Foundations and Trends® in Machine Learning, 2012, 5(1): 1-22.

Research on the Relationship between Producer services Subdivision industry and Manufacturing based on Lotka-Volterra Model

Xueyuan Wang[1], Rui Ma[1], Ting He[2], Bin Qiu[3]

[1]School of Economics and Management, Harbin University of Science and Technology, Harbin, 150080 China

[1]College of Computer Science and Technology, Huaqiao University, Xiamen, 361021 China

[3]Beijing Shengzhou Aerospace Software Technology Co., Ltd., Beijing 100094 China

Abstract: The producer services industry and the manufacturing industry are interconnected; subdivision industries of producer services industry have different effects on the manufacturing industry. In order to accurately determine the relationship among them, and to guide China government to correctly make industrial planning, a variable analysis model is built based on theoretical analysis. Using increased value of industry to calculate their interacting relation after data smoothness treatment based on Lotka-Volterra model and Eviews software. The result shows that the relationship between manufacturing, technology services, and finance is predator-prey relation, while manufacturing and transportation & warehousing are in mutually reinforcing and symbiosis condition. After discovering their current relationship, recommendations for optimizing industrial relations in order to achieve industrial cooperation and mutual development are brought forward.

Keywords: manufacturing industry; service industry; symbiosis

1 Introduction

The producer services industry is independent from the manufacturing and service departments of the manufacturing industry, and now has become an emerging industry. From the experience of developed countries, after years of manufacturing industry development, service industry will take the leading place instead of manufacturing industry. Therefore, identifying and determining the current relationship between the two and predicting their further interacting trends, has great significance for reasonably making program and plans to promote the mutual and beneficial development of the two.

2 Construction of Symbiotic Model of Producer services and Manufacturing Industry

2.1 The theoretical basis of model construction

First of all, through the development of scientific research and technical (SRT) service sector, the process of self-innovation and intellectual property protection of manufacturing industries can be accelerated, product technological value can be increased, and industrial premium ability can be improved (Xing and Zhang, 2017) [1]; Therefore, to make up for the shortcomings of manufacturing technology, the integrated technology service should be adopted and developed to make full use of industry, university, and research institute technological resources, and shorten the transition period of scientific and technology achievements (Wang and Tan,2013) [2], and promote the rising of manufacturing industry to the high end of the value chain. Secondly, a reasonable development of the financial intermediation (FI) service sector can provide high-quality financing service for the manufacturing industry. On the one hand, manufacturing scale can expand and economies scale effect will emerge based on financial support (Vallejo and Arias-Pérez, 2017) [3]. On the other hand, for the help of financing, manufacturing industry could attract talents, enhance R&D capabilities, upgrade industry structure and achieve great-leap-forward development. Finally, developing transport, storage and post (TSP) service sector and strengthening the construction of railways, aviation, waterways and land transportation can provide high-quality and convenient transportation for manufacturing industry, which is helpful to integrate resources in the manufacturing process and improve product delivery efficiency and enhance industrial competitiveness.

In view of the major problems faced by China manufacturing enterprises in R&D quality, industrialization input, and product delivery cost, as well as the importance of the three-representative producer services subdivision industries to solve the problems for manufacturing enterprises development, SRT service sector, FI service sector and TSP service sector are selected in this paper to make relationship research with manufacturing industry.

2.2 The construction of a quantitative model

Manufacturing industry and producer services industries rely on economic and social enjoinment, they form an industrial ecosystem through the interaction of material flow, knowledge flow, and capital flow (Pang, 2012) [4]. Therefore, the relationship between producer services industry and manufacturing industry is similar with the ethnic groups relationship in ecology, which means the Lotka-Volterra model in

biological system can better reveal the quantitative relationship between the two industries (Suh and Kim, 2015) [5].

The Lotka-Volterra model can be constructed by simultaneous equations, where X is the development scale of manufacturing industry, Y is the development scale of a producer services sector (subdivision industry), r_1 is the growth rate of manufacturing industry, and r_2 is the growth rate of producer services sector. K_1 represents the largest amount of the manufacturing industry capacity that the environment can accommodate, and K_2 represents the largest amount of the producer services sector that the environment can accommodate. θ_1 denotes the coefficient of the producer services sector influencing manufacturing industry, and θ_2 denotes the coefficient of the manufacturing industry influencing the producer services sector. Assuming that the environmental variables remain unchanged, which means the market resources and policy requirements for the manufacturing industry and producer services sectors will remain unchanged, the following formula can be obtained:

$$\frac{dX}{dt} = F_1(X,Y) = r_1 X\left(\frac{K_1 - X + \theta_1 Y}{K_1}\right) \tag{1}$$

$$\frac{dY}{dt} - F_2(X,Y) - r_2 Y\left(\frac{K_2 - Y + \theta_2 X}{K_2}\right) \tag{2}$$

After the replacement and treatment, the formula can be deformed into the following ones:

$$dX/_{dt} = X(a_1 - b_1 X - c_{12} Y) = a_1 X - b_1 X^2 - c_1 XY \tag{3}$$

$$dY/_{dt} = Y(a_2 - b_2 Y - c_2 X) = a_2 Y - b_2 Y^2 - c_2 YX \tag{4}$$

3 Empirical Analysis

3.1 Data selection

The relevant data was selected from the China Statistical Yearbook and the China Industrial Economic Statistics Yearbook. The nonlinear least squares method and Eviews software is used for parameter estimation, and the results can be obtained.

3.2 Model Calculation and results analysis

(1) The estimated parameters of the relationship between manufacturing industry and SRT service sector are shown in Table 1.

Table 1 parameter estimation results of manufacturing industry and SRT

Dependent Variable: P2(1) Method: Least Squares				Dependent Variable: P6(1) Method: Least Squares			
	Coefficient	t-Statistic	Prob.		Coefficient	t-Statistic	Prob.
a_1	1.144262	26.27212	0.0000	a_2	1.027551	11.27664	0.0000
b_1	-0.071994	-1.959284	0.0858	b_2	0.121118	4.188853	0.0030
c_1	0.060279	3.511515	0.0079	c_2	-0.215679	-3.334688	0.0103
R-squared	0.998753			R-squared	0.997580		

From the estimation results, it can be seen that the two industries are developing well and the scale continues to grow; the manufacturing industry is developing faster. The current state of the manufacturing industry has a self-promoting effect, and the development trend of the SRT service sector is slightly inhibited. There is a predatory relationship between manufacturing industry and the SRT service sector.

(2) The estimated parameters of the relationship between manufacturing industry and FI service sector are shown in Table 2.

Table 2 parameter estimation results of manufacturing industry and FI service sector

Dependent Variable: P1 Method: Least Squares				Dependent Variable: P3 Method: Least Squares			
	Coefficient	t-Statistic	Prob.		Coefficient	t-Statistic	Prob.
a_1	1.169707	69.00239	0.0000	a_1	1.011111	16.14755	0.0000
b_1	-0.000692	-1.957380	0.0586	b_2	0.000409	3.202947	0.0030
c_1	0.000316	5.365356	0.0000	c_2	-0.003464	-3.920332	0.0004
R-squared	0.999258			R-squared	0.992921		

According to the estimation results, it can be seen that both the manufacturing industry and FI service sector are developing rapidly. There are weak self-promotion and self-suppression effects. The relationship between the manufacturing industry and the FI service sector is predator-prey.

(3) The estimated parameters of the relationship between manufacturing industry and TSP service sector are shown in Table 3.

Table 3 parameter estimation results of manufacturing industry and TSP service sector

Dependent Variable: P1 Method: Least Squares				Dependent Variable: P4 Method: Least Squares			
	Coefficient	t-Statistic	Prob.		Coefficient	t-Statistic	Prob.
a_1	1.143640	43.54485	0.0000	a_2	1.299659	26.90864	0.0000
b_1	0.005003	4.825943	0.0000	b_2	0.008201	3.744159	0.0007
c_1	-0.003889	-3.789680	0.0006	c_2	-0.007902	-3.548787	0.0012
R-squared	0.999032			R-squared	0.997237		

According to parameter estimation results, the development of TSP service sector is faster than the manufacturing industry; Although the existing industrial scale has certain influence on the market future supply and demand, the effects are not significant; The two are in a mutually reinforcing status.

4 Conclusions and recommendations

Using the model, the development status and interacting relation of the manufacturing industry and the producer services industry are measured. Accordingly, the following conclusions can be obtained:

(1) There is a self-promoting function in the development of the manufacturing industry, while a weak self-suppression effect exits in SRT service sector. With the rapid industry development, it is necessary to continuously improve the industry quality and avoid falling into stagnation and hard breaking out. SRT, FI and TSP service sectors need to effectively integrate with the real economy, to prevent inflationary bubbles and stagnation.

(2) Manufacturing industry has occupying effect to the SRT and FI service sector. The financial industry needs to break through downturn investment to entity economy. The manufacturing industry needs to separate its R&D and service departments from itself and provides external scientific and technological services to create new economic growth point.

(3) The manufacturing industry and the TSP service sector develop coordinately and synergistically.

References

[1] Xing, Y and Zhang, H. (2017) 'Producer Services FDI and Manufacturing Export Technical Progress: Threshold Effect Based on Intellectual Property Rights Protection', *Science of Science & Management of S & T*, Vol.38 No.8, pp.29-45

[2] Wang, B and Tan, Q. (2013) 'The Effect of Property Type, Firm Size, and Industrial Agglomeration on the Transformation Efficiency of Patent——The Data from High-Tech Industry', *Economic Management Journal*, Vol.35 No.8, pp.153-161

[3] Vallejo, A and Arias-Pérez, J. (2017) 'Approach to differences in product and process innovation capabilities and financial performance in manufacturing companies', *Espacios*, Vol.38 No.4, pp.11-13

[4] Pang, B. (2012) 'Research About Symbiotic Evolutionary Model of Producer Services and Manufacturing Industry in China', *Chinese Journal of Management Science*, Vol.20 No.2, pp.176-183

[5] Suh, Y and Kim, M. (2015) 'Dynamic change of manufacturing and service industries network in mobile ecosystems: The case of Korea', *Telematics & Informatics*, Vol. 32 No.4, pp.613-628

An Approach for Identifying the Abstraction Scopes of Business Process Petri Nets System Using Binary Search Tree

Huan Fang, Shuya Sun, Lulu He, Xianwen Fang

School of Mathematics & Big data, Anhui University of Science and Technology, Huainan city, Anhui province, China.

Abstract: Since it is difficult to form a quick overview understanding for large and complex business process models, the studies of the technologies and methods of model abstraction are crucial. The state-of-the-art abstraction studies are mostly concentrated on the abstraction method for various process systems, however they are a little vague about the scope orientation that is to be abstracted in the model. A search-tree-based abstraction scope identification method for a business process model is purposed in the paper, which is founded on the basis of behavioral relation theory of Petri nets and the Depth-First Search ideas. First, the concepts of three kinds of block structures and boundary places in work-flow Petri net systems are formalized, and the transition association tree of the system is then obtained. The transition association tree is further used to identify and locate the areas that are to be abstracted in the model, and the aim of model abstraction is then accomplished. Finally, case examples are applied to illustrate the validity and feasibility of the proposed method. Therefore, compared to the existing studies, the main contributions of this study are a sound block-based abstraction method and its corresponding block identification method, on which the well-performed properties of initial systems can be preserved after abstraction, and the proposed methods are in polynomial time complexity.

Keywords: abstraction scopes identification, abstraction, business process system, behavioral profile, Petri nets.

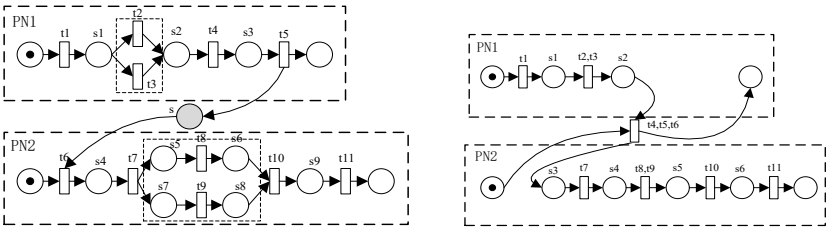
Reference to this paper should be made as follows: Fang H, Sun S Y, He L L, Fang X W. (xxxx) 'An Approach for Identifying the Abstraction Scopes of Business Process Petri Nets System Using Binary Search Tree', *Int. J. xxxxxxxxxx xxxxxxxxxx*,

1 Introduction

Large BPMSs may contain tens or hundreds of active elements, and these elements capture different aspects of features of the process model, such as structures, functions, resources or data information(Weber 2008,Mendling, Mendling2010). Various personalized process perspective views designed for different kinds of stakeholders appeared to be distinct, for example, the outline sketch views of a given process system are favored by business managers, while detailed ones are more inclined to be accepted by the process' participants(Polyvyanyy 2009). Therefore, in actual application, in order to iron out these difficulties, the sketch views of BPMS are put forward to simplify a complex process, that is the Business Process Model Abstraction (BPMA for short) in a

given conciseness degree. By BPMA(Li 2012, Smirnov2012, Zerguini 2004), it is easy to achieve different perspectives of the business process models.

An limitation is that the methods mentioned above lack the semantics understanding for a given BPMS, i.e., it is dim to indicate which elements should be abstracted into a higher level of activity. For example, Fig. 1(a) depicts a model that contains two components, where place *s* is a communication place accomplishing the interactive semantics between two counterparts. By using the method purposed of Sergey et al.2012, transitions *t*4, *t*5 and *t*6 can be simplified into a high-level transition in Fig 1. (b), however this abstraction operation seems to be unsuitable in real life, as the distinct works of different branches are rarely joined into a new branch due to the administration branches in enterprises.



(a)A BPMS model (2)An abstraction model by methods of Sergey et al.2012

Figure 1. An example for illustrating unsuitable abstraction

2. Transition association binary search tree

Algorithm 1: Boundary Place Identification Algorithm, BPI algorithm

Input: WF-systems $\Sigma = \{\Sigma_1, \Sigma_2, \cdots, \Sigma_n\}$,an initial visit place s.

Output: Boundary places set $S_{boundary}$.

Procedures:

Step1: Initialize a place visit vector visit[n], where n is the number of places;

Step2: for $\forall s \in \Sigma : visited[s] = False$;

Step3: Visit the node from the first one successively, if v_i satisfies visited[v_i]=False, then call function BPI(Σ, v_i)recursively; or else visit the next node;

Step4: All feasible traces $ext\sigma$ in Σ are obtained from Step 3. Comparing the set of all $ext\sigma$, the boundary places s_{ij} of Σ_i, Σ_j is calculated as $\forall ext\sigma_i \in \Sigma_i, \forall ext\sigma_j \in \Sigma_j : s_{ij} = ext\sigma_i \bigcap ext\sigma_j \bigcap S_i \bigcap S_j$ ($i \neq j$);

Step5: $S_{boundry} = \bigcup_{i=1, i\neq j}^{n} \bigcup_{j=1}^{n} s_{ij}$.

Theorem1: Let $T_\Sigma - tree$ be the transition association tree of WF-system Σ. If the number of ∞ of $T_\Sigma - tree$ is m, then there are $m+1$ components in Σ, which interact with each other by boundary places.

3. Method of identifying regions to be abstracted

Rule 1. Identification rule for Choice blocks *SB*

Rule 1.1: If t_1, t_2 are in a sequential block *SB*, denoted as $\{t_1, t_2\} \in SB$. $\forall t_1, t_2 \in T_\Sigma$ - $tree$, if the following two conditions are both satisfied: (1) $(t_1.left = t_2) \wedge (t_2.right = NIL)$, (2) $d(t_1, t_2) \neq \infty$.

Rule 1.2: if $\{t_1, t_2\} \in SB \wedge \{t_2, t_3\} \in SB$, then $\{t_1, t_2, t_3\} \in SB$.

Rule 2. Identification rule for Choice blocks *ChB*, Concurrent blocks *CB*

For $\forall t_1, t_2, t_3 \in T_\Sigma$ - $tree$:

Rule 2.1: if both of $t_1.right = t_2$ and $d(t_1, t_2) \neq \infty$ are satisfied, then $\{t_1, t_2\} \in ChB \vee \{t_1, t_2\} \in CB$;

Rule 2.2: if $\{t_1, t_2\} \in ChB \wedge \{t_2, t_3\} \in ChB$, then $\{t_1, t_2, t_3\} \in ChB$;

Rule 2.3: if $\{t_1, t_2\} \in CB \wedge \{t_2, t_3\} \in CB$, then $\{t_1, t_2, t_3\} \in CB$.

Rule 3. Abstraction sequence for Choice blocks *ChB* and Concurrent blocks *CB*

For $t_1, t_2, t_3, t_4 \in T_\Sigma - tree$:

Rule 3.1: if either $\{t_1, t_2\} \in ChB \wedge \{t_2, t_3\} \in ChB \wedge \{t_2, t_4\} \in SB$ or $\{t_1, t_2\} \in CB \wedge \{t_2, t_3\} \in CB \wedge \{t_2, t_4\} \in SB$ holds, then the abstraction operation of sequential blocks are prior to the concurrent blocks or choice blocks.

Rule 3.2: if either $\{t_1, t_2\} \in SB \wedge \{t_2, t_3\} \in SB \wedge \{t_2, t_4\} \in ChB$ or $\{t_1, t_2\} \in SB \wedge \{t_2, t_3\} \in SB \wedge \{t_2, t_4\} \in CB$ holds, then the abstraction operation of the concurrent blocks or choice blocks are prior to sequential blocks.

4. Property analysis

Theorem 2. Let $N=(S,T;F)$, $N'=(S',T';F')$ be the Petri net model before and after STA operation respectively; M_0 is the initial marking of N, and the initial marking M_0' of N' satisfies $\forall s'\in S': M_0(s')=\sum_{\exists s_i\in S\wedge h(N_{s_i},s_i)=s} M_0(s_i)$, where N_{s_i} is a subset of N, that is $N_{s_i}\subset S\cup T$.

(1) if $\sum=(N,M_0)$ is a sound WF-PN, then $\sum'=(N',M_0')$ is also a sound one.

(2) if $\sum=(N,M_0)$ is a live (or deadlock free), then $\sum'=(N',M_0')$ is also live (or deadlock free).

(3)if $\sum=(N,M_0)$ is a safe WF-PN, then $\sum'=(N',M_0')$ is also a safe one.

Theorem 3. Let $N=(S,T;F)$, $N'=(S',T';F')$ be the WF-PN before and after STA algorithm respectively, M_0 is the initial marking of N, and the initial marking M_0' of N' satisfies $\forall s'\in S': M_0(s')=\sum_{\exists s_i\in S\wedge h(N_{s_i},s_i)=s} M_0(s_i)$, in which N_{s_i} is a subset of N, that is $N_{s_i}\subset S\cup T$. If $\sum=(N,M_0)$ is a well-performed WF-PN, then $\sum'=(N',M_0')$ is also a well-performed one.

Theorem 4. If $\sum=(N,M_0)$, $\sum'=(N',M_0')$ are the Petri net models before and after STA algorithm respectively, then $behavior(\Sigma)\underset{abstraction}{\equiv} behavior(\Sigma')$.

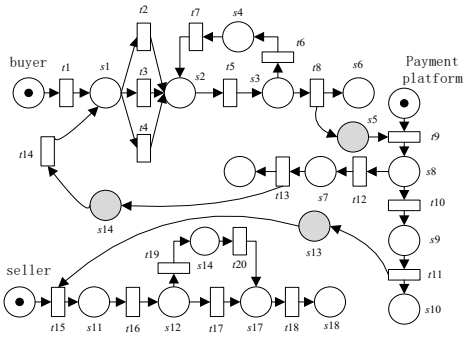
5. Case study

Fig. 8 depicts a payment BPMS, and the practical use semantics can be referred in the work of Fang et al.2018. The meaning of each transition is manifested in table 1.

6. Conclusions and discussions

The complexity of the process model brings different degrees of difficulty to the user's rapid understanding and analysis, so it is a practical and crucial perspective to study the simplification and abstraction method for BPMSs. Based on previous research, this paper presents a search-tree based abstraction scope identification method, whose main target is determining the areas that are to be abstracted and preserving the well-performed properties of the BPMS. The main contributions can be concluded as the following: (1) three kinds of sound block structures are purposed, which are sequential block, concurrent block and choice block; (2) a kind of transition association DFS tree $T_\Sigma-tree$ and its binary counterpart $T_\Sigma-bitree$ are formalized, which help to locate the boundary places and preserve the interactive semantics of BPMS; and (3) an identification method for determining the scopes that are to be abstracted to simplify the complexity of BPMS.

In future work, we aim to investigate techniques for computing block structures that do not satisfy SESE, that is, the systems that do not meet the basic assumptions of sound WF systems in this work. Furthermore, we aim to find an abstraction method that can consider more complicated relationships than the three weak orders in the behavior profile, such as behavior inclusion and casual behavior relationships.



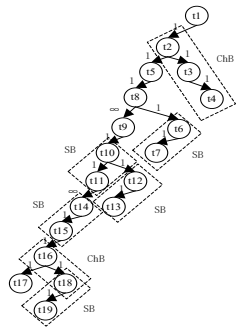


Figure 8. A payment BPMS model. Figure 9. The blocks identification in Fig.8.

Reference

Fang H, He L L, Fang X W, Wang L L.(2018)'A search-tree-based abstraction method for business process Petri nets models',*Control Theory &Application*,Vol.35 No.1,pp.92-102.

Li J., Zhou M.C., Dai X. Z. (2012),'Reduction and Refinement by Algebraic Operations for Petri Net Transformation',*IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*,Vol.42 No.5,pp.1244-1255.

Mendling J. (2007) On the Detection and Prediction of Errors in EPC Business Process Models.*EMISA Forum*, Vol. 27, No.2,pp.52-59.

Mendling, J. Reijers, H A. and van der Aalst W. M. P.(2010) 'Seven Process Modeling Guidelines (7PMG)'. *Information and Software Technology*, Vol. 52, No.2, pp.127-136.

Polyvyanyy A., Smirnov S.(2009), 'Weske M. The Triconnected Abstraction of Process Models'.*International Conference on Business Process Management*. Springer-Verlag, pp.229-244.

Sergey S., Matthias W., Jan M. (2012),'Business Process Model Abstraction Based On Synthesis From Well-Structured Behavioral Profiles',*International Journal of Cooperative Information Systems*,Vol.21 No.1,pp.55-83.

Smirnov S., Reijers H. A., Mathias W.(2012),'Business Process Model Abstraction: a Definition, Catalog and Survey',*Distributed and Parallel Databases*,Vol.30 No.1,pp.63-99.

Weber, B. and Reichert, M.(2008) 'Refactoring Process Models in Large Process Repositories'. *International Conference on Advanced Information Systems Engineering, CAISE 2008, Montpellier, France,* June 16-20,pp.124-139.

Zerguini L.(2004),'A Novel Hierarchical Method For Decomposition And Design Of Workflow Models', *Journal of Integrated Design & Process Science*,Vol.8 No.2,pp.65-74.

Log Automaton under Conditions of Infrequent Behavior Mining

Xianwen Fang, Juan Li, Lili Wang

School of Mathematics & Big data, Anhui University of Science and Technology, Huainan, 232001, China.

Abstract: In the existing process mining methods, infrequent behaviors are often considered as noise is ignored, but some infrequent behaviors have an important role in business process management. Firstly, the knowledge of log automaton is applied to the low-frequency log to delete infrequent behavior in the logs; secondly, the processed logs are added into attributes. Then, the condition-dependent value of the communication characteristics of different module networks is compared with the threshold, and the effective infrequent log is retained to optimize the model. Finally, a practical case is applied, which indicates the effectiveness and validation of the proposed method.

Keywords: process mining; Log automaton; infrequent behavior; Conditional dependency measure.

Reference to this paper should be made as follows: Xianwen Fang, Juan Li, Lili Wang. ' Log Automaton under Conditions of Infrequent Behavior Mining', *Proceedings of the 11th International Conference on Service Science, 2018*

1 Introduction

Most contemporary process mining techniques are paying more attention to discover frequent behaviors which reveal common parts of a process. There are several algorithms[1,2] have been proposed to discover frequently behaviors. Most of infrequent behaviors have been removed to reduce the complexity of the model without a high decrease in the fitness[3,4]. Some techniques are of limited use in real-life setting because they assumed noise-free event logs(e.g., the Alpha[5])in fact, event logs often contain noise[6]However, experiments show that infrequent behavior can monitor and enhance a process. (detecting intrusions in the networks[7].) WoMine-I algorithm[9] to retrieve infrequent behavioral patterns form a process model. This proposal has been validated with a set of synthetic and real process models. a novel approach has proposed algorithm[10] to detect deviation on the event level by identifying frequent common behavior and uncommon behavior among executed process instances, what more The approach is implemented in Prom and was evaluated in a controlled setting with artificial logs and real-life logs. A data-aware heuristic mining algorithm (DHM)[11], which is a process discovery method that uses data attributes to distinguish between infrequent paths and random noise.The existing infrequency behavioral discovery studies mostly focus on noise and abnormal behavior processing, but there are certain limitations in practical

applications. and less attention to the availability of low-frequency behavior, Our focus is on finding effective infrequent behavior from infrequent behavior in the logs.

2 Preliminaries

In this section we recall some related[12,14] , and the others are omitted for simplicity.

Definition 2.1. (Log Automaton) A log automaton for an event log is defined as a directed graph. $A = (\Gamma, \to)$.

Definition 2.2. (Conditional dependency measure). Given activities $a,b \in \sum$ and dependency conditions C. We define $a \Rightarrow^{C,L} b : \sum \times \sum \to [-1,1]$ as the strength of the causal dependency from a to b under condition $C_{a,b}(x)$ in the event log:

$$a \Rightarrow^{C,L} b = \begin{cases} \dfrac{\left|a >^{C,L} b\right| - \left|b >^{C,L} a\right|}{\left|a >^{C,L} b\right| + \left|b >^{C,L} a\right| + 1} & for\ a \neq b \\ \dfrac{\left|a >^{C,L} a\right|}{\left|a >^{C,L} a\right| + 1} & otherwise \end{cases}$$

3 Conditional infrequent behavior mining algorithm based on log automaton

Infrequent behavior is divided into effective infrequent behavior and ineffective infrequent behavior. Deleting effective infrequent behavior leads to the deletion of core information in the business process. This causes certain negative impact on the accuracy and applicability of the business process, and cannot reach the expectation of enterprise or modeler. In order to solve this problem, the classification process of infrequent behavior eliminates invalid infrequent behavior that is not conducive to the execution of business processes, and retains effective infrequent behavior, which is great significance to the development and optimization of business processes. Specific algorithm is as follows.

Algorithm: behavior mining algorithm under infrequent log.

Input: Source model M_0 and threshold θ_1 , θ_2 .

Output: Optimized Business Process Communication Model M_T .

Step1: Query all executable traces in the source model M_0, and discard the traces of imperfect events. Getting the executable event logs recorded $L = \{\alpha_1,\ \alpha_2,\ \alpha_3 \cdots \alpha_n\}$, $n \in N_+$,and the infrequent logs recorded as $L = \{\alpha_1,\ \alpha_2,\ \alpha_3 \cdots \alpha_k\}\ k \in N_+$..

Step2: According to Definition 2.2 Convert Infrequent Logs $L = \{\alpha_1,\ \alpha_2,\ \alpha_3 \cdots \alpha_k\}$ to Log Automaton.

Step3: Calculate the arc frequency between different modules of communication network according to the formula: $c(x,y) = \dfrac{2 \times \#_{\to}(x,y)}{\#_{\Gamma}(x) + \#_{\Gamma}(y)}$, $C_0 = \{c_1,\ c_2,\ c_3 \cdots c_n\}$.

Step4: Comparing the frequency of the obtained arc $C_0 = \{c_1,\ c_2,\ c_3 \cdots c_n\}$ with θ_1 , if $c_i < \theta_1,\ \ i \in [1,n]$ that c_i is an infrequent arc, denoted as $C_1 = \{c_1,\ c_2,\ c_3 \cdots c_k\}\ k \in [1,n]$.

Step5: 5.1Remove infrequent arcs $C_1 = \{c_1,\ c_2,\ c_3 \cdots c_k\}$ in the log automaton.

5.2The infrequent behavior in the log is deleted under the conditions of maintaining connectivity and maintaining the required state.

5.3Get filtered infrequent logs $L' = \{\sigma_1',\ \sigma_2',\ \sigma_3' \cdots \sigma_k'\}$.

Step6: Add the related attributes to infrequent log $L' = \{\sigma_1',\ \sigma_2',\ \sigma_3' \cdots \sigma_k'\}$.

Step7: Calculate the frequency of occurrence of following relation $a >^{C,L} b$ $\left|a >^{C,L} b\right| = |\{e \in E \mid \#_{act}(\bullet(e)) = a \wedge \bullet(e) \neq \perp \wedge \#_{act}(e) = b \wedge C_{a,b}(val(e)) = 1\}|$

Step8: The condition dependency metric $a \Rightarrow^{C,L} b$ from activity a to activity b.

Step9: If $0 < \left|a \Rightarrow^{C,L} b\right| < \theta_2$ then the conditional dependency metric from a to b indicates that the activity pair is weak for dependencies.Otherwise $\left|a \Rightarrow^{C,L} b\right| > \theta_2$ indicates that the activity pair is strong for dependencies.

4 Case Analysis

Event log traces described in table1 depicts a car-hailing app process model, it contains 58 transitions and 3 modules (passenger, service, driver).

We can detect 4 infrequent logs $L = \{\sigma_4,\ \sigma_5,\ \sigma_6,\ \sigma_7\}$ from Table 1 because frequencies have less than 30. Figure 1 is based on 4 infrequent logs we draw out the log automaton.

In order to improve log executable, infrequent behavior in the log should be removed after deleting infrequent arcs while maintaining the required state set connectivity. We can get the processed log $L' = \{\sigma_4',\ \sigma_5',\ \sigma_6',\ \sigma_7'\}$ in Table 2.

Table 1 Event Log Traces

coding	Frequency	Event log traces
σ_1	1999	$t_{11}t_{12}t_{14}t_{13}t_{15}t_{17}t_{1.13}t_{1.18}t_{25}t_{29}t_{28}t_{38}t_{39}t_{3.10}t_{1.19}t_{1.20}t_{1.21}t_{1.22}t_{1.23}t_{1.26}t_{1.27}$
σ_2	2711	$t_{31}t_{32}t_{2.11}t_{2.12}t_{2.13}t_{2.14}t_{36}t_{39}t_{3.10}t_{1.19}t_{1.20}t_{1.21}t_{3.13}t_{3.14}$
σ_3	1025	$t_{31}t_{32}t_{3.4}t_{3.7}t_{3.8}t_{3.9}t_{3.10}t_{1.19}t_{1.20}t_{1.21}t_{3.13}t_{3.14}$
σ_4	24	$t_{1.1}t_{1.2}t_{1.2}t_{1.3}t_{2.3}t_{1.4}t_{1.5}t_{1.6}t_{1.9}t_{1.18}t_{2.5}t_{2.6}t_{2.8}t_{2.4}t_{3.8}t_{3.9}t_{3.10}t_{1.19}t_{1.20}t_{3.11}t_{1.21}t_{3.13}t_{1.22}t_{1.26}t_{1.27}$
σ_5	8	$t_{1.1}t_{1.2}t_{1.2}t_{1.3}t_{2.3}t_{1.4}t_{1.5}t_{1.7}t_{1.9}t_{1.18}t_{2.5}t_{2.6}t_{2.8}t_{3.8}t_{3.9}t_{3.10}t_{1.19}t_{1.21}t_{3.11}t_{1.21}t_{1.22}t_{2.4}t_{1.26}t_{1.27}$
σ_6	10	$t_{1.1}t_{1.2}t_{1.2}t_{1.2}t_{1.9}t_{2.3}t_{1.3}t_{1.4}t_{1.5}t_{1.8}t_{1.15}t_{1.7}t_{1.18}t_{2.5}t_{2.6}t_{2.7}t_{3.9}t_{3.10}t_{1.19}t_{1.20}t_{3.11}t_{1.21}t_{3.13}t_{1.22}t_{1.25}t_{1.26}t_{2.4}t_{1.2}$
σ_7	2	$t_{1.1}t_{1.2}t_{1.3}t_{1.2}t_{2.2}t_{2.3}t_{1.4}t_{1.5}t_{1.6}t_{1.10}t_{1.11}t_{2.5}t_{1.18}t_{3.8}t_{2.8}t_{1.19}t_{3.10}t_{3.11}t_{1.20}t_{1.21}t_{1.22}t_{1.23}t_{1.24}t_{1.26}t_{1.27}$

According to the step3 of the algorithm, calculate the frequency of each arc in different modules of the communication Petri net with interactive relationship in the log automaton. In Figure 1these arcs which frequency less than $\theta_1 = 0.4$ are marked with a cross to delete.In order to improve log executable, infrequent behavior in the log should be removed. We can get the processed log $L' = \{\sigma_4',\ \sigma_5',\ \sigma_6',\ \sigma_7'\}$ in Table 2.

Table 2 Filtered log traces

coding	Frequency	Filtered log traces
σ'_4	24	$t_{1.1}t_{1.2}t_{1.2}t_{1.3}t_{2.3}t_{1.4}t_{1.5}t_{1.6}t_{1.9}t_{1.18}t_{2.5}t_{2.6}t_{2.8}t_{3.8}t_{3.9}t_{3.10}t_{1.19}t_{1.20}t_{3.11}t_{1.21}t_{3.13}t_{1.22}t_{1.26}t_{1.27}$
σ'_5	8	$t_{1.1}t_{1.2}t_{1.2}t_{1.3}t_{2.3}t_{1.4}t_{1.5}t_{1.7}t_{1.9}t_{1.18}t_{2.5}t_{2.6}t_{2.8}t_{3.8}t_{3.9}t_{3.10}t_{1.19}t_{3.11}t_{1.21}t_{1.22}t_{1.26}t_{1.27}$
σ'_6	10	$t_{1.1}t_{1.2}t_{1.2}t_{1.2}t_{2.3}t_{1.4}t_{1.5}t_{1.8}t_{1.15}t_{1.7}t_{1.18}t_{2.5}t_{2.6}t_{2.7}t_{3.9}t_{3.10}t_{1.19}t_{1.20}t_{3.11}t_{1.21}t_{3.13}t_{1.22}t_{1.25}t_{1.26}t_{1.27}$
σ'_7	2	$t_{1.1}t_{1.2}t_{1.3}t_{1.2}t_{2.3}t_{1.4}t_{1.5}t_{1.6}t_{1.10}t_{1.11}t_{3.8}t_{2.8}t_{1.20}t_{1.21}t_{1.22}t_{1.23}t_{1.24}t_{1.26}t_{1.27}$

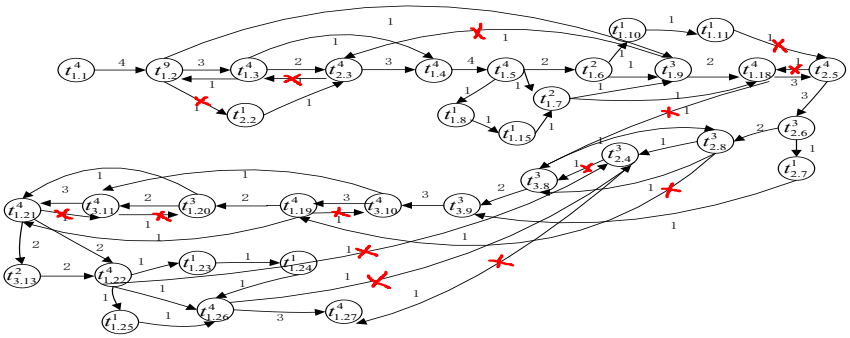


Figure 1 the log automaton built by infrequent logs L

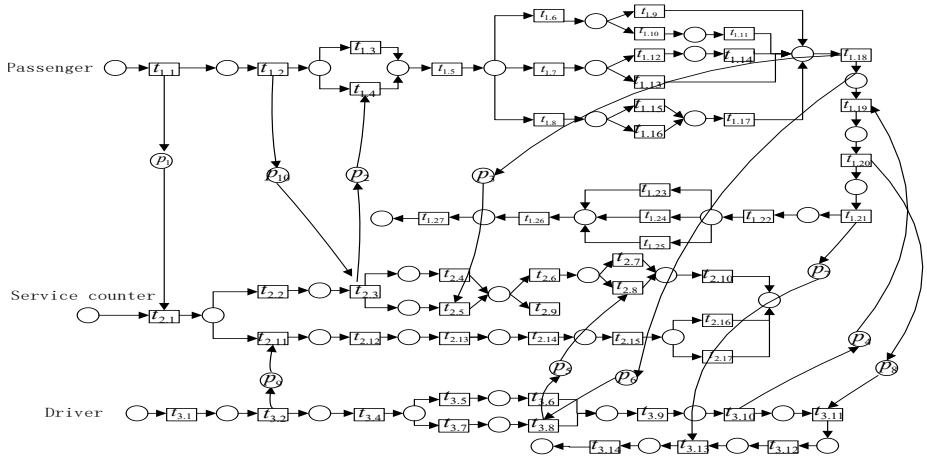
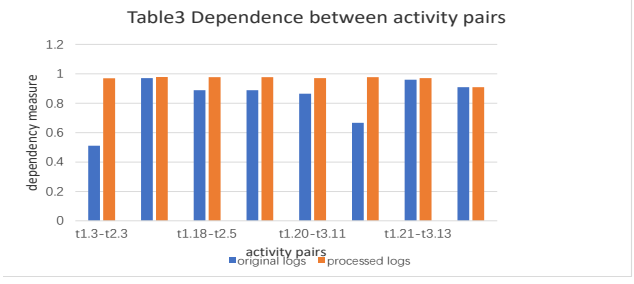


Figure 2 Optimization of A car-hailing app business process model



5 Conclusion

In this paper we presented an algorithm designed to search effective infrequent behaviors which would be useful in many domains of process mining. We have compared the technique with other traditional methods, showing that our approach discovers uncommon behavior that other technique is not able to detect. We propose a more scientific method to deal with infrequent behaviors by using log automaton and conditional directly measure. In Business process the results show significant improvement over fitness. In the future, we plan to consider other approach of mining infrequent behaviors.

Acknowledgement

This work is partially supported by the National Natural Science Foundation of China under Grant No.61572035, No.61402011, Anhui Provincial Natural Science Foundation (1608085QF149), the Huainan Science & Technology Project(No. 2016A23).We also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

Reference

8.Adriansyah, A A. (2014) 'Aligning observed and modeled behavior',*Technische Universiteitndhoven*, 2014.

2. Conforti, R., Rosa, M L. and Hofstede, A H. M T. (2017) 'Filtering Out Infrequent Behavior from Business Process Event Logs'. *IEEE Transactions on Knowledge and Data Engineering*,Vol.29 No.2,pp.300-314.

3. Ghionna, L., Greco, G. and Guzzo, A. (2008) 'Outlier Detection Techniques for Process Mining Applications', Foundations of Intelligent Systems, *International Symposium*, Ismis Toronto, Canada, pp.150-159

7. Jia, G., Cheng, G. et al. (2017) 'Traffic anomaly detection using k-means clustering', Vol.40 No.6,pp.403-410.

6. Suriadi, S.,et al.(2016) ' Event log imperfection patterns for process mining towards a systematic approach to cleaning event logs', *Information Systems*, 64pp.132-150.

9. Tax, N., et al. (2016) 'Mining local process models', J*ournal of Innovation in Digital Ecosystems*,Vol.3No.2,pp.183-196.

4. Vázquez-Barreiros, B., Mucientes, M. and Lama M. (2015) 'ProDiGen: Mining complete, precise and minimal structure process models with a genetic algorithm'. *Information Sciences*,pp.315-333.

5. Van der Aalst, W., Weijters, T., & Maruster, L. (2004). Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*,Vol.16No. 9,pp.1128-1142.

1. Wang, Y., et al. (2013) 'Investigating Service Behavior Variance in Port Logistics from a Process Perspective'*,International Conference on Business Process Management.* Springer, Cham, pp.318-329.

Quantifying the Emergence of New Domains: Using Cybersecurity as A Case

Xiaoli Hu[1,2], Shizhan Chen[1,2], Zhiyong Feng[1,3], Keman Huang*[,4]
[1]*Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin 300350, China*
[2]*School of Computer Science and Technology, Tianjin University, Tianjin 300350, China*
[3]*School of Computer Software, Tianjin University, Tianjin 300350, China*
[4]*Cybersecurity@MIT Sloan, MIT Sloan School of Management, Cambridge, MA 02142, USA*
{*xiaoli_hu, shizhan, zyfeng*}*@tju.edu.cn, keman@mit.edu*

***Abstract*—Understanding the evolution mechanism, especially the emergence of the new domains, is critical to promote the growth of the service ecosystem. Furthermore, the availability of various big scholarly data, enable us to develop methods to dig deep into the generation of new domains. In this paper, especially, we take cybersecurity disciplinary as an example, collect the relevant data from Microsoft Academic, and then extract the citation and reference relations among different domains. We further propose the domain-derived space, representing the inspiration relations for the emergence domains, by identifying the emergence of a domain and the significant derived relations. In the context of the developed domain-derived space, we develop methods to study the growth of a domain as well as the characteristics of its ancestral domains. The results reveal the dissipation of the interdisciplinary effect and the importance of domains in the early stage for the emergence of the new domains. This study suggests future research to understand the mechanisms of new service domains in the different ecosystems.**

***Keywords*-Domain-Derived Space; Cybersecurity; Service Ecosystem**

I. Introduction

Understanding the evolution patterns of the service ecosystem plays an important role in the applications of services, including service development, composition, and recommendation [1] [2]. Especially, the emergence of the new domains is even more critical for the service ecosystem as it will create new functionality and promote the growth of the service ecosystem. Though several typical service ecosystems, including web service ecosystem [3], mobile app ecosystem [4] and human service ecosystem [5], have been studied over these years, the emergence of the new domains is not studied yet. This is because of the unavailable of the necessary data, especially the missing of the relations among the different domains, makes it difficult to study these relations.

From a different perspective, with the availability of various big scholarly data, including citation networks, co-author networks, co-citation networks, co-words networks or hybrid networks, etc., we can look into science itself. One typical direction is to use the citations to understand the impact of scientific works [6], individual scientific influence [7] or the diffusion of the scientific knowledge [8]. More importantly, for our study of the ecosystem evolution mechanism, the citations also enable us to look into the relations among different domains. Though big scholarly data is not specific to the service ecosystems, the developed methodologies and the empirical results will suggest the further study on the service ecosystem evolutions. Hence, in this article, we intend to dig into the emergence pattern of the new domains in the scientific scholarly data to gain insight for the evolution procedure, focusing on how a new domain emergence, grow mature and inspire other new domains.

To do so, we focus on the cybersecurity disciplinary and collect 1,824,829 articles belonging to 798 different domains from Microsoft Academic [9], as the articles in cybersecurity disciplinary are organized into different domains and fields, which enables us to study the emergence of the new domains. We will discuss more details about the data in Section II. Based on the collected data, the first challenge is to identify the emergence of the new domains. It is straightforward to suppose that given a domain, its first article can be considered as its beginning. However, there may be no citation nor reference for this first article, making it an isolated point for the scientific ecosystem that we should not consider it as the validated beginning for the domain. Moreover, there exist some randomness for these first articles when they chose the references. These raise the first research question on understanding the emergence of the new domains:

Question 1: How to identify the validated, significant derived relations among the domains so that we can quantify the emergence of new domains?

To answer this question, based on the ideas that *"the articles which indicate the beginning of a domain should have impacts on this domain while their references should not belong to this domain"*, we propose an approach to identify the validated articles revealing the beginning of each domain, named *first articles*. Based on the references of these first articles, we further develop a methodology to reduce the randomness in the domain-derived relations, constructing the *"domain-derived space"* to represent the statistically validated derived relations for the emergence domains.

Furthermore, in the context of this domain-derived space, we seek for an in-depth understanding of the growth of these

new domains as well as the evolution patterns by asking the following question:

Question 2: How does the new domain grow into mature and how does it inspire new domains over time?

It is important to dig deep into the characteristics of the domains which inspire the emergence of the new domains, or ancestral domains. In another word, we are interested in what domains are more likely to drive the emergence of new domains. Therefore, our final research question is:

Question 3: What are the characteristics, including the dynamic patterns, of the ancestral domains which inspire the new domains' emergence?

Using the collected data and the approaches, we construct the domain-derived space. Examining this space reveals that: (1) there exist common patterns during the growth of the domains; (2) it is becoming difficult to generate new domains in a disciplinary; (3) the interdisciplinary effect is significant at the beginning of a domain but decreasing over time as the domain grows mature; (4) instead of inspired by the mature domains, the ancestral domains for the emergent ones tend to be in a relatively early stage. These empirical results will enable us to identify the potential new domains. Additionally, it suggests the further study to dig into the emergence of the new services, service domains for the ecosystem using the approaches developed in this article.

The remainder of this article is organized as follows: Section II describes the dataset we use for this study. Section III constructs the domain-derived space, including identifying the beginning of new domains and their significant derived relations. Section IV reports the methodologies for quantifying the domain production dynamics and analysis of the domain-derived trends. Section V reveals results about the ancestral domain dynamics to understand the domain-derived reasons and its early influence. Section VI discusses the related work and Section VII summarizes this article.

II. Background and Data Set

A. Microsoft Academic

Using entity mining, machine learning, and search technologies, Microsoft Academic is created as a semantic search engine to retrieve relevant information on academic articles, scientists, conferences, journals, and research domains [9]. It is also a test platform for many research ideas such as vertical search at the object level, extraction, and disambiguation of named entities, data visualization, etc.

Microsoft Academic includes 7 main types of entities [9]: *"Affiliation"* refers to the institution the author was affiliated with at the time when the article is published; *"Author"* refers to the individual author of a publication; *"Conference"* series refers to the name of academic conference; *"Field of study"* refers to the research fields, which are identified by publisher keywords and *Moving Average* algorithms. Moving Average[1] is a method for analyzing data points by creating a series averages of different subsets. Microsoft Academic further divides these fields into different *"domains"* by using the same method; *"Journal"* refers to the name of the scholarly journal; *"Title"* refers to the publication title; *"Year"* refers to the year of publication.

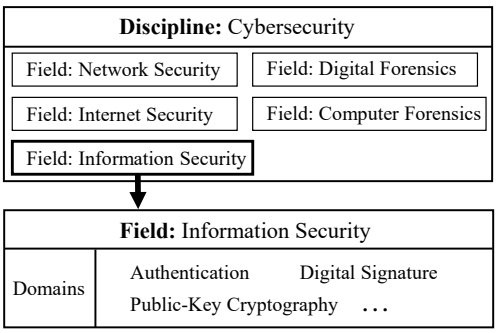


Figure 1. The Structural Characteristics of Dataset

B. Data Set

In this paper, we take the cybersecurity discipline as an example and collect the fields, domains, and articles in this discipline. As shown in Figure 1, cybersecurity contains multiple different types of fields, such as *"Network Security"*. Each field also can be divided into multiple domains, such as the field *"Information Securit"* can be divided into the domains *"Authentication"*, *"Digital Signature"* and *"Public-Key Cryptography"*. Finally, we succeed to get 1,824,829 articles, published from 1970 to 2016. We don't consider the articles before 1970 because the cybersecurity concept was firstly introduced in 1969 [10]. These articles are grouped into 798 fields, which in turn are clustered into 12 fields in our dataset.

For each article belonging to the *"first articles"*, which we will discuss more details in the next section, we crawled their citations and references so that we can identify the articles which inspire these first articles and those research which are influenced by them. This will enable us to dig deep into the emergence of the new domains. Note that, in this paper, we are only focusing on the cybersecurity discipline so that those articles which are not included in this discipline which be labeled as other disciplines and we don't distinguish their specific fields or domains.

III. Domain-Derived Space

Based on the product space presented by Hidalgo et al. [11], we propose the domain-derived space to understand how different domains perform derivative continuation to promote the continuous development and evolution of disciplines.

[1]https://en.wikipedia.org/wiki/Moving average.

Definition 1 (Domain-Derived Space (G_{sd})). Domain-Derived Space represents the derived dependency relations among different domains, which can be modelled as a directed network $G_{sd} =< S, E >$ consisting of nodes S representing the different domains and directed edges E representing the derived relations $e(i \rightarrow j)$ from domain i to j. In another word, domain i contributes to the emergence of the domain j. Hence, domain i can be considered as one ancestral domain for domain j.

A. Quantify the Emergence of Domains

For constructing the domain-derived space, the first task we need to do is to identify the beginning of a domain. Straightforwardly, the earliest article in a domain can be considered as the beginning of this domain. However, this is problematic because: (1) the emergence of a domain may not just because of the publishing of one article, but several influential articles; (2) some articles may not have any citation nor reference, which makes them an isolated article in the ecosystem; (3) through some articles may be grouped into the specific domain, they are never cited by this domain which means that they don't contribute to the domain's development. Without any influence on the domain, these articles should not be considered as the beginning of a domain. Following these discussions, we can define the *"first articles"* as the *earliest validated* articles which are not only the earliest articles belonging to this domain but also can contribute to its growth.

Definition 2 (First Articles $f\_a(j)$). If an article a is considered as one of the first articles for a domain j, it needs to meet these requirements:

(1) The article is cited at least once by other articles in the domain j.
(2) None of the article's references belongs to the domain j.
(3) There should be no articles meeting the above two requirements as well as published in an earlier year.

Algorithm 1 details the procedure to identify the first articles for a specific domain. For a specific article a, lines 3∼14 evaluate whether its published year is the earliest one and whether it has at least one citation belonging to the same domain j, indicating whether it belongs to the first articles for j or not. Note that it is possible that all the articles published in the first year of a domain cannot match the requirements. In this case, we will move to the next year until we obtain the first articles for the domain. Lines 15∼17 are designed for this logic.

B. Domains-Derived Dependency Network

Straightforwardly, if the article which belongs to one domain i is cited by one of the first articles $f\_a(j)$ in another domain j, the domain i can be considered as an ancestral domain for domain j which inspires the emergence of domain j, so that we can build a directed edge from domain

Algorithm 1 Determine First Articles of the Domain

Require: a: the article in the domain j; y_a: the published year of the article a; $Minyear$: the minimum of all y_a; $Maxyear$: the maximum of all y_a; ca: the citations of the a;
Ensure: $f\_a(j)$: the first articles for the domain j;

```
1: initial is_obtain ← 0, min ← 0;
2: for min ← Minyear to Maxyear do
3:     for each a do
4:         if y_a = min then
5:             for each ca do
6:                 if ca ∈ j then
7:                     is_obtain ← 1, add a to f_a(j);
8:                 end if
9:                 if is_obtain = 1 then
10:                    Break;
11:                end if
12:            end for
13:        end if
14:    end for
15:    if is_obtain = 1 then
16:        Break;
17:    end if
18: end for
19: return f_a(j);
```

i to domain j. The comparative advantage for $e(i \rightarrow j)$ can be evaluated as follows:

$$AD(i \rightarrow j) = \frac{WN(i,j)}{WN(j)}, \tag{1}$$

where $WN(i,j)$ refers to the number of references for articles in $f\_a(j)$ which belong to domain i, $WN(j)$ refers to the number of all references for articles in $f\_a(j)$. Hence $AD(i \rightarrow j)$ represents the confidence that domain i can be considered as the ancestral domain for domain j. If the reference does not belong to cybersecurity, it will be ignored because we can not identify its domain, and this will result in the disappearance of some edges, which in turn will make some domains become isolated nodes and be omitted. Note that to make the domain analyzable, we only choose domains which appeared in 1970∼2009, have an accurate emergence year as well as the total number of articles covered by the domain until 2016 is over 100 so that we can look into the evolution of its productivity. Using this way, we can achieve a directed network, named *Domains-Derived Dependency Network (DDDN)*, consisting of 587 nodes and 4,887 edges, with a 1.42% network density.

C. Domains-Derived Dependency Statistically Validated Network

In general, we can not judge the reason for an article refers to another is the relevance or the randomness. To

reduce this randomness for the domain-derived relations and generate a *Domains-Derived Dependency Statistically Validated Network (DDDSVN)*, for each first article in each domain, the *Fisher-Yates shuffle* algorithm is used to shuffle some references, which are selected from all references for all first articles, so that we can generate the random ancestral domains and then get the *Random Domain-Derived Dependency Network (RDDDN)*. Algorithm 2 shows the details of generating the *RDDDN*. Lines 02∼06 select those references which are published before the first article; lines 07∼13 shuffle these references, and then choose the random reference to replace the original reference to generate the random domain-derived relations among the domains. Finally, using this procedure for all the first articles, we can construct the *RDDDN* as the return.

Algorithm 2 Generate the *RDDDN*

Require: $f\_a(j)$: the first articles of the domain j; EY_j: the emergence year of the domain j; ra: the references for all first articles; Y_{ra}: the published year of each ra; $count\_fa$: the number of references for each $f\_a(j)$;

Ensure: *RDDDN*: the random domain-derived dependency network;

```
for each f_a(j) do
    for each ra do
        if Y_ra ≤ EY_j then
            generate the list of references ralist
        end if
    end for
    for count ← 1 to count_fa do
        use Fisher-Yates shuffle algorithm on the ralist;
        generate a random number m;
        take the m-th value in the ralist as the random reference, mapping to the domain i
        generate the random domain-derived relation e(i → j)
        add e(i → j) to RDDDN
    end for
end for
return RDDDN;
```

We repeat this procedure to generate $K = 10000$ random domain-derived dependency networks so that for each domain-derived relation $e(i \rightarrow j)$ in the original *DDDN*, we can get K random comparative advantage values: $\{RAD_k(i \rightarrow j)|k = 1, \ldots, K\}$. If the $e(i \rightarrow j)$ is not included in a *RDDDN*, we will set the random comparative advantage value as 0. Hence, the statistically significance for $e(i \rightarrow j)$ can be defined as follows:

$$p\_value(i \rightarrow j) = \frac{1}{K} \sum_{k=1}^{K} I(AD(i \rightarrow j) \leq RAD_k(i \rightarrow j)), \quad (2)$$

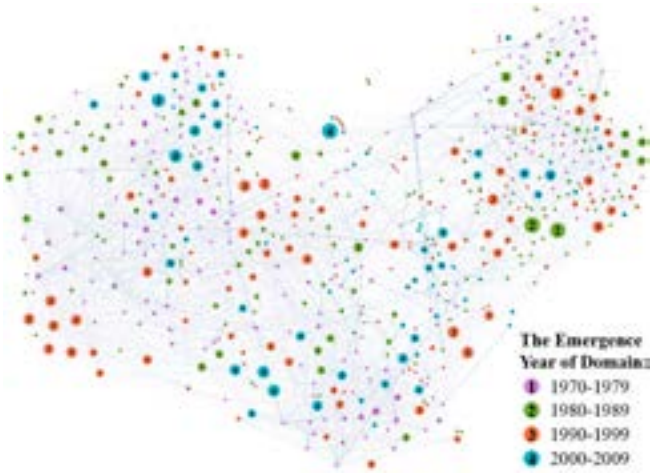


Figure 2. Domains-Derived Dependency Statistically Validated Network

where I is the counting function and $p\_value(i \rightarrow j)$ indicates the proportion that the given derived relation $e(i \rightarrow j)$ has a smaller or equal comparative advantage in the original *DDDN* than in the *RDDDN*. Finally, only the edge with $p\_value < 0.05$ can be considered as significant and included in the *DDDSVN*, some edges are removed compared to the original *DDDN*, for example, the edge from domain *Network Security Policy* to domain *Threat* has a 0.9952 p_value, which means that *Network Security Policy* should not be considered as the ancestral domain for *Threat*. Since these edges with high randomness are removed, the domain *Classified Information*, *Cross-zone Scripting*, and *Clipper Chip* are deleted so that leading to the network consisting of 584 nodes and 3,504 directed edges, with a 1.03% network density.

Figure 2 shows the overview of the *DDDSVN*, where each node represents the domain, and each edge represents the domain-derived relation. To make it clearer, we organize these domains into four groups, which will further discuss in the next section, and the nodes from different groups are allocated with the different colors. The size of the nodes represents the different *in-degree* values. As shown in Table I, the top five domains with the highest in-degree and out-degree, domain *Integrity Aware Parallelizable Mode (IAPM)* has the top in-degree, 168 while domain *Public-Key Cryptography (PKC)* has the top *out-degree*, 150. This is reasonable as the *PKC* is a very fundamental domain for the cybersecurity disciplinary. *IAPM* emerges as a domain in 2001 and requires knowledge from many different domains, likes *Encryption*, *Probabilistic Encryption*, and *40-bit Encryption*.

IV. Domain Production Dynamics

A. Domain Productivity

Based on the number of new articles published in each year, we can calculate the cumulative production $N_i(t)$ for

Table I
THE HIGHEST IN-DEGREE AND OUT-DEGREE OF DOMAINS

	In-degree	Out-degree
1	Integrity Aware Parallelizable Mode	Public-Key Cryptography
2	Spoofing Attack	Computer Security Model
3	Ip Address Spoofing	Key Distribution
4	MD5	Digital Signature
5	NSA Suite B Cryptography	Security Service

domain i, where $N_i(t)$ represents the total number of articles published within t years after the domain i appeared. $N_i(t)$ can be approximated as: $N_i(t) \equiv \sum_{t'=1}^{t} n_i(t')$, where $n_i(t')$ refers to the number of articles published in the t-th year after the domain i appeared.

Since the change range of domains' productions in the early stage is different from the recent one, we use 10-year intervals for each group to ensure that the domains in the same groups have the similar cumulative production trajectories. Thus, based on the emergence year of each domain, we analyze 584 domains and divide them into 4 groups: a) *groupA* corresponds to the domains appeared in 1970∼1979; b) *groupB* corresponds to the domains appeared in 1980∼1989; c) *groupC* corresponds to the domains appeared in 1990∼1999; d) *groupD* corresponds to the domains appeared in 2000∼2009.

To analyze the cumulative production patterns of different groups, we standardize the cumulative production trajectories and then define the standardized trajectories of the domain i as: $N_i'(t) \equiv N_i(t)/\langle n_i \rangle$, where $\langle n_i \rangle$ refers to the average annual production of domain i. Therefore, the average standardized production $ASP_{groupj}(t)$ in four groups $groupj = \{groupA, groupB, groupC, groupD\}$ is calculated as follows:

$$\begin{aligned} ASP_{groupj}(t) &\equiv \langle N_i'(t) \rangle \equiv \left\langle \frac{N_i(t)}{\langle n_i \rangle} \right\rangle \\ &\equiv \frac{1}{C_{groupj}(t)} \sum_{i=1}^{C_{groupj}(t)} \frac{N_i(t)}{\langle n_i \rangle}, \ i \in groupj, \end{aligned} \tag{3}$$

where $C_{groupj}(t)$ represents the number of domains belonging to $groupj$ in the t-th year.

For each group, we calculate its growth rate of normalized cumulative production $gr_{groupj}(t)$, where $gr_{groupj}(t)$ represents the annual growth rate for the cumulative production of the $groupj$ in the t-th year compared with its previous year [6]. This means: $gr_{groupj}(t) \equiv ASP_{groupj}(t) - ASP_{groupj}(t-1)$. Moreover, for comparing the changes and the time intervals of the growth rate in four groups, we need to use the same scale to measure, so the normalized

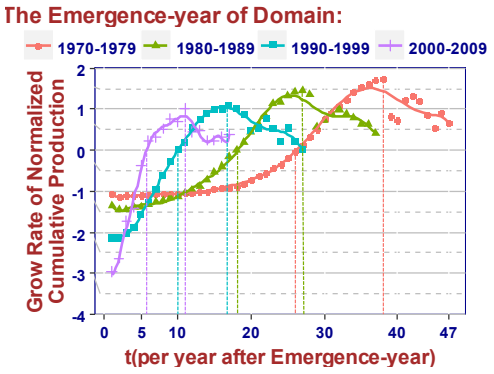


Figure 3. The Growth Rate of Cumulative Production

growth rate $Ngr_{groupj}(t)$ is calculated as follows:

$$Ngr_{groupj}(t) \equiv [gr_{groupj}(t) - \langle gr_{groupj} \rangle]/\sigma_{groupj}(gr), \tag{4}$$

where $\langle gr_{groupj} \rangle$ refers to the average growth rate for the $groupj$ in its life-cycle, $\sigma_{groupj}(gr)$ refers to the variance.

As shown in Figure 3, it can be seen that these four groups have a very similar curve shape, the main difference between these four groups is that the time needed for a domain to grow into the peak point is decreasing, meaning that the life-cycle for each domain is shortening. Therefore, it is necessary to understand the emergence of each domain. The relatively large fluctuations of each group in the latter part may be due to the existence of some unexpected events [12] or the knowledge transfer [13] between different disciplines.

Therefore, based on the $gr_{groupj}(t)$, we can further divide each group into three stages:

Definition 3 (The Initial Formation Stage). The $Ngr_{groupj}(t)$ smaller than 0 represents the $gr_{groupj}(t)$ is smaller than $\langle gr_{groupj} \rangle$, which can be defined as *The Initial Formation Stage*.

Definition 4 (The Rapid Development Stage). When the trajectory is increasing, the $Ngr_{groupj}(t)$ bigger than 0 and smaller than $max(Ngr_{groupj}(t))$ represents $gr_{groupj}(t)$ is bigger than $\langle gr_{groupj} \rangle$ and increasing to the maximum value, which can be defined as *The Rapid Development Stage*.

Definition 5 (The Steady Mature Stage). When the trajectory is decreasing, the $Ngr_{groupj}(t)$ bigger than 0 and smaller than $max(Ngr_{groupj}(t))$ represents $gr_{groupj}(t)$ is also bigger than $\langle gr_{groupj} \rangle$ and decreasing from the maximum value, which can be defined as *The Steady Mature Stage*.

B. Domain Interdependency Dynamic

To understand the domains' growth patterns, it is necessary to look into the interdependency among different domains. For each node in *DDDSVN*, the *in-degree* refers to the number of its ancestral domains and the *out-degree* refers to the number of domains which are inspired by it. Hence, based on domain-derived relations, we can calculate the

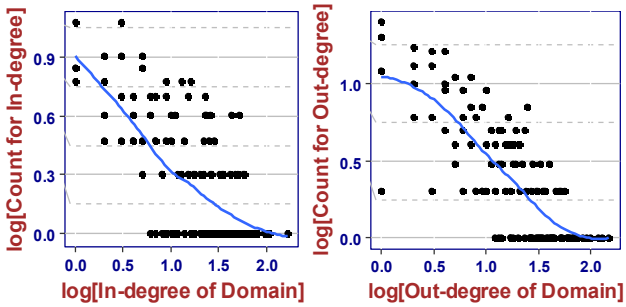


Figure 4. The Distribution of In-degree and Out-degree

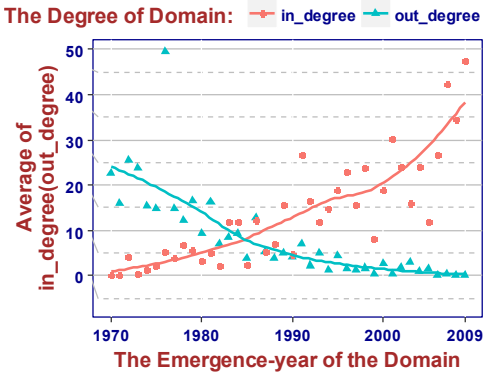


Figure 5. The Distribution of Average In-degree and Average Out-degree

value of in-degree d_i^{in} and out-degree d_i^{out} for the domain i, and then count for each different d_i^{in} and d_i^{out} in all domains, thus we can evaluate the distributions of them to understand the domain's heterogeneity in the cybersecurity discipline. As reported in Figure 4, these two distributions both show a power-law phenomenon. This suggests significant domain's heterogeneity in the *DDDSVN* and a few HUB domains exist in this discipline.

To understand the evolution for the emergence of the new domains, we calculate the average of d_i^{in} and d_i^{out} for the domains emerging in the same year. We have got d_i^{in} and d_i^{out} of each domain, so we can calculate the average of in-degree $Ave(d_t^{in})$ in t-year as follows:

$$Ave(d_t^{in}) = \frac{1}{C_t} \sum_{EY_i=t} d_i^{in}, \tag{5}$$

where C_t refers to the count for domains which emerged in t-year, EY_i refers to the emergence year of the domain i. Similarly, we can calculate the average of out-degree $Ave(d_t^{out})$ in t-year as follows:

$$Ave(d_t^{out}) = \frac{1}{C_t} \sum_{EY_i=t} d_i^{out}, \tag{6}$$

It is intuitive from Figure 5 that in-degree is increasing which means that for a new domain, it will require knowledge from more and more domains. This reveals that the interdependencies within the discipline are getting stronger. On the contrary, the out-degree is decreasing, meaning the capability of motivating new domains is reducing.

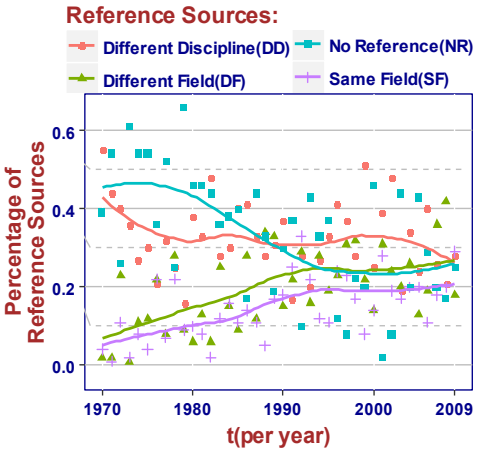


Figure 6. The Proportions of Reference Source

V. Ancestral Domain Dynamics

A. Derived Reasons

For each reference of each first article for the given domain, based on its field, we can identify four different types of *Reference Sources (RS)*: *Different Discipline (DD)* referring that the reference doesn't belong to the computer security discipline; *Different Field (DF)* referring that the reference has a different field with domain i; *Same Field (SF)* referring that the reference has the same field with domain i; *No Reference (NR)* referring that there exists no reference for the article.

Based on each type of reference sources $RS_j = \{DD, DF, SF, NR\}$, we assume that there are K references for the first articles in the domain i, for each reference $ra_k = \{ra_1, ra_2, ..., ra_K\}$, the proportion $P^i_{RS_j}$ of the reference source RS_j for the domain i is calculated as follows:

$$P^i_{RS_j} = \frac{\sum_{k=1}^{K} [ra_k \in RS_j]}{\sum_{k=1}^{K} [ra_k]}, \tag{7}$$

where $P^i_{RS_j}$ represents that the probability of the domain i is derived from RS_j. If the domains emerged in the same year, we can get the average of $P^i_{RS_j}$. Therefore, the average proportion $Ave(P^t_{RS_j})$ of the reference sources RS_j in t-year is calculated as follow:

$$Ave(P^t_{RS_j}) = \frac{1}{C_t} \sum_{EY_i=t} P^i_{RS_j}, \tag{8}$$

where C_t refers to the count for domains which emerged in t-year. By comparing the evolution of $Ave(P^t_{RS_j})$ at

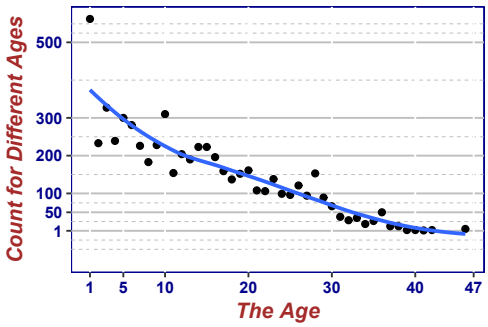


Figure 7. The Distribution of the Count for Different Ages

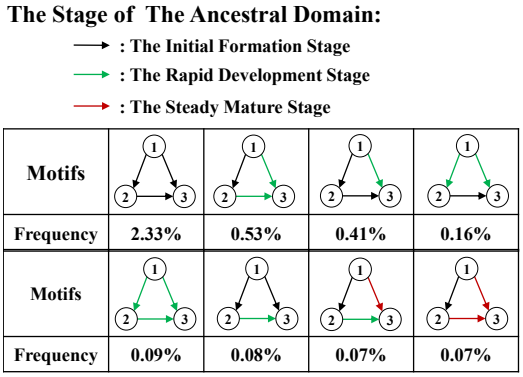


Figure 8. The Change of Motif

different years, we can observe that with the development of the discipline, the derivation reasons of domains are gradually changing.

As shown in Figure 6, the proportion of the type *Different Discipline* is decreasing in spite of its quite high value in the 70s. On the other hand, the proportion of the two types *Different Field* and *Same Field* is increasing respectively. This reveals that as time goes by, the interdisciplinary effect for the computer security disciplinary is decreasing and this disciplinary is growing mature as the endogenous power for new domains development is increasing.

B. Early Influence

1) The Age of References: For each reference which motivates the emergence of the new domains, we can calculate its age, representing the time intervals between its published year and the emergence year of the domain it belongs to. Specifically, we have got the emergence year and the first articles of the domain i, and also mapped several of the references $ra_k = \{ra_1, ra_2, ..., ra_K\}$. For these references, we can determine their published year Y_{ra_k}, as well as the domain j, which is the ra_k belongs, and then get the emergence year EY_j for the domain j. Therefore, the age of each reference Age_{ra_k} can be calculated as follows:

$$Age_{ra_k} = Y_{ra_k} - EY_j, \tag{9}$$

note that different references may have the same ages so that we count the number of different age distributions, to discuss the location distribution characteristics for all references in their belonged domains. As shown in Figure 7, intuitively, we can see that the number of different ages is decreasing as the age is increasing, that is most of these references have a related small age, meaning that the new domains are more likely to be inspired by domains which are in the early stage, instead of the relatively mature ones.

2) The Motif: The Motif is defined as recurrent and statistically significant sub-graphs or patterns may reflect a framework in which particular functions are achieved efficiently. We use *fanmod* [14], which is a tool for fast network motif detection, to detect the *"feed-forward loop"* [15] network motif in *DDDSVN*, representing the development stages of the two ancestral domains (domain 1 and domain 2 in Figure 8) for a new domain (domain 3). As shown in Figure 8, it can be seen that the domains in its initial formation stage are not only capable of inspiring new child domains but also collaborating with these child domains to motivate new domains. However, if a domain is developed while its ancestral domain is in the steady mature stage, these two domains are rarely to inspire the new one. These reveal the critical value of the newly developed domains for a discipline. One reason for this is that for each domain in our dataset, research in the early stage are more fundamental and open so that it can inspire new research directions.

VI. Related Work

A. Service Ecosystem Evolution

Large researchers concerned about service ecosystem evolution. K Huang et al. [1] proposed a rank-aggregation-based link prediction method to predict the evolution of the ecosystem. Y Zhong et al. [16] present a method to extract service evolution patterns by exploiting Latent Dirichlet Allocation (LDA) and time series prediction. Z Gao et al. [2] derive topic dependencies and describe it as a directed topic evolution graph, where four topic evolution patterns are identified. A novel methodology, named Dependency Compensated Service Co-occurrence LDA (DC-SeCo-LDA), is developed to calculate the directed dependencies between different topics, build the topic evolution graph. Thanks to the complexity and randomness of domains, the existing evolutionary analysis of service ecosystems have not involved observing the changing patterns of their systems by analyzing and predicting the derived relations among different domains yet.

B. Scientific Research

There is much research on different type networks. Wang et al. [17] derive a mechanistic model for the citation dynamics of individual articles. Sinatra et al. [7] quantify the changes in impact and productivity throughout a career in

science. Alexander M et al. [6] study the annual production of a given scientist by analyzing longitudinal career data, and develop a stochastic model as a heuristic tool to better understand the effects of long-term vs. short-term contracts. These approaches have been proposed to study the hidden meaning of the mapping relations in the network and then to analyze the data to explain the essence of the existing phenomena and the universality of the law. However, these existing methods are mostly confined only to solve the reconstruction of the existing relationship network, most of these analyses are not specific to the actual network and feature extraction.

VII. Conclusions

Understanding the emergence and growth of the new domains plays an important role in the evolution of an ecosystem. In this paper, using the scientific articles in cybersecurity discipline as the example, we present the domain-derived space by identifying the emergence of the new domains as well as the validated, significant derived relations between domains. In the context of the developed domain-derived space, the productivity analysis suggests that the growth of the new domains follows the similar patterns and it becomes difficult to generate new domains in a discipline as the interconnection among these domains is increasing. Besides, the interdisciplinary effect, meaning the influence from the other disciplines, is significant at the beginning. However, this effect is decreasing while new domains are more and more inspired by the existing domains, meaning the emergence of the new domains is transferred from the interdisciplinary-driven to endogenous impact. Finally, it shows that instead of inspired by the mature domains, the new domains tend to be motivated by those domains which are in the early stages. All these results suggest the further study to predict the emergence of the new domains.

More importantly, though this paper is based on the scientific articles, not directly to the online services nor human services due to the unavailability of the related data, this suggests the necessary to understand the emergence of the new domains within these service ecosystem and the proposed methodology will enable us to do so.

VIII. Acknowledgments

This work is supported by the National Natural Science Foundation of China grants 61502333, 61572350, 61672377.

References

[1] K. Huang, Y. Fan, W. Tan, and X. Li, "Service recommendation in an evolving ecosystem: A link prediction approach," in *IEEE International Conference on Web Services*, 2013, pp. 507–514.

[2] Z. Gao, Y. Fan, C. Wu, W. Tan, and J. Zhang, "Service recommendation from the evolution of composition patterns," in *IEEE International Conference on Services Computing*, 2017, pp. 108–115.

[3] H. K. Dam, "Predicting change impact in web service ecosystems," *International Journal of Web Information Systems*, vol. 10, no. 3, pp. 275–290, 2014.

[4] T. Petsas, A. Papadogiannakis, M. Polychronakis, E. P. Markatos, and T. Karagiannis, "Rise of the planet of the apps:a systematic study of the mobile app ecosystem," in *Conference on Internet Measurement Conference*, 2013, pp. 277–290.

[5] K. Huang, J. Yao, J. Zhang, and Z. Feng, "Human-as-a-service: Growth in human service ecosystem," in *IEEE International Conference on Services Computing*, 2016, pp. 90–97.

[6] A. M. Petersen, M. Riccaboni, H. E. Stanley, and F. Pammolli, "Persistence and uncertainty in the academic career," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 109, no. 14, pp. 5213–8, 2012.

[7] R. Sinatra, D. Wang, P. Deville, C. Song, and A. L. Barab si, "Quantifying the evolution of individual scientific impact," *Science*, vol. 354, no. 6312, pp. aaf5239–aaf5239, 2016.

[8] O. Sorenson and L. Fleming, "Science and the diffusion of knowledge," *Research Policy*, vol. 33, no. 10, pp. 1615–1634, 2004.

[9] A. Sinha, Z. Shen, Y. Song, H. Ma, D. Eide, B. J. Hsu, and K. Wang, "An overview of microsoft academic service (mas) and applications," pp. 243–246, 2015.

[10] B. Fischhoff, P. J. Weinberger, and L. I. Millett, "Foundational cybersecurity research : improving science, engineering, and institutions," 1969.

[11] C. Hidalgo, B. Klinger, A. L. Barabasi, and R. Hausmann, "The product space and its consequences for economic growth," in *APS Meeting*, 2007, p. 439446.

[12] O. Mryglod, Y. Holovatch, R. Kenna, and B. Berche, "Quantifying the evolution of a scientific topic: reaction of the academic community to the chornobyl disaster," *Scientometrics*, vol. 106, no. 3, pp. 1151–1166, 2015.

[13] R. Reagans and B. Mcevily, "Network structure and knowledge transfer: The effects of cohesion and range," in *Administrative Science Quarterly*, 2003, pp. 240–267.

[14] S. Wernicke and F. Rasche, "Fanmod: a tool for fast network motif detection," *Bioinformatics*, vol. 22, no. 9, pp. 1152–3, 2006.

[15] S. Mangan and U. Alon, "Structure and function of the feed-forward loop network motif," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 100, no. 21, pp. 11 980–11 985, 2003.

[16] Y. Zhong, Y. Fan, K. Huang, W. Tan, and J. Zhang, "Time-aware service recommendation for mashup creation in an evolving service ecosystem," in *IEEE International Conference on Web Services*, 2014, pp. 25–32.

[17] D. Wang, C. Song, and A. L. Barabsi, "Quantifying long-term scientific impact." *Science*, vol. 342, no. 6154, pp. 127–32, 2013.

CKGECS: a Chinese Knowledge Graph for Elderly Care Service

JingXuan Li, HanChuan Xu, LanShun Nie, XiaoFei Xu

School of Computer Science and Technology, Harbin Institute of Technology, China

{lijingxuan, xhc, nls, xiaofei}@hit.edu.cn

Abstract:

Developing high-quality elderly care services supporting with modern information techniques is important to solve the increasing aging problem. Knowledge base is essential to the development of elderly care services. In which, knowledge graph is the most promising knowledge representation methodology. However, there are still more challenges on construction of Chinese knowledge graph, especially for elderly care service. Thus, based on three existing knowledge graphs, we proposed a new knowledge processing and fusion method to construct the CKGECS(Chinese Knowledge Graph for Elderly Care Service). The experiments show that the proposed method can construct knowledge graph in a rapid way and the CKGECS has better quality for elderly care service.

Keywords: Chinese knowledge graph, Elderly care service, Knowledge data filter, Knowledge fusion

1 Introduction

As a result of the growing aged population, the elderly care service industry is booming. The elderly require many kinds of services such as home-based care service, community nursing service, institutional care service, and the demand is constantly expanding, the quality requirements are constantly increasing [1-3]. With the development of computer technology, the use of service computing, internet of things, cloud computing, big data and other modern information techniques are the key to realize the improvement of the elderly care services, which can provide old people with convenient, high-quality, intelligent services and meet their requirement.

On the construction of elderly care services, precise service demand perception and acquisition, accurate service recommendation, rapid service aggregation and combination are the essential techniques. In which, domain knowledge is important. Only a high-quality knowledge base can provide sufficient information and intelligent decision. There are some widely used knowledge representation and processing, such as knowledge graph, knowledge graph[4], semantic network[5], frames[6] and other typical technologies. In these techniques, knowledge graph is the most promising one which is a new type of knowledge base and derived from the semantic network. It describes various entities and concepts existing in the real world and the relationship between these entities and concepts. (subject,

predicate, object) triple has been widely used , where subject and object are entities and predicate is the relation between them. It can provide structured semantic information so that knowledge can be interpreted by computers. Thus knowledge graph for elderly care service is suitable for solving intelligent question-answer related to the entity, and it can help researcher to understand the demand of old people. Meanwhile it provide a way of searching the most precise information in a few steps, which makes knowledge graph useful in service recommendation and service aggregation.

Knowledge graph has drawn great attention from academic and industry in the past few years. The prime industry example is Knowledge Vault [7], which currently stores 18 billion facts about 570 million entities for Google's search engine. The most famous academic one is the DBpedia[8-9], which stores 1 billion facts about 4 million entities with 48293 relationships, it is a open large-scale multilingual encyclopedia. Knowledge graph can also be domain related. Unified Medical Language System (UMLS) [10] provide 135 entities 49 relationships and 6800 facts about biomedical.

Knowledge graph construction covers natural language processing, machine learning, knowledge representation, knowledge reasoning, data management and many other fields, it is a systematic project with a large workload. However, less work has been done in Chinese knowledge graph, especially for elderly care service. Because elderly care service is really a wide field. Besides, the lack of resources of Chinese language makes it more difficult to construct a Chinese domain specific knowledge graph. Even there are related knowledge, it is usually provided by an open group of volunteers, so we cannot assure the knowledge quality.

Therefore, we reuse existing resources to construct a new one after doing a survey of the elderly care service domin based on others' research and analysing the selected existing resource in great detail, which save time while obtaining high-quality Chinese knowledge graph for elderly care service(CKGECS).

2. Methodology for the construction of CKGECS

2.1 Methodology Architecture

We find that it is difficult to construct a CKGECS for the following reason:

1. Elderly care service covers many fields
2. Little available high-quality Chinese resources that we can directly use. While other resource like unstructured text need to be processed with large workload.
3. The quality is hard to improve, because related knowledge is usually provided by an open group of volunteers.

Fig 1 gives our methodology architecture to address with these problem. We clearly define the scope of our research for requirements of Elderly care service in section 3.1 to limit the field. Then we decide to reuse the existing knowledge resources CN-DBPedia[11], PKU-PIE[12], and use requirement-related resource to filter so that we can get domain knowledge. The detail is in section 3.2 3.3. Also, the professional knowledge resources TCMLS [13-14] need to be processed for using. Finally, we fuse multi-source knowledge to get a high quality knowledge graph.

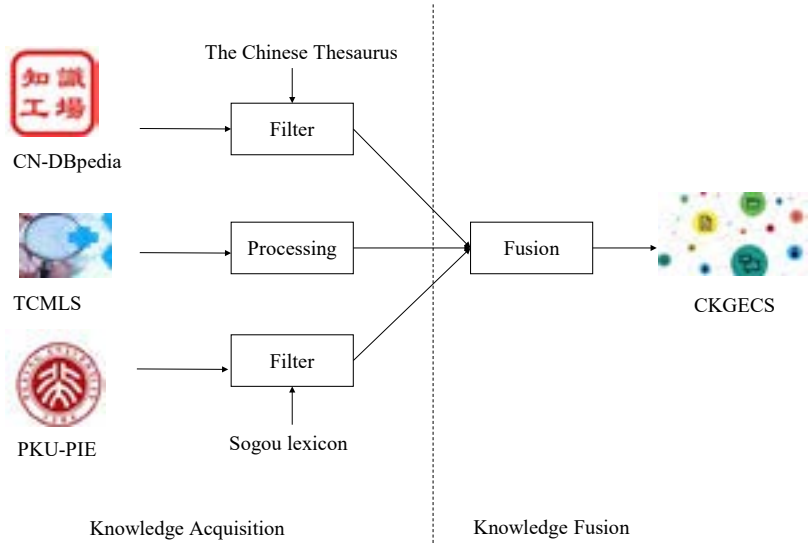


Fig.1 Methodology architecture for the construction of CKGECS

2.2 The requirements of Elderly care service

Elderly care service is an important branch of social service activities. It refers to the specialized, diversified, and multi-level services provided by country and society to meet the requirements of elderly. Western scholars have summarized the requirements of the elderly as three M, this means Money, Medicare and Mental. *Rights protection law of the People's Republic of elderly people* has also proposed the concept of five old, including a sense of security, good Medicare and education, worthiness and happiness that old people want to get. Therefore, elderly care service is a large-scale industry that covers a wide range of fields, including daily life supplies, nursing services, tourism and other related fields and industries.

Table 1 Sample statistics on elderly people's dissatisfaction with elderly care service

	Emotion	Living condition	Infrastructure	Enterta inment	Medical service	Family	Social security	Grand total
Amount	6	27	24	16	34	12	25	144
Percentage	4.2%	18.7%	16.7%	11.1%	23.6%	8.3%	17.4%	100%

Although there are many policies in China, there are already a large number of service organizations for elderly care, the quality and satisfaction of service still to be improved. Hao [15] counted dissatisfaction with several aspects of old-age care. As it has been shown in table 1. The average rate was 14.3% and the dissatisfaction with medical services, living conditions, social security, and infrastructure accounted for a relatively high proportion and is worthy of attention.

In summary, medicine, food, and tourism are core requirements of old people and need to be improved.

2.3 Existing knowledge resources

We decide to reuse the existing knowledge resources on the Internet to save constructing time. So in this part, we discuss which one we select and the resources' detail. Most of the Chinese open source knowledge graph can be obtained from OpenKG. By the end of 2018.3.17, there are already 78 related resources provided by 56 institutions. The

statistics of some open Chinese knowledge graph is in Section 2.3.By comparing all general Chinese knowledge graph, we select CN-DBPedia and PKU-PIE as our raw data, because they have large scale and are extracted by different kinds of state-of-art methods. We think they cover part of the old-age knowledge and by fusing these into new knowledge, the quality can be improved. We also can get some professional knowledge from domain related knowledge graph. We try to use TCMLS, which is constructed under the supervision of health experts and have a high accuracy.

A.CN-DBPedia

In CN-DBPedia, there are 88,454,264 facts about 10,341,196 entities with 377,912 relation types. It is difficult to check out such large amount entities, so we learn about this knowledge graph by counting the relation. It seems most of triples are not related to elderly service.Therefore, We choose to filter entity with BaiduTAG (most data have category).

There are a total of 27,833 BaiduTAG, of which only 1344 (4.8%) have a frequency more than 10, 7686 (27.6%) frequency more than 1. In List 1, there are top 20 popular BaiduTAG in CN-DBPedia. We can easily find that top 20 popular mostly unsuitable for the elderly, when bolding the elderly care service related BaiduTAG. So in this knowledge graph, a lot of information is not related and can be removed to reduce the scale.

List 1 Top 20 popular BaiduTAG in CN-DBPedia (We bold the useful part for elderly service)

➢ 文学书籍	➢ 人物	➢ **食品**	➢ 村庄
➢ 文学作品	➢ **地点**	➢ 电子产品	➢ 体育人物
➢ 小说	➢ 组织机构	➢ **中国其他行政区划**	➢ 电器
➢ 小说作品	➢ 公司	➢ 游戏	➢ **自然地理**
➢ 书籍	➢ 字词	➢ 词语	➢ **酒店**

We also analyse 3816284 triples without BaiduTAG by randomly selecting 1000 samples for 30 times. On average, 103 triples among 1000 samples are BaiduCARD, which are unprocessed natural texts with low quality. For example, （水平尺，BaiduCARD，一种长距水平尺…）. Also, about 894 samples are too professional vocabulary, such as （012Al，优点，模具使用寿命翻倍）, are not relevant to elderly care service. There are little data that are considered to be able to join the knowledge base, like（珀斯大使酒店，周围景观，西澳洲艺术馆）. Therefore, the average availability per 1000 data is only at 0.3%, which means they are not useful and not considered after this part.

B.PKU-PIE

The same as 2.3 A, we count the relation to analyse the 55,452,038 facts. Due to predicate integration, there are only 7472 relationships. Observing the relations, there is little correlation with elderly care service. We choose not to filter with relationship.

C. TCMLS

This manually constructed knowledge graph have a small size so that we can quick scan through. We also count the relation, since we know all entities are Traditional Chinese Medicine related, such as 上党人参 止血药 ,we think the relations between them are knowledge what we need.

2.4 Filter and processing

In order to get elderly care service related knowledge from 2.3 A and 2.3 B, we need to choose appropriate filter data. Besides, we should process TCMLS, which is in the owl format, to the triples.

A. Filter with CN-DBPedia

As we know about BaiduTag is hierarchical when using Baidu Baike, we should also find a hierarchical Tag Filter, which can apply to the domain of elderly care service. We only find a general one. The Chinese Thesaurus is the first large comprehensive thesaurus in China. It completed in 1975 by the China Institute of Science and Technology Information and the Beijing Library. The full thesaurus includes 108,568 topic words. There are 3707 word families, 58 primary levels, 674 secondary levels, and 1080 tertiary levels. Observe the three-level vocabulary in List 2.

List 2 An example three-level vocabulary in The Chinese Thesaurus

01． 综合经济（77 个）

10D 卫生

医院 中医 医疗 医药 药材 防疫 疾病 计划生育妇幼保健 检验 检疫

We find that there are vocabulary items such as 计划生育, which are not used so much now, so this vocabulary should be checked manually to filter tag in 3.3 A.

We finally select words in List 3 as filiter.

List 3 selected words to filter with CN-DBPedia

“酒店","景点","自然","旅游","文物古迹","公园","博物馆","特产","国家","图书馆","森林公园","自然保护区","纪念馆","教堂","药品","疾病","医学","中药","中医","医院","疗法","医疗","医药","医疗卫生","医疗保健","植物药","中草药","药物","药材","中成药","处方药","药用植物","草药","非处方药","穴位","饮食","食品","动物","植物","生物","微生物","饮料","茶","酒","细菌","软体动物","贝类动物","农作物"

B. Filter with PKU-PIE

In order to filer with entities in PKU-PIE, we should find a lexicon that contains most advanced words. Sogou provided a 12 categories of 11.27 million words in 2015.10.22. They are classified, some of which overly professional, such as video games, agriculture, forestry, livestock, engineering, and applied sciences. We only choose vocabulary for living like medicine, food, and transportation to filter PKU-PIE data. There are 660239 words related. In a triple, subject and predicate are both matched with the seed vocabulary since we avoid filter related ones.

C. TCMLS Processing

Because entities in the same concept describe the same word, so to combine, any two are synonym. We use combinatorial so that all useful knowledge can be reserved. And the relationship of P0 to P30 as well as rdf:type of each concept is also be represented with all entities in.

2.4 Knowledge fusion

Data fusion is defined as using combination of multiple sources to obtain higher quality, larger amounts of improved information. Information source usually with

1. Redundancy: Indicates that the result of representation, description or interpretation of the target by the multi-source data is the same;

2. Complementarity: refers to information is from different degrees of freedom and independence

3. Cooperativeness: different information has dependencies on other information when processing;

Therefore, data fusion means integrate the information from multiple sources, to eliminates the possible redundancy and contradiction between the information, and complements each other to obtain higher-quality, larger-quantitative improvements.

Knowledge fusion refers to the fusion of knowledge extracted from multiple data sources. The main difference from the traditional data fusion [29] task is that knowledge fusion may use multiple knowledge extraction tools while data fusion does not consider

multiple extraction. Therefore, in addition to coping with possible noise in the extracted facts, knowledge fusion introduces more noise than data fusion, which means different extraction tools may produce different results through entity linking and ontology matching.

We choose to match the same entity in 2.3 A B C to ease the influence of the different filter, so that useful knowledge is in and useless one can be removed. Then we delete highly similar triple to reduce the redundancy.

List 4 similar knowledge in CN-DBPedia and PKU-PIE

Knowledge in CN-DBPedia
新疆阿里巴巴烤大串（十字街店）,地区,哈尔滨市

Knowledge in PKU-PIE
新疆阿里巴巴烤大串（十字街店）,地区,哈尔滨

We use Levenshtein distance to measure the similarity between the two comparable knowledge listed in List 4 which are from CN-DBPedia and PKU-PIE, respectively. Levenshtein distance between two words is the minimum number of single-character edits (insertions, deletions or substitutions) required to change one word into the other, it can also be referred to as edit distance [16].

Mathematically, the Levenshtein distance between two strings a,b (of length |a| and |b| respectively) is given by $\mathrm{lea}_{a,b}(|a|,|b|)$ where

$$\mathrm{lea}_{a,b}(\mathrm{i},\mathrm{j}) = \begin{cases} \max(i,j) & if\ \min(i,j)=0 \\ \min\begin{cases} \mathrm{lea}_{a,b}(\mathrm{i}-1,\mathrm{j}) \\ \mathrm{lea}_{a,b}(\mathrm{i},\mathrm{j}-1) \\ \mathrm{lea}_{a,b}(\mathrm{i}-1,\mathrm{j}-1)+1_{(a_i\neq b_j)} \end{cases} & otherwise. \end{cases} \quad (3\text{-}1)$$

Where $1_{(a_i\neq b_j)}$ is the indicator function equal to 0 where $a_i = b_j$ and equal to 1 otherwise, and $\mathrm{lea}_{a,b}(i,\mathrm{j})$ is the distance between the first i characters of a and the first j characters of b. For example, the Levenshtein distance between ‘abc’ and ‘ab’ is 1, since only one edit (deletion) changes one into the other.

Because the Levenshtein distances don’t refer to length of string a and b, which is also useful to describe similarity, so Levenshtein ratio, is given as

Levenshtein ratio (a,b) = (|a|+|b|-Levenshtein distance(a,b) / (|a|)+|b|)

Note that in calculation of Levenshtein distance, insertion, deletion and substitution all cost one edit, while computing Levenshtein ratio substitution cost two edit and insertion, deletion still cost one. For example, Levenshtein ratio (‘abc’, ‘ab’) = (2+3-1)/ (2+3) =4/5=0.8.

After lot of tests, we choose 0.8 as threshold of Levenshtein ratio to decide whether two words is similar, so that the redundant knowledge can be removed.

3 Experiments and analysis

3.1 Evaluation indicators of knowledge base

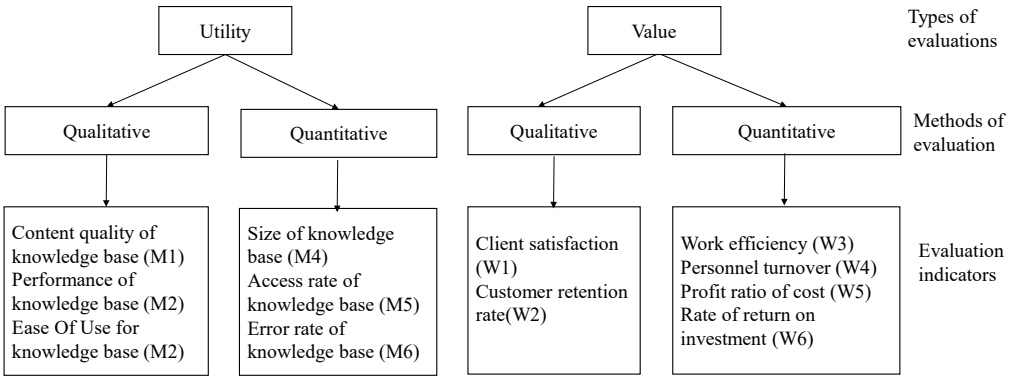


Fig.2 Framework for evaluating the construction effectiveness of knowledge base

Sun [17] proposed an evaluation framework of enterprise knowledge base construction effectiveness with universal guiding significance. The Framework is showed in Fig.2. In the evaluation of the effectiveness of our academic knowledge graph, we choose the main indicators M1, M4 and M6.

3.2. Result

To our best knowledge, there is not specific knowledge graph for elderly care service, so we choose the three knowledge graphs we used to compare with. CN-DBPedia and PKU-PIE are generic, we filtered them to only keep data which are related to elderly care served and TCMLS is for Chinese medicine. The detail is in Table 2.

Table 2 Comparison of the related work

Knowledge graph		CN-DBPedia	PKU-PIE	TCMLS	CKGECS
size	Number of entities	980,148	3,950,674	5755	650,927
	Number of relation types	69812	6499	33	56560
	Number of triples	6,462,134	**8,473,022**	104575	5,802,766
Content quality		85.9%	43.2%	**100%**	99.5%
Error rate		0.5%	**0**	**0**	**0**

We measure the size at first. CKGECS has the same size scale to the filtered CN-DBPedia and PKU-PIE, and all of them have a larger size than TCMLS.

Because there are millions of triples, it is difficult to evaluate all data manually. We calculate the content quality and error rate by random sampling method. In more detail, we randomly select 1000 samples from all triples for 100 times, and get statistical average results.

We define content quality in our research is the available percentage for elderly service. In the filtered CN-DBPedia data, the rate of unrelated professional triples is 13.6%, as shown in List 5. 57.8% of filtered PKU-PIE are unrelated, like (苏焕智,出生地,台湾), (考格兰,国籍,爱尔兰) .In CKGECS, 99.5% are available, of which is still partially professional knowledge reserved, like (津单 122,审定编号,津农种审玉 2000003).

List 5 Unrelated professional triples in the filtered CN-DBPedia data

- (S)-1-苯基-1,2,3,4-四氢异喹啉，BaiduTAG，医疗，
- (S)-1-苯基-1,2,3,4-四氢异喹啉，性状，白色或淡黄色结晶性粉末
- (S)-1-苯基-1,2,3,4-四氢异喹啉，熔点，86.0 ~ 90.0℃

We also analyse the error rates of the four knowledge graphs. For TCMLS is created by experts, we suppose there is no error in it. We randomly select 1000 samples each time, and do the selection 100 times to calculate the error rates. We don't find any error in PKU-PIE and CKGECS, so their error rates are both zero. The error rate of CN-DBPedia is 0.5%.

From the above analysis, we can draw the conclusion that for elderly care service our CKGECS has a large scale with high quality and low error rate. Besides, it is easy to construct because the knowledge reuse.

4. Conclusion

In this paper, aiming to providing high-quality Chinese knowledge graph for elderly care service, we designed a method with knowledge processing and fusion based on three existing knowledge graphs. We use The Chinese Thesaurus and Sogou lexicon to filter general graphs CN-DBPedia and PKU-PIE to keep related data, we process the TCMLS so that all useful information can be reserved. We fuse multiple sources to obtain higher quality, considerable amounts of improved knowledge.

In our future work, we will conduct more experiments to analyse our CKGECS, especially employ it into demand perception and acquisition, service recommendation, service aggregation and combination. So that we can make a further evaluation of CKGECS and validate the effectiveness of it.

References

1.Min Wu, RESEARCH ON THE DEVELOPMENT OF NURSING HOME [D], Shandong University, 2011 (in Chinese)

2.Xi Li, The Research on Demand and Supply of Endowment Service in Urban Community ——Based on the Investigation of Huaiyin District in ji'nan City [D], University of Jinan, 2012.5 (in Chinese)

3. Liya Ma, The study on the problems and countermeasures of home care services in changchun city [D], Northeast Normal University, 2013.9 (in Chinese)

4. Singhal, Amit (May 16, 2012). "Introducing the Knowledge Graph: Things, Not Strings". Google Official Blog. Retrieved September 6, 2014.

5. J. F. Sowa, "Semantic networks," Encyclopedia of Cognitive Science, 2006.

6. M. Minsky, "A framework for representing knowledge," MIT-AI Laboratory Memo 306, 1974.

7. X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang, "Knowledge Vault: A Web-scale Approach to Probabilistic Knowledge Fusion," in Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: ACM, 2014, pp. 601–610.

8. Auer S, Bizer C, Kobilarov G, et al. DBpedia: A nucleus for a Web of open Data.[c]// the semantic Web, International Semantic Web Conference, Asian Semantic Web Conference, ISWC 2007 + Aswc 2007, Busan, Korea, November. DBLP, 2007:722-735.

9.Bizer C, Lehmann J, Kobilarov G, et al. DBpedia - a crystallization Point for the Web of Data[J]. Web semantics science services and agents on the World Wide Web, 2009, 7(3):154-165.

10. O Bodenreider The Unified Medical Language System (UMLS): integrating biomedical terminology. 《Nucleic Acids Research》 , 2004 , 32 (Database issue) :267-70
11. Xu B, Xu Y, Liang J, et al. CN-DBpedia: A Never-Ending Chinese Knowledge Extraction System [J]. 2017:428-438.
12. http://openkg.cn/dataset/pku-pie
13.YU Tong, CUI Meng, LI Hai-yan, et al. Semantic Network Framework of Traditional Chinese Medicine Language System: An Upper-Level Ontology for Traditional Chinese Medicine. China Digital Medicine, 2014 9(1): 44 to 47 (in Chinese)
14. YU Tong, JIA Li-rong, LIU Jing, YANG Shuo, DONG Yan, ZHU Ling, Research Overview on Traditional Chinese Medicine Language System, Chinese Journal of Library and Information Science for Traditional Chinese Medicine, 2015 , 39 (6) :56-60 (in Chinese)
15. Wanquan Hao, The Investigation and Research on the Current Situation of Old-age Care for the Elderly in Beijing [J], Legal System and Society, 2017(24). (in Chinese)
16. https://en.wikipedia.org/wiki/Levenshtein_distance
17. SUN Pei-shan, FAN Zhi-ping, CHEN Xi, KANG Feng, Framework and Process for Evaluating the Construction Effectiveness of Knowledge Base [J], Journal of Northeastern University (Natural Science), 2010, 31(9):1361-1364. (in Chinese)

A Caching Strategy Based on Dynamic Popularity for Named Data Networking

Meiju Yu, Ru Li*

College of Computer Science, Inner Mongolia University,
Hohhot, Inner Mongolia, china,
Email: {csymj & csliru}@imu.edu.cn

Abstract: Named Data Networking (NDN) is a prominent architecture for the future Internet. In NDN, routers have the capacity of in-network cache, which can completely improve network performance. However, the cache capacity in routers is limited and how to utilize the cache resources effectively is still a great challenge. To solve the problem, this study presents a dynamic popularity caching strategy based on additive increase multiplicative decrease for NDN (DPCA). DPCA takes content popularity and caching capacity into account and it utilizes AIMD algorithm to adjust the popularity threshold dynamically. At the same time, it also proposes a evict algorithm which takes the historical information of content popularity, the trend of content request and the interval from the last request time into account. The simulation results show that the DPCA strategy can effectively improve cache hit ratio, decrease network throughput and reduce the average hit distance compared with other schemes.

Keywords: named data networking; caching replacement policy; dynamic content popularity; additive increase multiplicative decrease; evict algorithm

Biographical notes: Meiju Yu was born in 1980. She majored in computer application and earned a Master's degree from North China Electric Power University in 2006. Now she is a Ph.D. candidate and lecturer at Inner Mongolia University and the member of CCF. Her research interests include future Internet and cooperative technology in network environment, etc.

Ru Li was born in 1974. She received the Ph.D. degree in computer science from Institute of Computing Technology, Chinese Academy of Sciences in 2005. Now she is a professor and Ph.D. supervisor at Inner Mongolia University, and the senior member of CCF. Her research interests include wireless network and future Internet, etc.

*Ru Li is the corresponding author.

1 Introduction

Currently, the scale of internet users is growing rapidly and most of the user requirements are focus on the distribution and retrieval of content. The routers in IP-based network do not have the ability of in-network caching, which causes repeated transmission of large amount of content, wastes bandwidth resources and increases accessing delay. Jacobson et al.(2009) and Zhang et al.(2010) propose Named Data Networking (NDN) as a promising architecture based on contents for future network. The

* This research is supported by the CERNET Innovation Project (NGII20161205) and the Research Project of Higher Education School of Inner Mongolia Autonomous Region under Grant NJZY18010.

routers in NDN can cache the contents passing by because they have the availability of named routing and in-networking caching.

Caching strategy is the key research area of NDN, which greatly affect the performance of network. NDN cache takes on several new characteristics which pose a lot of new challenges to NDN caching technologies(Zhang et al., 2013). In order to reduce the amount of redundant data and efficiently utilize caching resource, we propose a dynamic popularity caching strategy based on AIMD which is named DPCA for Named Data Networking to improve the overall performance of NDN in this paper. DPCA takes both content popularity and caching capacity into account.

The reminder of this paper is organized as follows. In section 2, we present the related work. In section 3, we illustrate the DPCA strategy in detail and show the simulation environment and the performances of our cache policy in Section 4 subsequently. Finally, we conclude the paper in section 5.

2 Related Work

Many researchers have proposed various solutions to manage the in-networking caches in order to improve the performance of network(Zhang et al., 2015; Abdullahi et al., 2015). The default in-network caching strategy named Leave Copies Everywhere (LCE) caches all the contents in all on-path routers in the data reply path which brings a huge number of redundant data , the higher number of content replacements and lowly local resource utilization on the routers (Jacobson et al., 2009).

Hou et al. (2018) propose a sum-up Bloom-filter-based request node collaboration caching (BRCC) approach which caches the frequently requested content close to the request node. Ju and Lim (2017) present a cache sharing using bloom filters in named data networking(DPCP CC) which defines a summary packet based on a Bloom filter and proposes a method to share the summary with neighboring routers .

The research of caching strategy based on content popularity attracts people's attention. Bernardini et al. (2013) present a novel caching strategy, named Most Popular Content (MPC), which only caches popular content whose popularity exceeds a fixed threshold. Unlike MPC, a new caching policy, named Fine-Grained Popularity-based Caching (FGPC), is proposed which always caches coming content when content store is available (Ong et al., 2014). When the content store becomes full, it will keep only most popular content. Bohao Feng et al. (2015) proposed a cache permission policy defined as Cache-Filter, which can cache popular contents closer to users.

3 DPCA: Algorithm Design

A. Request Records

In DPCP, we use a new table named Request Record Table (RRT) which is maintained by the routers to store the information of requests. When an interest packet i arrives at the time t_j , the router will record (content name, $N_i, R_i(t), t_j$) into RRT. Where N_i is the number of requests in last period and $R_i(t)$ is the number of requests at time t in current period.

B. *Content Popularity*

When content i arrives, the popularity of content i is defined as follows:

$$P_i = \alpha_1 \times N_i + \alpha_2 \times R_i(t) \quad (1)$$

Where α_1 and α_2 are the weight coefficients and $\alpha_1 + \alpha_2 = 1$.

C. *DPCA Caching Permission Strategy*

In DPCA, when a data packet arrives, the popularity of the incoming data will be calculated according to formula (1) and compare the value with the popularity threshold. When the popularity of arriving data exceeds the threshold, it will be cached and forwarded. Otherwise, the arriving data will be only forwarded without caching.

The core algorithm of DPCA is AIMD which is used to control congestion in network. We employ this algorithm to dynamically adjust the popularity threshold, which consists of three main parts.

(1) When DPCA begins to start, the content popularity is small because of fewer requests. So all the content passing by will be cached to ensure efficient utilization of caching resources.

(2) When the caching space is full, AIMD assigns the average value of content popularity in CS to the popularity threshold as the initial value. If the popularity of a arriving data packet exceeds the popularity threshold, caching permission strategy will decide to cache the content and popularity threshold will add 1.

(3) When cache replacement has not occurred for many times, AIMD will reduce the popularity threshold to half of the original value.

D. *Caching Evict Algorithm*

When the caching permission strategy decide to cache a data packet and the caching space is full, a content will be evicted form CS. A caching Evict function is presented as follows:

$$evict_i(t) = \alpha \frac{P_i}{\sum_{k=1}^{n} P_k} + \beta \frac{R_i(t) - \overline{R}}{\sum_{k=1}^{n} \left| R_k - \overline{R} \right|} + \gamma \frac{\max_{1 \le k \le n} \{t - t_k\} - (t - t_i)}{\max_{1 \le k \le n} \{t - t_k\}} \quad (2)$$

$$\overline{R} = \left(\sum_{k=1}^{n} R_k(t) \right) / n \quad (3)$$

α, β, γ are the weight coefficients and $\alpha + \beta + \gamma = 1$.

Figure1 shows the modules of a router utilizing the DPCA caching strategy in detail.

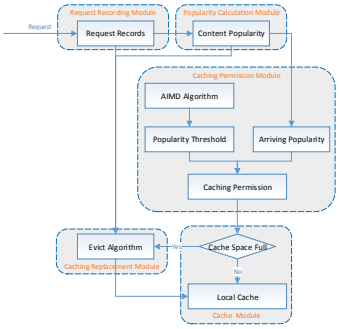


Fig. 1. The Modules of DPCA in a router

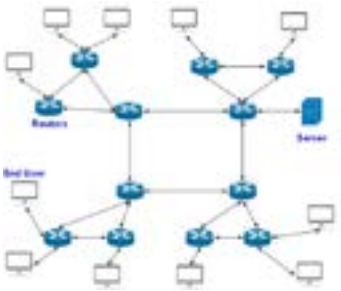


Fig. 2. Network Topolog

4 Experiment Evaluation

In this section, the simulation results of DPCA caching strategy are presented, compared and analyzed using ndnSIM (Mastorakis et al., 2017). Because of the limited caching space, we hope to observe the impact on the caching performance with the changes of cache size.

A . Simulation parameter setting

In all our simulations, we choose LRU (Least Recently Used) and FIFO (First In First Out) caching strategy as comparison strategies with our DPCA strategy and record experimental results in 200s. The network topology is depicted as Figure2 and the simulation configurable parameters are depicted as Table I.

TABLE I. SIMULATION PARAMETER SETTING

Parameter	Description	Value
n	Number of contents	200
Request Rate	Number of requests of user	50 req/s
Cache Size	Number of contents stored per cache	10,20,40,60,80,100
τ	Period time	3s
α	MZipf exponent parameter	1
q	MZipf plateau parameter	0

B . Experiment results and analysis

1) Cache Hit Ratio

Figure 3 is a comparison of the cache hit ratio of the DPCA，LUR and FIFO. The cache hit rate of DPCA strategy is always higher than that of LRU and FIFO, because DPCA caches the popular content whose popularity exceeds its dynamic popularity threshold. Especially when the cache size is the smallest one, the contrast is obvious. Because in DPCA, the routers have only cached contents whose popularity exceeds the popularity threshold in CS. While LRU and FIFO frequently replace content in routers without considering the content popularity. Therefore, when the cache size is small, the cache hit ratio of DPCA is enhanced by 26% compared with FIFO. However, with the increasing of the cache capacity of the routers, routers will better server the request from users and the difference of the cache hit ratio becomes smaller.

2)Network Throughput

Figure 4 is a comparison of the throughput in NDN network between the DPCA, LRU and FIFO cache strategy. The throughput in DPCA is the lowest between the three strategies while it has the highest cache hit ratio. That is, when the cache size is 10, it decreases the network throughput by 15% and has achieved a 26% improvement at cache hit ratio. The results in Figure4 show that the throughput of DPCA are always lower than the others.

5 Conclusion

In this paper, we proposed the DPCA caching strategy for named data networking. In DPCA, if the content popularity of the arriving data exceeds the popularity threshold, then the data will be cached in CS. We employs an AIMD algorithm to dynamically adjust the content popularity threshold and also present a cache evict algorithm which

take full advantage of various factors. The simulation results show that the DPCA strategy shows better performance compared to FIFO and LRU.

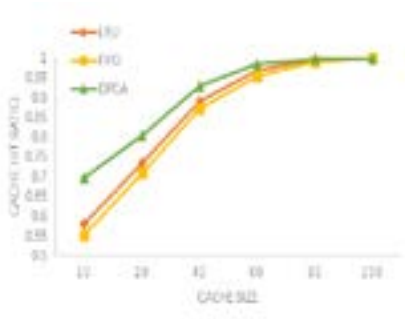


Fig. 3. Cache Hit Ratio

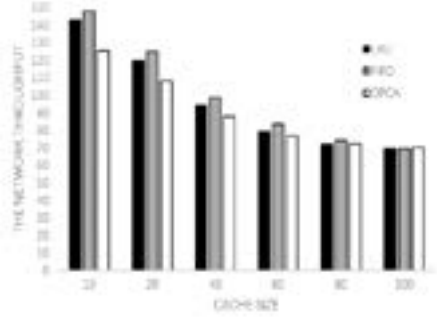


Fig. 4. The network throughput

References

Abdullahi, I., Arif, S. and Hassan, S. (2015) 'Survey on caching approaches in Information Centric Networking.' Journal of Network & Computer Applications 56.11:48-59.

Bernardini, C., Silverston, T. and Festor, O. (2013) 'Mpc: Popularity-based caching strategy for content centric networks'[C]//Communications (ICC), IEEE International Conference on. IEEE, pp.3619-3623, 2013

Feng, B. Zhou, H. Zhang, M. and Zhang, H.(2015) 'Cache-Filter : A Cache Permission Policy for Information-Centric Networking.' Ksii Transactions on Internet & Information Systems pp.4912-4933.

Hou, R., Zhang, L., Wu, T., Mao, T. and Luo, J.(2018)'Bloom-filter-based request node collaboration caching for named data networking.' Cluster Computing:1-12.

Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H. and Braynard, R. L.(2009) 'Networking named content,' in Proceedings of the 5th international conference on Emerging networking experiments and technologies, pp.1-12.

Ju, HM.and Lim., H.(2017) 'Cache sharing using bloom filters in named data networking.' Journal of Network & Computer Applications90:74-82.

Mastorakis, S., Afanasyev, A. and Zhang., L.(2017) 'On the Evolution of ndnSIM: an Open-Source Simulator for NDN Experimentation.' Acm Sigcomm Computer Communication Review 47.3:19-33.

Ong , M. D., Chen, M., Taleb,T., Wang, X. and Leung, V. C. M. (2014) 'FGPC: fine-grained popularity-based caching design for content centric networking'[C]//Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems. ACM, pp.295-302.

Yang, Y. and Zhu, J. (2016) 'Write Skew and Zipf Distribution: Evidence and Implications' [M]. ACM.

Zhang, L., Estrin, D., Burke, J., Jacobson, V., Thornton, J. D., & Smetters, D. K., et al. (2010). Named data networking (ndn) project. http://www.named-data.org/ndn-proj.pdf, 1892(1), 227-234.

Zhang, G., Li, Y. and Lin, T.(2013) 'Caching in information centric networking: A survey.' Computer Networks 57.16:3128-3141.

Zhang, M., Luo, H., and Zhang, H. (2015) 'A Survey of Caching Mechanisms in Information-Centric Networking.' IEEE Communications Surveys & Tutorials 17.3:1473-1499.

Seq2seq Neural Networks based Big Data Logs Predictive Analysis

Pin Wu, Quan Zhou, Zhidan Lei, Xiaoqiang Li

School of Computer Engineering and Science,
Shanghai University,
Shanghai, China
E-mail: jzhou8763@shu.edu.cn

Abstract: While bigdata processes high-volume data at high speed, it also generates a large amount of logs. However, people hardly went to smartly predict the most likely future events based on massive, multi-source, heterogeneous bigdata logs. This paper proposes a comprehensive method for smart computing and prediction of bigdata logs. The previous method to predict the accuracy of data based on time series is not good enough. In this work we firstly elaborate the use of distributed ways to collect and store bigdata logs, event-location and vectorized representations of logs. Then, we present log fusion algorithm to convert the logs (unstructured text data) of each component of bigdata into structured data by removing noise data, adding timestamps and classification labels. Finally, we introduce a predictive model which improves sequence-to-sequence algorithms by novel tricks of attention mechanism to balance deviations in the sequence and adjustor to globally fit the data distribution of output sequences with input sequences. Our experimental results show that the neural network model trained by our method has a good performance with the real-word data. Compared with the previous predictive method, the RMSE is reduced by 46.65% and the R2 fitting degree is improved by 14.28%.

Keywords: Service Computing, Smart Data Processs, Bigdata, Recurrent Neural Network, Log Analysis

Biographical notes: Pin Wu, associate professor at Shanghai University

Quan Zhou, graduate student at Shanghai University, master's degree at Fudan University.

Zhidan Lei,Shanghai University graduate student.

Xiaoqiang Li, associate researcher at Shanghai University, Ph.D. at Fudan University.

1 Introduction

In recent years, the explosion of information has led to the rapid development of bigdata. Bigdata provides people with fast computing and mass storage through parallel computing and distributed storage. In the process of providing bigdata services, a huge amount of log data is generated. However, these logs are bigdata by themselves. And because these logs are usually the only data that can be used to record the behavior of bigdata systems at

runtime, it is necessary to monitor and analyze these logs to better protect the functioning of bigdata services. Analyzing and processing these massive logs has gradually become an indispensable requirement for securing bigdata services. However, different bigdata systems often use different components to accomplish different tasks. These components include Hadoop HDFS, MapReduce, Hbase, Hive, Spark, Redis, Kafka, MPP, Zookeeper, YARN and others. These components generate their own log respectively during operation. Not only are these logs huge, but also they are independent and their data structures are different. In other words, bigdata system log has notable features, such as massive, multi-source, heterogeneous. These features of bigdata, system have caused many difficulties in the task of monitoring and analyzing the log. People are always trying hard to find the relevant method. Although great progress has been made, there is still room for improvement in terms of complexity and accuracy. With the development of neural networks in recent years, people began to try to use neural networks to mine and explore such data structures. This paper proposes a new method which use the recurrent neural network of sequence to sequence to learn the log of bigdata and generate the prediction model. After bigdata system real-time logs generated from different components were input to the model, we can do amazing data analysis such as predicting the possible problems of other components, generating the most possible log sequence and so on. Once smart and reliable log analysis methods are available, managers can better adjust node plans for bigdata systems.

2 Related Work

Time series is a very common data structure and people have always been very interested in studying it. Logs are generated chronologically and are a typical time series data structure. People began to analyze and predict them by using time-series-based algorithms. People use models like HMM , ARIMA , ANN in this case. However, HMM does not perform well when there is not enough prior knowledge about the given series. The basic idea of the ARIMA model is that the data sequence formed by the predicted object over time is considered as a random sequence, and a certain mathematical model is used to approximately describe the sequence. Once this model is identified, future values can be predicted from the past and present values of the time series. Modern statistical methods and econometric models have been able to help us to predict the future to some extent. Recurrent neural networks (RNNs) originated from Hopfield Networks proposed by Saratha Sathasivam . Since 1986, the network was replaced by fully connected neural networks and some traditional machine learning algorithms. However, the method of fully connected neural network requires massive parameters which greatly increase the computational load of the computer, and even worse, it can not acquire the time series information in the data. The appearance of recurrent neural networks can be very good for processing and predicting sequence data. In 1997, Sepp Hochreiter and Jurgen Schmidhuber proposed that Long Short Term Memory (LSTM) solves the problem of gradient disappearance of the earliest recurrent neural networks. Since then, people have been making various changes to LSTM to adapt to all kinds of tasks . This article is also based on the algorithm designed to predict the massive, multi-source, heterogeneous logs.Although DNNs work well whenever large labeled training sets are available, they cannot be used to map sequences to sequences. Le et al. present a general end-to-end approach to sequence learning that makes minimal assumptions on the sequence structure. In 2014, K Cho et al. propose a novel neural network model called RNN Encoder-Decoder that consists of two recurrent neural networks (RNN)

to improve the performance of a statistical machine translation system. Then, Dzmitry Bahdanau et al. propose to allow a model to automatically (soft-)search for parts of a source sentence that are relevant to predicting a target word, without encoding a source sentence into a fixed-length vector. People are constantly applying seq2seq to different fields and have achieved very good results. On this basis, we also apply seq2seq to the analysis of bigdata logs.

3 Bigdata Logs Predictive Analysis Model

3.1 Sequence Model Definition

Let l_1,l_2,l_3 ,..., l_n be the time series data with history window H and predicting window P.

3.2 Predictive Model Definition

In order to simplify the calculation, we assume that the probability of the current log only relevant to the previous n-1 log events. Once training is complete, we produce predictions by getting the most likely sequences according to our model.

3.3 Training And Predictive Model Framework

The inputs and outputs of training and predictive models are all sequences of vectors. We call these vectors log2vectors.Encoding module and decoding module is mainly composed of neural units, shown in Figure **??**. Let x_t be the input at time t and s_t in equation **??** be the state information, which can also be understood as the network memory. y_t is the output at time t, U, V and W are the three weight matrixes shared by the whole network.

3.4 Attention Mechanism

We will loss part of the information in the encoding stage. Once the long-term sequence information is lost, it will seriously affect the decoding result. The more recent logs tend to more influence the hidden state vector. Therefore, we introduce attention mechanism to solve this problem. Attention mechanism is proposed in computer vision, that is, high-resolution focus on a specific area of the picture and low-resolution perception of the image of the surrounding area mode . In other words, we should pay more attention to the important logs. On the contrary, we give less attention for unimportant logs. Recurrent sequence generators have recently achieved very good results in a variety of tasks including machine translation, handwriting and image title generation . So, we borrowed this idea to promote the sequence-to-sequence model.

3.5 History Window And Predicting Window

The history window is the number of events that occurs from one time past to the present. The predicting window is the number of events that happen from the future time to the current moment.

3.6 Adjustment Factor Definition

Our model not only predicts the output of each log of the big data log, but also predicts the statistics of certain types of logs, such as error logs, traffic, task volume, and so on. In predicting the latter task, we find that the variance of the data predicted by the RNN network is often smaller than the variance of the true value. Therefore, in order to better fit the true data, we introduce a adjustor, a process is added to adjust the output of the decoding module. This adjustor spatially converts the distribution of the predicting data.

3.7 Log Pattern Definition

The log pattern S is a finite set $D_1, D_2, ..., D_n$, where D_i (i = 1, ..., n) becomes an attribute domain. The most common log pattern is CaseID, message, @version, @timestamp, host, which specifies the metadata describing the log, where CaseID, message, etc. are the attribute fields.

4 Our Method

4.1 Our Method Architecture

The data flow is from left to right, as shown in Figure 5. First, we collect logs of individual components from the bigdata system. Second, the logs from these components are standardized according to the rules, that is, unstructured data is converted into structured data. Then, considering that massive logs require a distributed storage space, the normalized logs are then stored in a distributed full text indexing system named Elasticsearch . In this process we achieve a large number of heterogeneous log normalization and facilitate the massive log data query, analysis and processing. Again, a training set or prediction log sequence data is acquired from a full-text search database based on a sequence-to-sequence training and prediction model. Finally, when the data is ready, start training the model or input the trained model to make predictions. In the next section, we will highlight the framework and implementation steps for the training and predicting model.

4.2 Model Framework And Implementation Steps

The pipeline of the training and predicting model we designed is shown in Figure 6. Specific steps are as follows. Step 1: Collect Log Data Step 2: Classify Logs Step 3: Vectorize Logs Step 4: Diversify training set Step 5: Encode Log Vector Step 6: Decode Hidden State Vector Step 7: Get Target Information

4.3 Log Fusion Algorithm

In the log normalization process, the first task is to distinguish between the event boundaries in the log text stream, because the logs output by various components of the bigdata system often have their own format, and the combination of multiple lines of log information is an event. We set some rules by regular expression to normalize the logs based on the boundary of the event(see Sec 4.3). Obviously, the study of time-series data necessarily makes it very important to record timestamps (logs with event granularity). Note that we have to customize different filtering configuration file for different bigdata components.

4.4 Training And Predicting Algorithms

Our algorithm is mainly based on the seq2seq pattern of the recurrent neural network. This algorithm consists of two options, one is the training phase and the other is the predicting phase. In the training option, time series based bigdata logs changed into vector are input into the model to train a predictive model. In this phase, the neural network model can obtain weights, get the context information and relationships among the logs. Thus it havs the ability to predict a future log sequence given a log sequence. In the predictive option, we can input the real-time generated log into the trained model to obtain the predictive target data. The algorithm in Figure 8 describes our specific approach.

5 Experiment

5.1 Experimental Data and Environment

Our experimental data is a set of log files of various components of Hadoop's bigdata system provided by Beyondsoft (Shanghai) Co., Ltd. The bigdata system has a total of 32 nodes, installed HDFS, HBase, Hive, Spark, Solr, YARN and other components. We collected 2017 full year log data into Elasticsearch and trained and predicted on a machine equipped with Nivida GTX1080 GPU graphics. The total raw log data size is around 1.08TB from Namenode, Datanode, Resourcemanager, Hbase.

5.2 Experimental Tasks and Evaluation Indicators

This article uses several indicators to evaluate the proposed algorithm model. These indicators have root mean square error (RMSE), extended variance score (ESV), mean absolute error (MEANAE), mean absolute error (MEDIANAE), R2 SCORE, the standard deviation of the prediction data and the standard deviation of the real data.

5.3 Experimental Results

We trained and predicted "Hyren Bigdata Platform" error logs that manages the hadoop bigdata system. Those log data was selected 80% as the training set, 20% as the test set. The history window is set to 10 and the predicting window is set to 1. As shown in Figure, the data predicted by the proposed method has a very good fit both for the training data (R2 approx. 1) and the test data (R2 approx. 0.94).

6 Conclusion

In this paper, based on the massive, multi-source and heterogeneous characteristics of log of many components of bigdata platform, a method of predictive analysis is proposed. This shows that the method proposed in this paper has better prediction accuracy. Today, it is already the era of bigdata and artificial intelligence. By combining the methods proposed by bigdata and artificial intelligence algorithms, this paper focus on bigdata log predictive analysis to better analyze the past, present and future of bigdata¡¯s status.

References

User-Oriented and Decentralized Data Integrity Audit Scheme for Cloud Service

Yanan Jiang[1,3], Zhiyong Feng[1,3], Shizhan Chen[1,2], Keman Huang*[,4]
[1]*Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin 300072, China*
[2]*School of Computer Science and Technology, Tianjin University, Tianjin 300072, China*
[3]*School of Computer Software, Tianjin University, Tianjin 300072, China*
[4]*MIT Sloan School of Management, Cambridge, USA*
{*brucejiang, zyfeng, shizhan*}*@tju.edu.cn*, {*keman*}*@mit.edu*

***Abstract*—Could Computing is a long dreamed vision of computing as a utility, where users could remotely store their data into Cloud so as to enjoy the on-demand high-quality services from a shared pool of configurable computing devices. By data outsourcing, Users could be relieved from the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the possibly large scale of outsourced data makes data integrity protection in Cloud Computing an extremely challenging and potentially arduous task, especially for users with constrained computing resources and capabilities. Thus, enabling public auditability for could data storage security is of critical importance so that users could resort to an external audit party or themselves to verify the integrity of outsourced data when needed. There are available schemes of data integrity verification with public auditability which could not avoid the Third Party Auditor(TPA) or the Could Service Provider(CSP) itself. However, in a large-scale and dynamic environment, the reliability of the TPA-based scheme is far from satisfaction as well as the CSP-based scheme. To securely introduce an effective scheme, the following two fundamental requirements have to be met: 1) the scheme should be able efficiently to audit the cloud data storage without demanding the local copy of data; 2) the scheme should be more reliable rather than relying on TPA and CSP. In this paper, we utilize and uniquely combine the Blockchain and some knowledge of cryptography, such as homomorphic encryption, zero-knowledge proof and so on, to achieve the user-oriented decentralized data integrity auditing system, which meets all above requirements. The scheme could provide the user with his own execution and storage space, and at the same time, it could protect user's data from disclosure with an encrypted transmission. More importantly, the trustful network makes the result more reliable. Extensive security and performance analysis show the proposed scheme is provably secure and highly efficient.**

***Keywords*-BlockChain, Decentralization, User-Oriented, Data-Integrity Audit, Public Auditability, Cloud Service**

I. INTRODUCTION

Could Service has been envisioned as the next-generation architecture of IT enterprises and organizations. As a disruptive technology with profound implications, Cloud Computing is transforming the nature of how business use information technology. One fundamental aspect of this paradigm shifting is that data will be outsourced to the third-party Cloud Service Provider(CSP), which will improve the storage limitation of local devices. Recently, a couple of commercial cloud storage services, such as the sample storage service–Amazon S3[1] on-line data backup services of Amazon, Azure Blob[2] blob storage service of Microsoft and Object Storage Service(OOS)[3] of Aliyun, and also some practical cloud-based software Google Drive[4], Dropbox[5], Bitcasa[6] and so on, which have been built for cloud application.

Cloud Server is not trustworthy in nature. Since Cloud Service Providers(SCPs) are separate administrative entities, data outsourcing is actually relinquishing user's ultimate control over the fate of their data. As a result, the correctness of the data in the cloud is being put at risk due to the following reasons. For the sake of profit, Cloud Server may still persuade user that he owns the original data stored by the user, even if the data may be completely or partially lost. Cloud Server misconduct is varied, like repossessing storage space by maliciously discarding data that is not accessed infrequently and hiding lost events due to administrative errors, hardware failures, external or internal attacks, etc. On the other hand, if the user wants to know whether their cloud storage data is secure or not, they either blindly believe in cloud storage service providers or are time-consuming and laborious to retrieve all out stored data in Cloud Server and verify whether it is original data or not. In the latter case, the user is likely to have almost the same copy of the data locally as the cloud storage service provider. Since the cloud server may return invalid results, new forms of data integrity assurance are required to protect the security and privacy of cloud users' data.

To overcome the above critical security challenge of todays cloud storage services, apart from internal auditing model in Cloud Service which is called the CSP-based scheme, the most mainstream solution is the TPA-based scheme. When talking about the former, the auditing process for this solution is visible only within the cloud service provider, and it is only auditing results that will be disclosed externally. Therefore, unless the user blindly trusts Cloud Service, the result given by Cloud Server is not credible. As for the latter, TPA in this architecture is the core module. Mostly traditional solutions[7], [8], [9], [10] proposed in recent years are based on this foundational architecture in which centralized TPA plays a primary role when validating

data integrity. Under this model, TPA undertakes main auditing task. On the one hand, when meeting massive-scale data, TPA may have an efficiency problem because of lots of non-automatic procedure, such as manual review, multi-step auditing process and so on. When using manual review, privacy data of users may be disclosed to the reviewer, which is unsafe. So, it not easy for the given result to convince the user unless the user fully trusts TPA.

Therefore, we need some new solutions to solve trust issues. In this paper, the problems proposed above motivate users to explore a more trust and safe scheme, while achieving efficient data integrity auditing. In the end, we propose a construction which not only supports user data privacy protection during the data auditing processing but also makes an efficient, safe, secure and reliable procedure and result. Our idea is to apply some knowledge of cryptography, such as homomorphic encryption[15], zero knowledge proof[14] and Data Digest over information exchange process. More importantly, we use Blockchain[12] to ensure the execution of the information exchange process and preserves the verification results because we could make a decentralized and reliable network through Blockchain basic protocols. What's more, it's so convenient for us to build an user-oriented auditing model, which means each user has its own user space, with Smart Contract [13], [12].

A. Contribution

In this paper, we further study the problem of how to make a reliable, safe, and efficient data integrity auditing and the principle and application of Blockchain. Our contributions are three folds as follows:

1) provide a new model for data integrity auditing which is user oriented, decentralized, much more reliable and safe.
2) analysis the correctness of our model and give its poof, and introduce the prototype system model.
3) prove the security and justify the performance of our proposed scheme through concrete experiment system.

B. Paper Organization

The rest of the paper is organized as follows: Section II clarifies the theoretical model of our solution. Section III presents the basic system model. Section IV is the performance evaluation of the prototype system. Section V surveys related work and Section VI concludes this paper.

II. Theoretical Model

In this section, we will mainly introduce the basic theoretical model used by our solution, including the basic symbol definitions, the data exchange protocol.

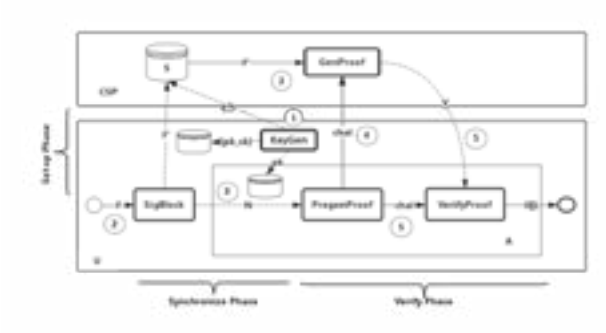


Figure 1. An Auditing System from Workflow

A. Preliminaries

User(U) wants to store a piece of data F, which often could be regarded as a file, into Cloud Server(S) and performs data integrity auditing through Auditing System(A). We denote the output x of an algorithm $\mathcal{A}$ by $o \leftarrow \wr$. We denote by $|o|$ the absolute value of o.

- F - a piece of outsourced data which could be denoted as a finite ordered collection of n blocks: $m_1, ..., m_n \in \mathbb{Z}_\wr$.
- $f_{key}(\cdot)$ - pseudo random function(PRF), defined as: $\{0,1\}^* \times key \rightarrow \mathbb{Z}_\wr$.
- $\pi_{key}(\cdot)$ - pseudo random permutation(PRP), defined as: $\{0,1\}^{log_2(n)} \times key \rightarrow \{0,1\}^{log_2(n)}$.
- $H(\cdot)$ - secure hash function, defined as: $\{0,1\}^* \rightarrow G$, where G is a multiplication cyclic group. It maps a string uniformly onto G.
- $h(\cdot)$ - secure hash function, defined as: $G \rightarrow \mathbb{Z}_\wr$ mapping the group elements of G uniformly onto $\mathbb{Z}_\wr$.

1) Bilinear Map: LetG_1, G_2 be additive groups and G_T a multiplicative group, all of prime order p(p is the RSA[11] module). Let $g_1 \in G_1$, $g_2 \in G_2$ be generators of G_1 and G_2, respectively. A pair is a map[11] e is a mapping from $G_1 \times G_2$ to G_T, and denoted as $e : G_1 \times G_2 \rightarrow G_T$, and satisfies the following properties: (1) Bilinearity: let $g_1 \in G_1, g_2 \in G_2, a \in \mathbb{Z}_\wr, b \in \mathbb{Z}_\wr$, such that $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$; (2) Non-degenerate: $\forall g_1 \in G_1/\{1\}$, $\exists g_2 \in G2$ such that $e(g_1, g_2) \neq 1$; (3) effective computability: for practical purposes, e has to be computable in an efficient manner.

2) Homomorphism Verifiable Digest(HVD): Given a message m (corresponding to a piece of data), we use H_m to represent its HVD. The data digest will be stored with the data segment F in the cloud storage service S. The HVD is metadata that is block validated, in addition to being unforgeable, and it will have the following attributes:

- *Idempotence*: Given two HVDs $H_{m_i} = g^{m_i} h^{r_i}$ and $H_{m_j} = g^{m_j} h^{r_j}$ of messages $m_1, m_2 \in \{0, 1, 2, ..., T\}$ respectively, anyone can create a uniformly distributed encryption of $m_1 + m_2 (mod\ N)$ by computing the product $H = H_{m_i} H_{m_j} h^r$. For $\forall r \in \{1, 2, 3, ..., N - 1\}$, because $C_1 C_2 h^r = (g^{m_1} h^{r_1})(g^{m_2} h^{r_2}) h^r =$

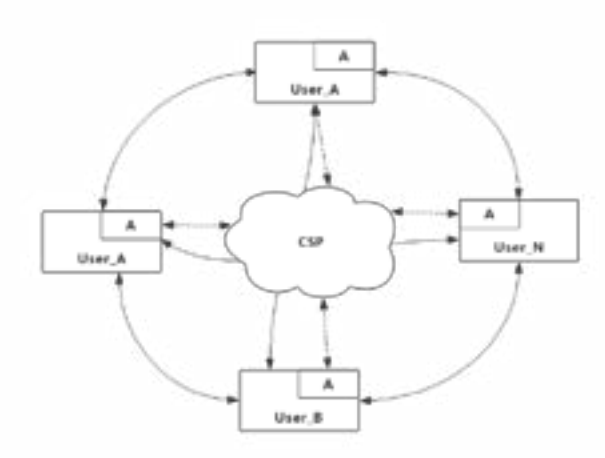


Figure 2. Multi-Node Model(MNM)

$g^{m_1+m_2}h^{r_1+r_2+r}$ is an encryption of $m_1 + m_2$.

- *Blockless Verification*: Using HVDs, the cloud server can construct a proof that allows the auditing system to verify whether the could server possesses certain file blocks, even when the auditing system does not have access to the actual file blocks.

B. Basic Protocol

First of all, we define a set of basic protocols based on [18] called *BP*, and it is a collection of five polynomial-time algorithms *(GenKey, SigBlock, PregenProof, GenProof, VerifyProof)*, such that:

- $KeyGen(\lambda) \rightarrow (pk, sk)$ is a key pair generated algorithm run by *A* to produce an account for *U*. It takes no parameter as input, and return a pair of public key pk and secret key sk.
- $SigBlock(pk, F) \rightarrow (\Phi)$ is an algorithm run by *A* to generate the HVD corresponding to each data block of F. It takes a public key pk, a piece data sequence F, and returns the collection of HVDs, Φ.
- $PregenProof(N) \rightarrow (chal)$ is an algorithm run by *A* to construct a random array $chal = \{(i, v_i)|i \in I\}$. It takes a value N which is the number of blocks for F as inputs, and returns $chal = \{(i, v_i)|i \in I\}$.
- $GenProof(pk, chal, F) \rightarrow (V)$ is an algorithm run by *S* to generate an auditing proof of data possession and integrity for *A*. It takes a random index array $chal$, a public key pk and a collection of data blocks F as inputs, and then returns an auditing proof $V = (\delta, \sigma, R)$.
- $VerifyProof(pk, chal, V) \rightarrow (\{0|1\})$ is an algorithm run by *A* to verify whether the result is $\{0|1\}$. It takes a public key pk, a random verifying array $chal$, and the auditing result V as inputs, then returns $\{0|1\}$.

It's the basic protocol that an Auditing System can be constructed from. It can be divided into three phase, which shows in Figure 1 from the basic business perspective:

- ***Setup Phase*** - The user *U* is in possession of data sequence F. *U* runs $(pk, sk) \leftarrow KeyGen(\lambda)$, and stores the key pair (sk, pk). And then, it broadcasts the public key pk on the network to Auditing System *A* and Cloud Storage Provider *CSP*.
- ***Synchronize Phase*** - Before sending the file *F* to *CSP*, the *U* runs $\Phi \leftarrow TagBlock(pk, F)$ for each data blocks in F to generate the collection of HVDs. Then, it sends F to *S* for storage and deletes F from its local storage. At the same time, it send Φ to *A*. Upon receiving the collection of HVDs, Φ, the *A* storages it.
- ***Verify Phase*** - When starting auditing process, *A* runs $PregenProof(N)$ to generate a random index array $chal$ to retrieve the data blocks. Then, it sends $chal$ to *S*. When receiving the random index array $chal$, *S* runs $V \leftarrow GenProof(pk, F, chal)$ to construct the possession proof V, and then, sends ti to *A*. Finally, *A* can check the validity of the proof V by running $CheckProof(pk, chal, V)$. Once in a while, A shows U the auditing result in detail.

In the **Setup** phase, *U* generates a key pair and broadcasts the public key pk to Cloud Storage Service and Auditing System. In the **Synchronize** phase , *U* computes HVD for each file block in file F, which are constructed as a HVDs collection denoted as Φ. Then, *U* stores F and Φ at *S* and *A*, respectively. In the **Verify** phase, *A* requests proof of possession for a subset of the blocks in F. *S* constructs the proof of possession, V, and sends it back to *A*. This phase can be executed an unlimited number of times in order to ascertain whether *S* still possesses the selected blocks.

Now, we will describe the basic protocol in detail. We propose to uniquely integrate the homomorphic authenticator with a random masking technique. In our basic protocol, the linear combination of sampled blocks in the cloud servers response is masked with randomness generated by a pseudo-random function (PRF) which is defined as previous and called f. With random masking, *A* no longer has all the necessary information to construct a correct group of linear equations and therefore cannot derive the users data content, no matter how many linear combinations of the same set of file blocks can be collected. Meanwhile, because of the algebraic property of the homomorphic authenticator, the correctness validation of the block-authenticator pairs will not be affected by the randomness generated from a PRF, which will be shown shortly. Also, the index of sampled blocks in *A* will be randomly generated from pseudorandom permutation(PRP) which is also defined as previous and called π. With random selected, it's extremely hard for the Cloud Server to gets a very similar index array which is used to select data blocks.

- $KeyGen(\lambda) \rightarrow (pk, sk)$ - Given a security parameter $\lambda \in \mathbb{Z}_{\text{l}}$, generate a tuple (p, q, G, G_1, e), where p and q are two distinct large primes, G is a cyclic group of order pq, and e is a pairing map $e : G \times G \rightarrow G_1$. Let $N = pq$. Pick up two random generators g, u from G

and set $w = u^q$. Then w is a random generator of the subgroup of G of order p. To simplify, the public key is $pk = \{N, e, g, w, q\}$. The private key $sk = \{p\}$.

- $SigBlock(pk, F) \rightarrow (\Phi)$ - Given data file $F = (m_1, ..., m_n)$, compute signature T_{m_i} for each block m_i : $T_{m_i} \leftarrow (H^i g_{m^i})^q mod N \in G(i = 1, ..., n)$. Denote the set of signatures by $\Phi = \{T_{m_i}\}_{1 \le i \le n}$.
- $PregenProof(N) \rightarrow (\{(i, v_i)|i \in I\})$ - pick a random c-element subset $I = \{s_1, ..., s_c\}$ of set $[1, n]$, where $s_t = \pi_{key}(t)$ for $1 \le t \le c$ and key is the randomly chosen PRP key. For each $i \in I$, a random v_i will also be generated from f, where $v_i = \pi_{key}()$. What's more, assume that $s_1 \le ... \le s_c$.
- $GenProof(chal, pk, F) \rightarrow (M)$ - choose a random element $r \leftarrow \mathbb{Z}_\iota$, via $r = f_{key}(chal)$, where key is the randomly chosen PRF key, and calculate $R = w^r \in G$. Let $\delta^{'}$ denote the linear combination of sampled blocks specified in $chal$: $\delta^{'} = \sum_{i \in I} v_i m_i$. To blind δ with r, compute $\delta = \delta^{'} + rh(R) \in \mathbb{Z}_\iota$. Meanwhile, calculate an aggregated signature $\sigma = \prod_{i \in I} \sigma_i^{v_i} \in G$.
- $VerifyProof(pk, chal, V) \rightarrow (\{0|1\})$ - To prove whether the data on *CSP* has been modified or not, validate verification equation as follows:

$$e(\sigma \cdot R^{h(R)}), u) \stackrel{?}{=} e(\prod_{s_1}^{s_c} H(i)^{v_i} \cdot g^{\delta}, w) \qquad (1)$$

Where e is a bilinear pair defined as previous.

C. Model Analysis

1) Homomorphic Verification: When *A* verifies the sampling proof, $V = (\sigma, \delta, R)$ returned by *CSP*, and if the sampled data blocks are not bad blocks, *A* will get the result after proving equation (1). Equation (1) could be elaborated as follows:

The left part

$$\begin{aligned} e(\sigma \cdot R^{h(R)}), u) &= e(\prod_{s_1}^{s_c} (H(i) \cdot g^{m_i})^{v_i q} \cdot g^{qrh(R)}, u) \\ &= e(\prod_{s_1}^{s_c} (H(i) \cdot g^{m_i})^{v_i} \cdot g^{rh(R)}, u)^q \\ &= e(\prod_{s_1}^{s_c} (H(i) \cdot g^{m_i})^{v_i} \cdot g^{rh(R)}, w) \\ &= e(\prod_{s_1}^{s_c} H(i)^{v_i} \cdot \prod_{s_1}^{s_c} g^{m_i v_i} \cdot g^{rh(R)}, w) \\ &= e(\prod_{s_1}^{s_c} H(i)^{v_i} \cdot g^{\sum_{s_1}^{s_c} m_i v_i + rh(R)}, w) \end{aligned}$$

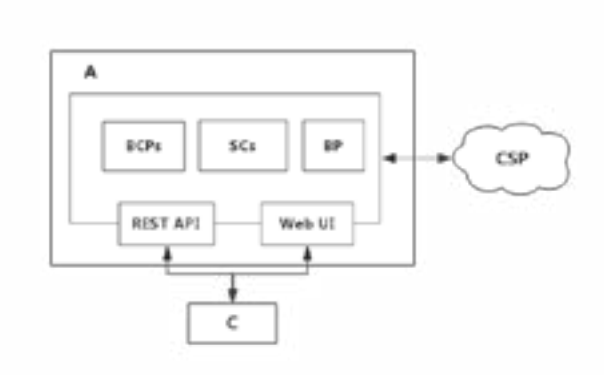


Figure 3. Certain Node Model(CNM)

The right part

$$\begin{aligned} e(\prod_{s_1}^{s_c} H(i)^{v_i} \cdot h^{\delta}, w) &= e(\prod_{s_1}^{s_c} H(i)^{v_i} \cdot g^{\delta^{'} + rh(R)}, w) \\ &= e(\prod_{s_1}^{s_c} H(i)^{v_i} \cdot g^{\sum_{s_1}^{s_c} + rh(R)}, w) \end{aligned}$$

As a result, we deduce that the left part of the verifying equation is equal to the right part. consequently, if *CSP* does not modify or delete data blocks of *U*, the *A* will get an equation when executing auditing process.

2) Sampling Accuracy Discussion: We will take sampling strategies provides high probability assurance. Now, We will make a brief demonstration. Assuming that there are currently *n* data blocks of an account *U* in *CSP*. *CSP* deletes or modifies t data blocks of *U*, denoted as bad data blocks. c is the number of data blocks that *A* requests to validate. X is a discrete random variable, which means that X of c data blocks selected by *A* are modified or deleted by *CSP*, such that $P_X = P\{X \ge 1\} = 1 - P\{X = 0\} = \frac{(n-t)}{n} \frac{(n-t-1)}{(n-1)} \cdots \frac{(n-t-c+1)}{(n-c+1)}$. Since $\frac{(n-i-t)}{(n-i)} \ge \frac{(n-i-t-1)}{(n-i-1)}$, so $1 - (\frac{n-t}{n})^c \le P_X \le 1 - (\frac{n-t-c+1}{n-c-1})^c$. Take the left part $1 - (\frac{n-t}{n})^c \le P_X$; then we get $c \ge \frac{\log(1-P_X)}{\log(1-\frac{t}{n})}$. We take Nature Logarithm and get the equation $1 - e^{\frac{ln(1-Px)}{c}} \ge \frac{t}{n}$. We set c as abscissa and $\frac{t}{n}$ as ordinate. When taking $Px = 99\%, 90\%, 80\%$, we will get the curve which was shown in Figure 4 We find that c depends on three parameters P_X , t, and n. In order to facilitate the testing, we assume that there are 1% percent of bad blocks in Cloud Server. At the moment , if taking $P_X = 99\%$ and 90%, c equals to about 460 and 230, respectively. When testing the prototype system, we will set $P_X = 99\%$ and $c \ge 460$ in the Performance Analysis and Evaluation Section, and present the experiment result based on these sampling strategies. Given the huge volume of data outsourced in the cloud, checking a portion of the data file is more affordable and practical for both *A* and *S* than checking all the data, as long as the sampling strategies provides high probability assurance.

III. Basic System Model

In this section, we will introduce the basic system model from part to whole. It will be divided into two parts: Certain Node Model(CNM) and Multi-Node Model(MNM). The CNM is a single node model, and in this subsection, we will introduce main function components, and how *A* works when collaborating with *U* and *S*.

1) Certain Node Model: As shown in Figure 3, its a Certain Node Model(CNM), which could be considered as an Auditing Service, and includes some components, *Web UI*, *REST API*, *BlockChain*[20], [21], [22] Protocols (BCPs), *Smart Contract Protocols* (SCPs)[12], [13], and *BP*.

U could interact with *A* through *Web UI* or *REST API*. The function of *Web UI* is so simple that it's only used to display the auditing results and monitor system performance. *REST API* has much richer features which not only provide all programming interfaces but also include visual interfaced, Web UI. Moreover, *BP* is used to assist *U* and *CSP* in both *Setup* and *Verify* procedures. *SCs* are the main protocols for the *U* to interact with *A*. *BCPs* is the interaction protocols between nodes in auditing network. The auditing network is a BlockChain Network Service. We build a BlockChain Network Service through *BCPs* and *SCPs* (using Ethereum[]), which is the basic service network for data integrity auditing service in our experimental system. And it will be fully described in the Multi-Node Model section.

For CNM, it provides each account with the execution and storage space(maybe user have many Account) that is isolated from the other accounts. Compared with other accounts, this feature makes every single individual activity of the account similar to one transaction. Moreover, it can well ensure that the user's activities are not disturbed by other users. So, it is extremely useful to protect user's private data. Furthermore, in the case without considering the capacity of the network, when giving each individual execution and storage space, it can provide more users auditing service.

2) Multi-Node Model: In the previous section, we describe the architecture of a certain node model and how it works. In this section, we will introduce the role that single-node service plays in the network, and features of this decentralized network. As shown in Figure 3, the dotted line reflects how an auditing node assists *U* and *CSP* to complete *Setup* and *Verify* phase; the solid-line constitutes a mesh network model, which is a decentralized network and includes various types of nodes, such as nodes dedicated to auditing, and auditing nodes embedded in *U*.

Then, we describe the characteristics of this mesh in detail. This decentralized network is built on BlockChain protocols and inherits all capabilities of BlockChain network. First of all, the dispersion of this decentralized network makes data in the network completely dispersed, and each node in the network shares and possesses all the data on

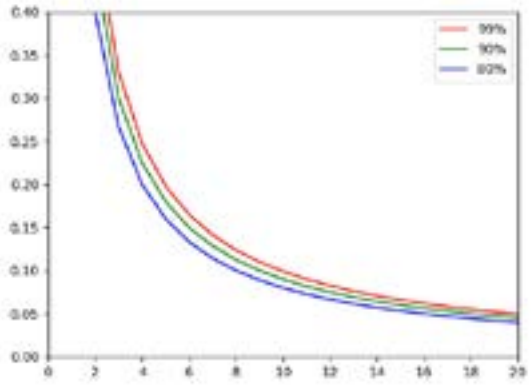


Figure 4. We show $\frac{t}{n}$ as a function of c (the number of data blocks queried by the client), t is the number of bad data blocks, n is the total of data blocks, $F_X = 80\%, 90\%, 99\%$(a discrete random variable)

the entire network. Second, data records in the network have only-increased features, which makes existing data records in the network cannot be tampered with at all. More importantly, every node in network witnesses the review and storage of data records, ensuring that entire business logic can be observed throughout the network. The visibility and tampering of data throughout the network guarantee the credibility of the data exposed in the network, making the network a decentralized and trusted network. Moreover, as a basic component supported by Blockchain, Smart Contract supported makes it possible that we could write the more complicated program. That is to say, all the logical business will be monitored by this trusted network, including input, execution, and output. Apart from data transmission module, the main auditing function is built on Smart Contract.

Finally, we describe how a deployed decentralized network works. First, *U* registers for auditing service provided by A. Second, with the cooperation of auditing system *A*, *U* stores its datasets which could be regarded as a file sequence, on the cloud storage server. *A* periodically completes data integrity auditing procedure with the help of S and records the auditing results on Blockchain. During this period, we know that all nodes in auditing service network witness the entire data stored procedure, participate in the process of auditing, verify data integrity and store the audit result. Also, each *U* that uses auditing service has a unique, isolated space to store its private data sets, and some other information.

IV. Performance Evaluation

We now assess the performance of the user-oriented and decentralized data integrity audit scheme. The experiment is conducted using go language on a Linux system with an Intel Core 5 processor running at 3.30 GHz, 12GB of RAM.

We will focus on the response speed of the experimental prototype system to evaluate system performance. We assume that the data is available in the form of different sizes of data blocks and a data set is composed of some

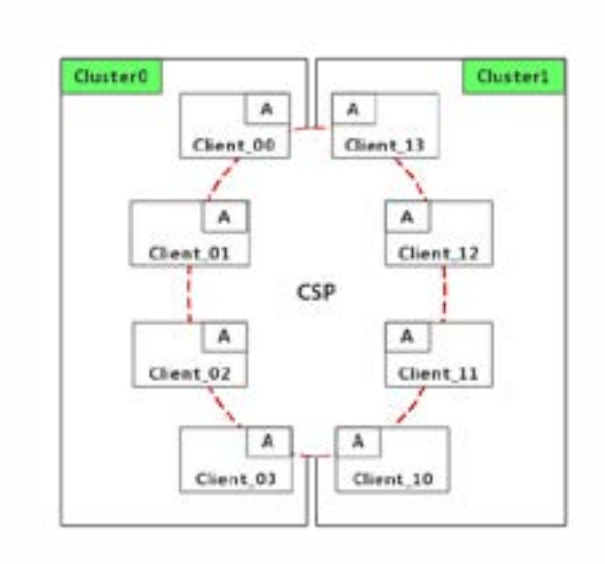


Figure 5. Testing Network

those data blocks. In general, we should check the integrity of data blocks before the integrity of the whole data sets can be verified. Our experimental prototype system is built upon a private Blockchain network with a minimum of two clusters, Cluster0 and Cluster1, where Cluster0 are used as SPV[20], [21] and Cluster1 are used as a full-time[20], [21] audit node at the moment, and each cluster has a minimum of 4 nodes which denotes as client with auditing module. All nodes in two clusters are connected to *CSP* to constitute a testing network, as shown in Figure 5.

A. System Efficiency Evaluation

Algorithms use the Pairing-Based Cryptography (PBC) library[1] version 0.5.14. The elliptic curve utilized in the experiment is an MNT curve, with the base field size of 159 bits and the embedding degree 6. The security level is chosen to be 80 bit, which means $|v_i| = 80$ and $|p| = 160$. Moreover, The *HVD* will be 1024 bits long. All experimental results represent the mean of 20 trials.

We will mainly focus on *Synchronze Phase* because system performance bottlenecks depend on its two subprocesses, (1) data transmission and storage process from *U* to *S* and (2) data transmission and writing process from *U* onto Blockchain(we use Ethereum [21] Private Local Network for experiments). In our experiments, if we are to transfer more than 1M data each time, we will divide it into *1M* size blocks. Thus, the greater the amount of data are transmitted, the more HVDs will be constructed. We set the amount of the data to be transferred each time as 1KB, 10KB, 10^2KB, 10^3KB, 10^4KB, 10^5KB and 10^6KB, the time-consumption trend of the two subprocesses, (1) and (2), are shown by the red and blue polylines in Figure 6. For now, we know each Blockchain platform [20], [21], [22] limits the size of a block

[1]https://github.com/blynn/pbc

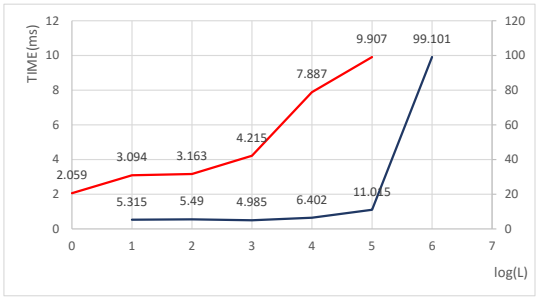


Figure 6. time-consumption of two sub-process in *Synchronize Phase*, abscissa denotes as common logarithm of data block size, denoted as $\log(L)$, Principal coordinate and sub coordinate denotes as time-consumption of the two sub-process with red and blue polylines in different scale, and unit is ms.

to balance the speed of block construction.Thus, the more HVDs are generated, the greater the writing pressure on Blockchain will be. So, we could find that when the amount of transferred data grows from 10^5 to 10^6, time-consumption of (2) increases rapidly, and (1) is the same. Therefore, we conclude that in the case of a large scale of data, no matter how the data block is cut, there is a steep inflection point in both curves. In this case, in order to balance the speed of (1) and (2), we take a balance point in two incremental curves with the point $(5, 0)$ on a horizontal axis. That's is to say, the size of every data block is *1M*.

As for the auditing result writing process, we don't care about the writing speed. On the contrary, it's is the auditing result itself that we focus on. What's more, auditing response time. Most of the time, we think cloud server audit response time is much faster than the data writing, because of the high efficiency of data retrieval on the cloud server.The process of building a proof can be time-consuming, but its time complexity will be negligible on the same order of magnitude as the HVD generated during data synchronization.

B. Execution Result

To facilitate the visual display of audit results, we design a simple *Web UI*, which shows the audit results of an account which may store its data on a different remote storage server. Figure 7 shows the *Web UI* of *TDAS*, in which each row represents the result of an audit task. Its content consists of (1) *StartingTime* denotes starting time of one audit task; (2) *Parameters*. *NoSM* indicates the number of sampling data block; *TNoM* denotes Total Number of Data Block; *t/N* denotes bad data block ratio; (3) *Result* denotes result of audit task; (4) *Audit-Time* denotes how long an audit task will be; (5) *Server Address* denotes where the data is stored.

V. Related Work

A. Cloud Storage Service

Cloud Storage Service Data Integrity Audit verifies the data integrity of an untrusted cloud storage server and prevents user data from being tampered with or deleted in any way, so as to restrict cloud storage service providers' misconduct.

For providing the integrity and availability of remote cloud store, the Cloud Storage Service need to be public auditability. All previous solutions are based on this principle. Previous studies of cloud storage data integrity mainly focus on Provable Data Possession (PDP) and Proof of Retrievability (POR). Ateniese et al.[8] is the first to consider public auditability in their defined "provable data possession" (PDP) model for ensuring possession of data files on untrusted storages. Their scheme utilizes the RSA-based homomorphic authenticators for auditing outsourced data and suggests randomly sampling a few blocks of the file. However, the public auditability in their scheme demands the linear combination of sampled blocks exposed to an external auditor. When used directly, their protocol is not provably privacy-preserving, and thus may leak user data information to the auditor. Juels et al. [23] describe a "proof of retrievability" (POR) model, where spot-checking and error-correcting codes are used to ensure both "possession" and retrievability of data files on remote archive service systems. However, the number of audit challenges a user can perform is a fixed prior, and public auditability is not supported in their main scheme. Although they describe a straightforward Merkle-tree construction for public PoRs, this approach only works with encrypted data. Based on the security model defined in [7], Shacham et al. [16] designed an improved PoR scheme with BLS signature construction, which fully proved the security. Similar to the constructs in [8], they use verifiable homomorphic validators built from BLS signatures that prove secure. Based on the BLS signature build, the scheme achieves public auditability, but their methods do not support privacy protection. Wang et al.[19] proposed an improvement on the [8] scheme to provide tighter user privacy protection strategies and proposed a batch audit solution. This solution places the audit efficiency on a mass audit, and the audit results are believed to be at the authority of the audit institution. On the one hand, the scheme does not consider the irreconcilability between the audit efficiency and the resource consumption of the third-party auditing institutions; the batch audit will either increase the input of resources or perform the automated audit. At this moment, the credibility of the audit result is questionable. More importantly, there is a common problem with current TPA-based data auditing techniques, which is time-consuming and labor-intensive compared to an automated auditing process. Furthermore, even with the appropriate encryption technology, there is still a risk of information disclosure due to the addition of third-party personnel.

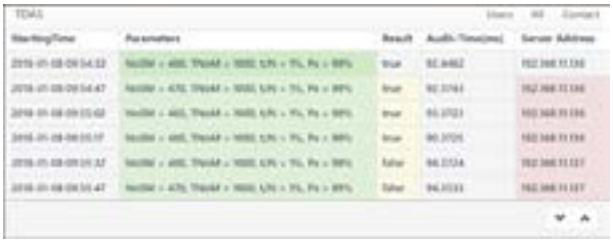


Figure 7. Visual Display of Audit Results

B. Blockchain

Blockchain[17] is based on a list of key protocols such as cryptographic authorization protocol, consensus autonomy protocol and distributed storage protocol. With the blockchain protocol, we can build a network of trusted values on untrusted network nodes - called distributed general ledger technology. Bitcoin[20], Ethereum[21] and HyperLedger[22] is typical platforms for the current Blockchain.

Encryption Authorization protocol to ensure the privacy of user information and data privacy. There is data privacy, data privacy protection, our common approach is to all data storage and query data encryption tools to protect the privacy of data. Mandate mandatory data, the data is called First, data access should be considered. The blockchain is an openly distributed ledger. Under the precondition of data sharing in the whole network, there will be the problem of whether the data is provided or not. Encryption Authorization Protocol addresses the issue of secrecy and authorization during data transfer, storage, and shared access.

Consensus autonomy guarantees record traceable and tamper-proof. As the core protocol in the blockchain protocol group, consensus self-made protocol is a distributed consistency algorithm that solves Byzantine fault tolerance. Traditional distributed conformance protocols require that the network environment is trusted, by default there are no malicious nodes that will attack the network and tamper with the data. Blockchain networks based on P2P networks cannot provide this guarantee. Therefore, the addition of Byzantine fault tolerance mechanism provides a guarantee for the credibility of nodes. Consensus mechanism based on consensus self-made protocol guarantees the anti-tampering characteristics of data in the network and provides a powerful guarantee for the credibility of data in the network (but there will still be 51% of attacks). Of course, one of the major problems of the consensus mechanism is the low data write rate caused by the network synchronization, but the query efficiency is high. This problem in the audit process should be the most affected during the data synchronization process homomorphic validation summary write problem, we will solve this problem by other means.

Distributed storage protocol to ensure that data records are

not lost. Distributed storage protocol built on P2P networks ensures that all nodes in the network have a copy of all network data. This decentralized storage ensures that data records in the network are not lost.

In addition, Smart Contract technology[12], [13] is based on the second generation of blockchain technology. Blockchain-based smart contracts include transaction processing and preservation mechanisms, as well as a complete state machine for accepting and processing various types of smart contracts; and everything is saved and the state is handled on the blockchain. The transaction mainly contains the data to be sent; the event is the description of the data. Transaction and event information into the smart contract, the resource status of the contract resource set will be updated, and then trigger the smart contract state machine judge. If the trigger condition of one or several actions in the automatic state machine is satisfied, the state machine will automatically execute the contract action according to the preset information.

VI. Conclusion

In this work, we introduce a new scheme–User Oriented and Decentralized Data Integrity Auditing Scheme to solve the public data integrity problem on Cloud Storage Service. It can not only make a efficient data integrity auditing, but also make sure the process safe and the result trustful and reliable. We applied it to audit the data integrity on Could Storage Service, which demonstrates efficacy and reliability on large-scale data. And also, it could be deployed as a foundational data and authority auditing component in Cloud Computing architecture model. In future work, we plan to expand our work to more domains including, for example, Internet of things, Data Manager on Internet and so on.

However, the limitations of this solution are also obvious. Due to the defect of Blockchain principle, its writing speed has always been criticized. Therefore, when encountering large-scale data, limited by the writing speed of the Blockchain, the audit system write rate may be a performance bottleneck. However, with the continuous development of Blockchain technology and the continuous introduction of various solutions, this situation is gradually being improved, for example, solutions such as Sidechain, lightning network and isolation witness etc. Therefore, in the long run, our solutions will become more effective.

Acknowledgment

The authors would like to thank the anonymous reviewers for their comments. This work is supported by the National Natural Science Foundation of China grants 61502333, 61572350, 61672377 and the Tianjin Research Program of Application Foundation and Advanced Technology grant 14JCYBJC15600. Thank for corresponding author Keman Huang.

References

[1] Amazon. (2007). Amazon simple storage service (amazon s3). Amazon [Online]. Available: *http://aws.amazon.com/s3/*.

[2] Microsoft Azure. (2008). Microsoft Azure Blob Storage Service. Microsoft [online]. Avaiable: *https://azure.microsoft.com/*.

[3] Aliyun Object Storage Service(Aliyun OOS). (2011). Aliyun Object Storage Service. Alibaba [online] Avaiable: *https://www.aliyun.com/*.

[4] Google. (2005). Google drive. Google [Online]. Available: *http://drive.google.com/*.

[5] Dropbox. (2007). A file-storage and sharing service. Dropbox [Online]. Available: *http://www.dropbox.com/*.

[6] Bitcasa. (2011). Inifinite storage. Bitcasa [Online]. Available: *http://www.bitcasa.com/*.

[7] T. Jiang, X. Chen, and J. Ma. public integrity auditing for shared dynamic cloud data with group user revocation. *IEEE Trans. Computers*, 65(8):23632373, 2016.

[8] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song. Provable data possession at untrusted stores. *Cryptolog-y ePrint Archive*, Report 2007/202, 2007, *http://eprint.iacr.org/*.

[9] H. Shacham and B. Waters. Compact Proofs of Retrievability. *Proc. of Asiacrypt 2008*, vol. 5350, Dec 2008, pp. 90107.

[10] M. Venkatesh, M.R. Sumalatha and C. SelvaKumar. Improving public auditability, data possession in data storage security for cloud computing. *2012 International Conference on Recent Trends in Information Technology (ICRTIT)*, 463-7, 2012.

[11] F G. ZHANG. From bilinear pairings to multilinear maps[J]. *Journal of Cryptologic Research, 2016*, 3(3): 211228.

[12] K. Christidis and M. Devetsikiotis. Blockchains and Smart Contracts for the Internet of Things. *IEEE ACCESS*, 2292-2303, 2016.

[13] Kosba, A; Miller, A ; Shi, E; Wen, Z; Papamanthou, C. Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts. *IEEE Symposium on Security and Privacy (SP)*, MAY 23-25, 2016.

[14] M. Bellare and A. Palacio. The knowledge-of-exponent assumptions and 3-round zero-knowledge protocols. *In Proc. Of CRYPTO 04, Lecture Notes in Computer Science*, pages 273289. Springer, 2004.

[15] Gentry and Craig. A fully homomorphic encryption scheme. *Stanford University*,2009.

[16] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia. Dynamic provable data possession. *in Proc. of CCS*,09, 2009.

[17] Anh, Dinh Tien Tuan and Zhang, Meihui and Ooi, Beng Chin and Chen, Gang. Untangling Blockchain: A Data Processing View of Blockchain Systems. *IEEE Transactions on Knowledge & Data Engineering*, 1-1, 2017.

[18] Dan, Boneh and Goh, Eu Jin and Nissim, Kobbi. Evaluating 2-DNF formulas on ciphertexts. *Theory of Cryptography Conference*, 05, 2005, pp. 325-341.

[19] Cong Wang, Qian Wang, Kui Ren, Wenjing Lou Privacy-Preserving Public Auditing for Data Storage. *INFOCOM, 2010 Proceedings IEEE*.

[20] Bitcoin. (2008). Bitcoin [Online]. Available: *https://www.bitcoin.com/*.

[21] Ethereum. (2014). Ethereum [Online]. Available: *https://www.ethereum.org/*.

[22] Hyperledger. (2015). Hyperledger [Online]. Available: *https://www.hyperledger.org/*.

[23] A. Juels and J. Burton S. Kaliski. Pors: Proofs of retrievability for large files . *in Proc. of CCS07*,Alexandria, VA, October 2007, pp. 584597.

Comprehensive Evaluation of Cloud Services based on Fuzzy Grey Method

Wenjuan Li[1,2,3], Jian Cao[1] and Shiyou Qian[1]

1. Computer Science and Technology, Shanghai Jiao Tong University, Shanghai 200240, China

2. Qianjiang College, Hangzhou Normal University, Hangzhou 310036, China

3. Cloud Computing and Distributed Systems (CLOUDS) Laboratory, School of Computing and Information Systems, the University of Melbourne, Melbourne VIC 3010, Australia

Abstract: Cloud-based applications have become more and more popular. However, it remains a big challenge to comprehensive evaluate the reliability and performance of cloud services due to the unique features of cloud. The grey theory is adapted to handle the problems of blurring and uncertainty. Therefore, based on the grey comprehensive evaluation method, this paper proposes a novel trust comprehensive evaluation method for cloud services, also a comprehensive user satisfaction evaluation method for the better selection of suitable providers. In addition, it discuss in detail the construction and calculation of the evaluation model by case study.

Keywords: Cloud computing, Grey comprehensive evaluation, trust, user satisfaction

1 Introduction

Cloud computing is an Internet-based service sharing platform which has received extensive attention [1]. At present, the service evaluation system of cloud is still immature due to the reason that it is a multi-attribute, multi-factor and uncertainty issue. Therefore, in the face of increasing options, cloud users often find it difficult to obtain the most cost-effective service through the simple comparisons. So, it is necessary and urgent to propose a suitable comprehensive evaluation method for cloud services.

This paper introduced the grey comprehensive evaluation method (GCE) into cloud computing. And based on GCE method, it studied the comprehensive evaluation model of cloud services, put forward new methods for the comprehensive trust and user satisfaction evaluation of cloud services.

The main contributions of this paper are as follows:

(1) It introduces GCE method to construct the evaluation model for cloud services.

(2) It proposes a novel method of comprehensive trust evaluation based on GCE.

(3) It proposes a novel method of comprehensive user satisfaction evaluation based on GCE.

This paper is organized as follows: section 2 introduces the method of grey comprehensive evaluation method, section 3 proposes a comprehensive trust evaluation model of cloud providers based on GCE method, and section 4 presents a user satisfaction evaluation model based on GCE method. The last section are the conclusions.

2. Grey Comprehensive Evaluation Method

Grey Comprehensive Evaluation (GCE) can quantify the difference between the evaluation object and the desired object, conduct a comprehensive evaluation of the evaluation object, so as to compare, sort, and help users to make a decision.

Suppose the final evaluation result of m objects that need to be evaluated is vector R (R $= [r_1, r_2, \dots, r_m]^T$). R is obtained by the GCE model R = E × W。

The evaluation result of the absolute relevance of m objects is the following matrix E.

$$\mathrm{E} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \cdots & \varepsilon_{1n} \\ \varepsilon_{21} & \varepsilon_{22} & \cdots & \varepsilon_{2n} \\ \vdots & \vdots & & \\ \varepsilon_{m1} & \varepsilon_{m2} & \cdots & \varepsilon_{mn} \end{bmatrix}$$

Here, $\varepsilon_{ik}$is the correlation coefficient between the k-th index and the k-th best index of the i-th alternative. W (W $= [w_1, w_2, \dots, w_n]^T, \sum_{i=1}^{n} w_i = 1$） is the weight vector of n evaluation indicators, when considering the weight of different indicators.

The main steps of GCE are: (1) Determine the optimal indicator vector; (2) Construct an initial evaluation matrix; (3) Normalized the data; (4) Obtain the evaluation matrix; (5) Obtain the final evaluation result R based on the weight vector and the evaluation matrix.

3. Trust Comprehensive Evaluation based on GCE

Trust is an effective alternative of security in the distributed computing systems like Cloud. However, trust evaluation is a comprehensive evaluation problem.

3.1 Trust Comprehensive Evaluation Model

Assume that only three kinds of cloud services (computing service, network service and storage service) are considered. The vector CT_i represents the comprehensive trust of provider i.

$$CT_i = (t_{cpu}, t_{bd}, t_{storage})$$

Here, t_{cpu} refers to the trust of the provider when providing computation services, t_{bd} means the trust of providing network services, and $t_{storage}$ is the trust of providing storage services.

Fig.1 shows the general steps of trust comprehensive evaluation based on GCE method.

Fig.1 The basic process of GCE based comprehensive trust evaluation

3.2 A Case Study

Assume that a cloud user wants to choose a most credible provider from the following five cloud service providers. The historical best value is used as a reference to each index. The historical trust evaluation of the five providers are shown in Tab. 1.

Tab.1 The original trust value of the providers

Provider	t_{cpu}	t_{bd}	$t_{storage}$
P1	0.8	0.5	0.9
P2	1	0.8	0.9
P3	0.7	0.9	0.7
P4	0.6	0.7	0.8
P5	0.9	0.6	0.6
Optimal Value	1	0.9	0.9

Thus, the original data matrix D is as follows.

$$D = \begin{pmatrix} 1 & 0.9 & 0.9 \\ 0.8 & 0.5 & 0.9 \\ 1 & 0.8 & 0.9 \\ 0.7 & 0.9 & 0.7 \\ 0.6 & 0.7 & 0.8 \\ 0.9 & 0.6 & 0.6 \end{pmatrix}.$$

Take the resolution coefficient $\alpha = 0.5$, according to the GCE method, we obtain the following evaluation matrix E.

$$E = \begin{pmatrix} 0.50 & 0.33 & 1 \\ 1 & 0.67 & 1 \\ 0.40 & 1 & 0.50 \\ 0.33 & 0.50 & 0.67 \\ 0.67 & 0.40 & 0.40 \end{pmatrix}$$

Assume that the weight vector of the above evaluation index is W=(0.30, 0.10, 0.60). After calculation, the comprehensive trust of each provider is finally obtained.

t_1= 0.30*0.50+0.10*0.33+0.60*1=0.783

Similarly, we can get t_2=0.967, t_3=0.52, t_4=0.551, t_5=0.481. Therefore, the credibility order of the service providers is P2>P1>P4>P3>P5.

4. User Satisfaction Evaluation based on GCE

User satisfaction is an important basis for evaluating the quality of products and services. However, satisfaction evaluation often involves many different evaluation indicators, and the final conclusions are usually obtained after a comprehensive assessment of all the various factors. To carry out an accurate user satisfaction evaluation, we firstly need to determine the indicators that users are concerned with. Secondly, the optimal value for each indicator need to be decided. The third step is to obtain the user feedback data of the providers. Finally, the user satisfaction of the providers are calculated using GCE method introduced on the former chapter based on the reference evaluation indicators, so as to provide a basis for users to effectively select service companies. Fig.2 shows the main step of GCE based user satisfaction evaluation method.

5. Conclusions

The evaluation of cloud services belongs to the multi-attribute evaluation problem, which is also subjective and ambiguous. Based on the GCE method, this paper proposes a comprehensive trust evaluation model and a user satisfaction evaluation model and through the case study, it describes in detail the implementation process of the models.

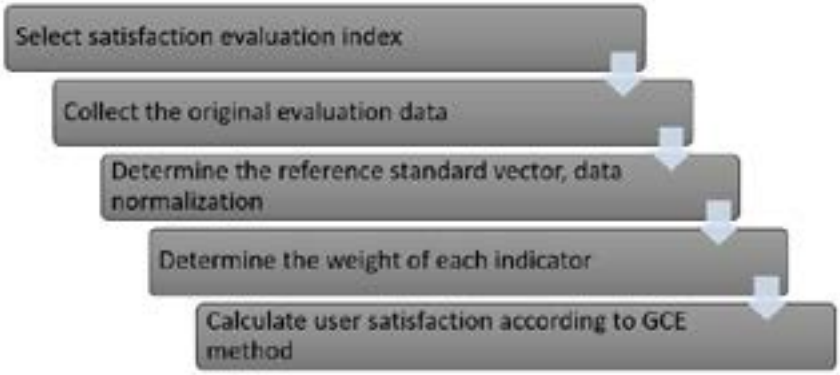


Fig.2 The basic process of GCE based user satisfaction evaluation

Acknowledgments. This work was supported by a grant from National Natural Science Foundation of China No. 61702151, 61772334 and 61472253, Zhejiang Provincial Natural Science Foundation No. LY17E070004 and the Research Project for Department of Education of Zhejiang Province No. Y201635438, the scholarship for visiting scholar from the China Scholarship Council (CSC) No. 201709645006.

References

[1] P. Liu, Cloud Computing (the third edition), Beijing, China: Electronic Industry, 2015.
[2] D. Du, Q. H. Pang and Y. WU, Modern Comprehensive Evaluation methods and cases selection, (second edition), Tsinghua University Press, Beijing, 2014.
[3] W. Li, L. Ping, Q. Qiu, Q. Zhang, Research on trust management strategies in cloud computing environment, Journal of Computational Information Systems，2012.03.10, 8(4):1757-1763.
[4] S. Wu, T. Zhang. Multi-level Fuzzy-Gray Comprehensive Evaluation of Information Security Risk. Proceedings of 2010 International Conference on Management and Service Science (MASS), IEEE, 2010.
[5] k. Haung, Z. Ma, L. Sun. Performance evaluation of node in cloud storage. Proceedings of 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), IEEE, 2012.
[6] Y. Song, L. Wang. Software Trusted Comprehensive Evaluation Model Based on Fuzzy Grey Method. Proceedings of 2010 2nd International Conference on Network Security Wireless Communications and Trusted Computing (NSWCTC), IEEE, 2010.
[7] R. Liu. Euclid distance with weight and its applications. Mathematical Statistics and Applications, 2002,21(5):17-19.
[8] J. Luo and W. Zhu. User similarity function considering weight of items similarity. Computer Engineering and Applications, 2015,51(8): 123-127.

A Dynamic Programming-based Approach For Cloud Instance Types Selection and Optimization

Pengwei Wang, Wanjun Zhou, Xiaobo Zhang, Yinghui Lei, Zhaohui Zhang

School of Computer Science and Technology, Donghua University, Shanghai, China, E-mail: wangpengwei@dhu.edu.cn

Abstract: With the advantages of cloud computing gradually highlighted, users increasingly want to deploy services in the cloud to reduce costs. Cloud providers (e.g. Amazon) at home and abroad provide a large amount of cloud instance types optimized to fit different use cases, such as compute optimized. Faced with such a great quantity of different cloud instance types in the public cloud market, it is a big challenge for users to select appropriate cloud instance types so as to meet their requirements and constraints, and more importantly, to achieve the goal of optimization on the criteria as cost and performance. In this paper, a dynamic programming-based approach is proposed for cloud instance types selection, which can provide optimal combination of cloud instance types to users. Extensive experiments are performed to show the effectiveness of the proposed method.

Keywords: cloud computing, cloud instance type, dynamic programming

1 Introduction

With the advantages of cloud computing gradually highlighted, the users increasingly want to deploy services in the cloud to reduce costs. In order to fit different use cases, these cloud providers, such as Amazon (https://aws.amazon.com/cn), Microsoft (https://www.azure.cn/) and Ali (https://www.aliyun.com/), have introduced numerous cloud instance products. Cloud instance products contain a variety of cloud instance families, such as optimized computing. Each cloud instance family contains different cloud instance series. The cloud instance series contain specific cloud instance types (Instance types comprise varying combinations of CPU, memory, storage, and networking capacity). Thus, there are a large number of different cloud instance types in the public cloud market. For example Amazon (https://aws.amazon.com/cn), as shown in Table 1, launches their own cloud products.

Faced with this situation, it is difficult for users to select appropriate cloud instance types to meet their requirement. We propose a novel approach for cloud users which provides an

Table 1 Amazon EC2 launch cloud instance type families

General Purpose			Compute Optimized		Memory Optimized			Storage Optimized		GPU Instances		
T2	M4	M3	C4	C3	X1	R4	R3	I3	D2	P2	G2	F1
t2.nano	m4.large	m3.medium	c4.large	c3.large	x1.32xlarge	r4.large	r3.large	i3.large	d2.xlarge	p2.xlarge	g2.xlarge	f1.2xlarge
t2.micro	m4.xlarge	m3.large	c4.xlarge	c3.xlarge	x1.16xlarge	r4.xlarge	r3.xlarge	i3.xlarge	d2.2xlarge	p2.8xlarge	g2.8xlarge	f1.16xlarge
t2.small	m4.2xlarge	m3.xlarge	c4.2xlarge	c3.2xlarge		r4.2xlarge	r3.2xlarge	i3.2xlarge	d2.4xlarge	p2.16xlarge		
...	...	...	...	...	...	...	...	...	...	...	...	...

optimal combination of instance types. The experimental results demonstrate the approach we proposed can provide an optimal combination of instance types for users.

2 Related Work

Cloud computing as a commercial computing model and service model was put forward to bring a great influence [Wang (2009)]. This phenomenon has aroused people's attention and research. This phenomenon has aroused people's attention and research. The selection and optimization of cloud instance types was a hot spot and a challenge in recent years. At the beginning, these researches mainly focused on single cloud. However, there were a number of problems and challenges in single cloud environment, such as vendor lock-in, availability. Therefore, people turned their eyes to multi-cloud computing in recent years.

The problem of optimal selection of cloud instance types existed in cloud computing environment(single cloud and multi-cloud). At present, the research work about the optimal selection of cloud instance types is mainly focused on the optimization of virtual machine in multi-cloud environment.

When the virtual machine needs to be migrated, the authors should select the optimal instance types or combination. Therefore, the dynamic migration of virtual machine involves the optimization and selection of cloud instance types. There are many researches on virtual machines migration. Such as, Li et al. (2011) proposed a linear programming model to solve the problem of dynamic cloud scheduling through virtual machine migration in multi-cloud environment. Lucas-Simarro et al. (2013) proposed the cloud agent architecture, which was used to record the number of available cloud providers and users' needs to migrate virtual machines dynamically for cloud users. Tordsson et al. (2012) proposed an optimized deployment of virtual machines through cloud agent mechanism. In multi-cloud environment, there were few studies on the selection and optimization of cloud instance types, and they just only considered to apply several cloud instance types.

Based on the above problems, we present a approach based on dynamic programming. The approach can provide users with a combination of optimal cloud instance types.

3 Problem Definition

In this section, we mainly define the mathematical model.

We assume that there are m available cloud providers ($P_1 \cdots P_m$), cloud providers supply n instance type series totally ($ITS_1 \cdots ITS_n$). Instance type series have k specific instance types ($IT_1 \cdots IT_l$) in total. Hence, the total price of the selected instance types (TIP) satisfies

$$TIP = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{l} (C_{ijk} * P_{ijk}) \tag{1}$$

Where P_{ijk} is the price of selected instance types. $C_{ijk} \in (0, 1, 2, \cdots)$ means the number of selection of each instance type.

In this article, there is no better way to evaluate performance. Thus, we temporarily use the data of Amazon. The ultimate overall performance of selected instance types (TIPro) is defined by

$$TIPro = \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{l} (C_{ijk} * Pro_{ijk}) \tag{2}$$

Where Pro_{ijk} is the performance of selected instance type.

In order to better solve the problem, we introduce several constraints. The users can specify the following restriction constraints in the service description template.

- *Cost constraints*. The users specify that agent can use the maximum cost. TIP satisfies

$$TIP = \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{l}(C_{ijk} * P_{ijk}) \leq CostBudget \tag{3}$$

  Where $CostBudget$ is the cost budget specified by the cloud users.

- *Instance types proportional constraints*. The cloud users specify the percentage of a certain of cloud instance type in the cloud instance types combination. For example, the users may specify that the proportion of small instance type in the instance types combination is no less than 30% and no more than 60%.

$$it_{min}(k) \leq \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ijk}}{\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{l} c_{ijk}} \leq it_{max}(k), 1 \leq k \leq l \tag{4}$$

  Where $it_{min}(k)$ and $it_{max}(k)$ refer to the minimum and maximum percentage of k^{th} cloud instance type used in each optimal combination.

It is worth noting that in this article we assume that the number of cloud instance types is sufficient. All in all, the mathematical model can be expressed: ***maximize*** the optimization objective TIPro (2) and the constraints is (3), (4).

4 Dynamic programming-based approach

We analyze and describe how to solve the problem in this section.

4.1 Problem Analysis

The core issue of this paper lies in solving the problem about how to select the optimal combination from plenty of cloud instance types and achieve the goal of optimization on the criteria as cost and performance. Similarly, the knapsack issue in dynamic programming is consistent with the core problem we demand for. In this paper, We consider the performance of the cloud instance types as the value of item, while consider the cost constraint of the cloud instance types as the capacity of the knapsack.

From the mathematical model, we can see that the optimization goal in this paper is the total performance of the cloud instance types combination, while the constraint conditions are cost constraint and instance types constraint. For the sake of convenience, we only consider the cost constraint. Therefore, we consider the performance of the cloud instance types as the value of item, while consider the cost constraint of the cloud instance types as the capacity of the backpack. The core mathematical formula is as follows:

$$SPro_{k,c} = max\{SPro_{k-1,c}, SPro_{k-1,c-P_k} + Pro_k\} \tag{5}$$

Where $SPro_{k,c}$ denotes the current total performance when the former k cloud instance types are put into the combination of instance types with cost constraint of c. Similarly, $SPro_{k-1,c}$ denotes the current total performance when the former $k-1$ cloud instance types are put into the combination of instance types with cost constraint of c. $c - P_k$ represents the rest cost budget when the price P_k of the k^{th} cloud instance type is put

into the combination of instance types with cost constraint c. $SPro_{k-1,c-p_k}$ represents the total performance of the combination when the former $k-1$ cloud instance types are put in combination of instance types with cost constraint of $c - P_k$. Pro_k represents the performance of the k^{th} cloud instance type.

4.2 Problem Solution

In this paper, we apply the core idea of unbound knapsack to select the cloud instance types. The capacity of the backpack is an integer. However, the price of cloud instance types are decimal, which needs to be handled. The core idea of the process is to obtain the minimum price in all cloud instance type and calculate the performance and user-specified cost magnification times according to the minimum price of cloud instance types. Afterwards, the scheduling algorithm selects the cloud instance types according to the method we proposed. The core idea of we proposed method is based on dynamic programming.

5 Evaluation and Discussion

In order to prove the effectiveness of the method, We compare the experimental results with Li et al. (2011). They present a linear integer programming model to find suitable cloud instance type or its combination for dynamic cloud scheduling. We mainly compare the work for solving cloud instance type or its combination. The data used in this paper is crawled from Amazon.

Table 2 The detail information of the instance type used in Figure 3

Instance Type	c3.large	c3.xlarge	c3.2xlarge	c3.4xlarge
Computing capacity	7	14	28	55
instance type prices($/hour)				
US	0.066	0.131	0.263	0.426
EU	0.066	0.131	0.263	0.526
AP	0.07	0.14	0.281	0.562

We use Java as programing language and apply same dataset. We also consider that cost budget is 5$ per hour and lifetime of infrastructure is 1 Hour to run virtual machines. The detail information is in Table 2.

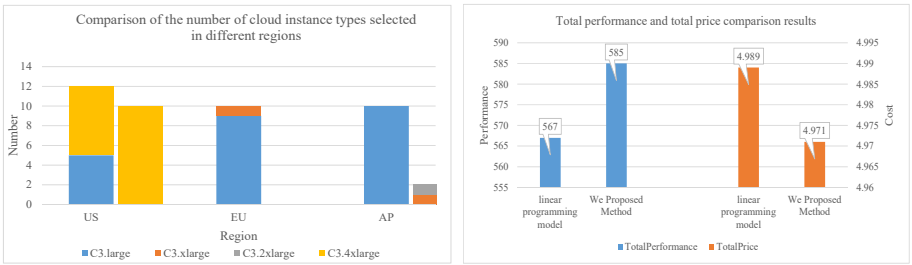


Figure 1: The results in different schedule model

In Figure 2, the left column is the result of linear programming model and the right is the result of our proposed method. The total number of the instance types used in our method is 12 and the total cost of the instance types is 4.971$, which used in Li et al. (2011) is 32 and 4.989$. In the results of the optimal instance types combination, the overall performance

of our method is 585, and which used in Li et al. (2011) is 567, the overall performance is increased by 3.17%. From the run time analysis, the running time of our model is 12 milliseconds, which in Li et al. (2011) is 44 milliseconds. When we expand the datasets, we proposed method is superior to Li et al. (2011) in the number of selected cloud instance types, total cost and total performance in Figure 3.

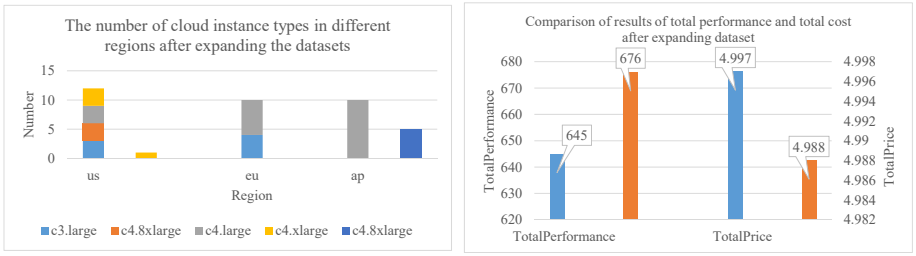


Figure 2: The results in different schedule model after expanding the datasets

6 Conclusion

In this paper, we propose a novel approach based on dynamic programming to solve the problem of cloud instance types selection and optimization, and provide an optimal combination of cloud instance types to the cloud users. More importantly, the experimental results indicate that we proposed approach can provide optimal combination. In the future, we will add optimization objectives, constraints and consider more scenarios.

7 Acknowledgements

This work was partially supported by the National Natural Science Foundation of China (NSFC) under Grant 61602109, Shanghai Sailing Program under Grant 16YF1400300, Shanghai Science & Technology Innovation Action Plan Project under Grant 16511100903, Fundamental Research Funds for the Central Universities, and the Initial Research Funds for Young Teachers of Donghua University.

References

Amazon, https://aws.amazon.com/cn/.

Microsoft Azure, https://www.azure.cn/.

Ali, https://www.aliyun.com/.

Wang, P. (2009). 'Cloud computing', *The People's Posts and Telecommunications Press*, pp. 53-75.

Tordsson, J., Montero, R. S., Moreno-Vozmediano, R., and Llorente, I. M. (2012). 'Cloud brokering mechanisms for optimized placement of virtual machines across multiple providers', *Future Generation Computer Systems*, Vol. 28, No. 2, pp. 358-367.

Li, W., Tordsson, J., and Elmroth, E. (2011). 'Modeling for dynamic cloud scheduling via migration of virtual machines', *Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on*, pp. 163-171.

Lucas-Simarro, J. L., Moreno-Vozmediano, R., Montero, R. S., and Llorente, I. M. (2013). 'Scheduling strategies for optimal service deployment across multiple clouds', *Future Generation Computer Systems*, Vol. 29, No. 6, pp. 1431-1441.

Provisioning Big Data Applications as Services on Containerized Cloud

Jing Gao, Zhuofeng Zhao, and Yanbo Han

Cloud Computing Research Center,
North China University of Technology,
Beijing, China
E-mail: {gaojing, edzhao}@ncut.edu.cn; yhan@ict.ac.cn

Wubin Li

Ericsson Research, Montréal, Quebec, Canada
E-mail: wubin.li@ericsson.com

Abstract: We present a framework aiming at dynamically provisioning big data applications as services on containerized cloud. The innovations are to optimize the whole lifecycle of big data applications in a holistic manner by the adoption of microservices methodologies. The feasibility of our approach is verified through a case study of provisioning a large-scale user traffic data processing application in a private cloud environment backed by Kubernetes.

Keywords: Big data application; Cloud computing; Container; Microservices.

1 Introduction

The rise of big data presents severe challenges for managing and gaining insights from vast amounts of data. It has significantly driven the development of series of new solutions, referred to as *big data applications*, which are usually built on open source software and can be deployed and scaled on commodity hardware.

Recent technology trends in the microservices domain indicate that a solution eliminating the presumed complexity of deploying, maintaining, and operating big data applications may be in sight. Instead of building an application as a monolith, methodologies of microservices advocate to break down a complex monolith into small services that can be independently deployed and maintained. In this paper, we investigate the fundamental challenges for dynamic provisioning of big data applications. We then explore the advantages of applying microservices principles to tackle these challenges through our framework for dynamical provisioning big data applications as services on containerized cloud. Our research demonstrates the feasibility of lifecycle management for big data applications using a microservices approach.

2 Challenges for dynamic provisioning of big data applications

We identify five fundamental challenges that in our view must be addressed in a comprehensive manner for dynamic provisioning of big data applications.

- Configuration and deployment. The deployment of big data applications involves rigorous configuration management and tuning of the execution environment to satisfy various requirements, including performance, security, availability, etc.
- Elasticity. This is the ability to adapt to workload changes by provisioning and de-provisioning resources in an autonomic manner, such that resources allocated to the application match the current demand as closely as possible.
- High availability. For big data applications, any failure or fault in the data pipeline may hinder the availability of the overall system.
- Multi-tenancy. This requires adequate isolation of security, robustness and performance between multiple tenants.
- Version control and coexistence. It is very common that users might need to gracefully upgrade or roll back specific components in the data pipeline without bringing down the whole application. Different tenants might also have their own preferences on different distributions of the same component.

3 Architectural Design and Implementation

Essentially, microservices is a methodology of breaking large software projects into smaller, independent, and loosely coupled modules that are are responsible for discrete tasks. Individual modules are treated as services and communicate with each other through simple, universally accessible APIs.

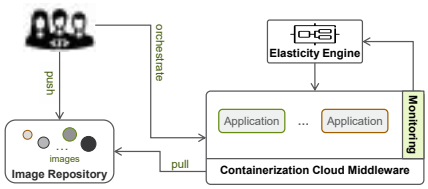


Figure 1: The architectural overview of the proposed approach.

Our framework consist of four major components, namely, the *Image Repository*, the *Containerization Cloud Middleware*, the *Monitoring*, and the *Elasticity Engine*. (i). The *Image Repository* is a place where developer are allowed to store, manage, and deploy container images for big data applications. Version control is supported by applying tags to images. (ii). The *Containerization Cloud Middleware* establishes a highly available containerized execution environment for application components. It takes inputs from developers and orchestrates the whole big data pipelines by pulling required container images from the Image Repository and then initializing instances accordingly. Through

execution of action plans received from the Elasticity Engine, the cloud middleware adapts the system to the change detected by the Monitoring component. (iii). The *Monitoring* component keeps track of the state of the whole system by collecting metrics from different layers including the infrastructure layer, the cloud middleware layer, and the application layer. Collected information is used by the Elasticity Engine for decision making. (iv). The *Elasticity Engine* continuously pulls system metrics from the monitoring component. By analyzing the received data, it produces action plans accordingly to optimally fulfill the application requirements.

3.1 An Implementation

Figure 2 presents an implementation of the proposed approach.

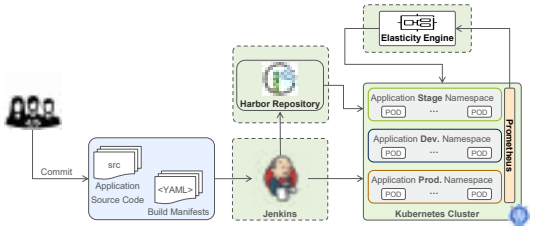


Figure 2: An implementation of the proposed approach.

Particularly, we choose Harbor [7] as our image repository. The main reason is that, in addition to basic functionalities that are also commonly supported by other alternatives, it has two significant features, i.e., *vulnerability scanning* which scans image regularly and warns users of vulnerabilities, and *notary* that ensures image authenticity. We believe these features are critical to the security and robustness of the big data applications. Kubernetes as our choice for the cloud middleware component, provides a robust and multi-tenancy support environment for hosting containerized applications. Its rolling updates feature allows deployment update to take place with zero downtime by incrementally updating instances with new ones. This is very important as users do not want to bring down the whole application during upgrade as mentioned previously in Section 2. More importantly, we allow three copies of each application co-located in our system. They are isolated through namespace. Each copy corresponds to its different phases in deployment, including development, staging, and production. By doing so, our system offers seamless transitions between deployment phases for applications, and provide fast delivery. We integrate Prometheus [6] as the monitoring component in our framework. Our current implementation of the Elasticity Engine is a set of predefined *if-then* rules consisting of logical conditions and corresponding actions. A full-fledged implementation is part of our future work.

4 Case Study on Large-scale User Traffic Data Processing

In this section, we present a case study of an online shopping traffic processing application, which extracts and visualizes insights by collecting and analyzing the real-time shopping log. The application is composed of three main components, including Kafka [4] for

handling real-time data feeds, Spark for data processing, and a dashboard built on Flask for visualization. Data is streamed into the Kafka cluster through multiple Kafka producers that continuously read logs from an experimental dataset [1], which contains anonymized users' shopping log collected by Tmall.com[1] between May and November in 2015.

The first task is to containerize each component in the application. This is typically done by building container images base on manifests that include the component specifications. Having all components containerized, components in the application are then deployed as services. Each service is backed by a set of container instances, ensuring high availability.

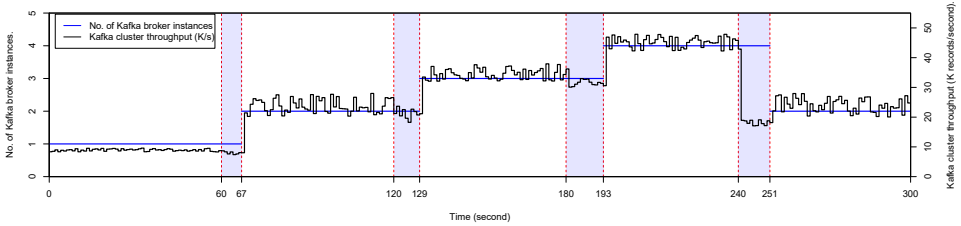


Figure 3: Elasticity control on Kafka cluster to adapt the workload changes.

In our study, we examine the behavior of our system through workload variation. We expect the components (Kafka and Spark) in the data pipeline to scale accordingly. Figure 3 presents the statistics of the Kafka cluster during a 5-minute experiment where the workload is adjusted every 60 seconds by changing the number of Kafka producers. Initially, we have only one Kafka broker instance backing the Kafka service. As the cluster is stressed, the number of Kafka broker automatically instances increases in the 60^{th}, 120^{th}, and 180^{th} second, and then decreases in the 240^{th} second after the peak.

However, as we can see in Figure 3, the number of Kafka broker instances does not change immediately as the workload varies. There is always a lag of a few seconds. Furthermore, the throughput of the cluster slightly goes down until the newly (de)commissioned instances are ready. The reason behind is that, when scaling, Kafka does not automatically share the load of existing partitions on other brokers. To redistribute the existing load among brokers, partitions are reassigned, resulting in performance degradation. This does not apply to the Spark component as elasticity control on Spark instances does not require data redistribution. Despite this lag, we can still conclude that our framework can provision the application efficiently. We also verify the effectiveness of our system in supporting version control and multi-tenancy respectively by allowing component instances to have different versions and duplicating the full application. In each scenario, our system is able to provide similar results as we presented above.

5 Related Work

Both microservices and provisioning of big data applications have recently received tremendous attention. There is a vast amount of research on each of these two topics, such as [5] for microservices management in cloud, and [3] for provisioning of big data applications. However, to the best of our knowledge, existing literatures rarely combine them together and address the big data application provisioning challenges using microservices methodologies.

The importance of automation of all stages of the big data applications development, deployment and management is discussed in [2]. In regards to the aspect of configuration, Zhang et al. highlight the configuration complexity in cloud-based analytics environments, and present an engine to alleviate this issue through recommendation based on a dedicated k-nearest neighbor algorithm [8]. The scope of their work is limited on job level, while our work is more focusing on component configurations.

6 Concluding Remarks

In this paper, we present a microservices-based approach that allows creating and managing big data applications through a unified environment. Application components are containerized and deployed as services. Big data applications are then provisioned through orchestration of multiple component services.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under the grant No. 61672042 and 61602437, in part by the Beijing Natural Science Foundation under grant No. 4162021, and in part by the Research Funding of North China University of Technology.

References

[1] AlibabaCloud. IJCAI-15 Dataset. `https://tianchi.aliyun.com/datalab/dataSet.htm?id=5`, visited Jan 2018.

[2] Demchenko et al. Cloud Computing Infrastructure for Data Intensive Applications. In *Big Data Analytics for Sensor-Network Collected Intelligence*, pages 21–62. Elsevier, 2017.

[3] Herodotos et al. Starfish: A Self-tuning System for Big Data Analytics. In *Proceedings of the 2011 biennial Conference on Innovative Data Systems Research (CIDR'11)*, pages 261–272, 2011.

[4] Jay et al. Kafka: A Distributed Messaging System for Log Processing. In *Proceedings of the NetDB*, pages 1–7, 2011.

[5] Karwowski et al. Swarm Based System for Management of Containerized Microservices in a Cloud Consisting of Heterogeneous Servers. In *Proceedings of the International Conference on Information Systems Architecture and Technology*, pages 262–271. Springer, 2017.

[6] The Prometheus Project. Prometheus: From Metrics to Insight. `https://prometheus.io/`, visited Jan 2018.

[7] VMware. Harbor: An Enterprise-class Container Registry Server based on Docker Distribution. `http://vmware.github.io/harbor/`, visited Jan 2018.

[8] R. Zhang, M. Li, and D. Hildebrand. Finding the Big Data Sweet Spot: Towards Automatically Recommending Configurations for Hadoop Clusters on Docker Containers. In *Proceedings of the 2015 IEEE International Conference on Cloud Engineering (IC2E'15)*, pages 365–368, March 2015.

A Case Study of MapReduce Based Expressway Traffic Data Analysis and Service System

Zhilong Hong
School of Software and Microelectronics
Peking University
Beijing, China
hongzhilong@pku.edu.cn

Tong Mo
School of Software and Microelectronics
Peking University
Beijing, China
motong@pku.edu.cn

Weilong Ding
Data Engineering Institute
North China University of Technology
Beijing, China
dingweilong@ncut.edu.cn

Jian Zhang
School of Economics and Management
Beijing Information Science and Technology University
Beijing, China
zhangjian@bistu.edu.cn

Weiping Li
School of Software and Microelectronics
Peking University
Beijing, China
wpli@ss.pku.edu.cn

Haochen Li
College of Science
China Agricultural University
Beijing, China
18811773968@139.com

***Abstract*—The scale of expressway information networking is constantly expanding. Currently the existing analysis system is still built on the relational database. Traffic data produced by the system has reached a data volume of 3 million items monthly. The performance requirements, including high concurrency, massive throughput, visualization, and scalability, are difficult to be satisfied. The Expressway Traffic Data Analysis System (ETDAS) is designed to meet the needs of the collection, analysis and visualization of increasing expressway traffic data by means of the distributed frameworks. The new system is expected to help regulate the road network traffic flow, reduce traffic congestion, and provide analytical support for the optimization strategy of road network. ETDAS has been deployed online.**

Keywords—Expressway Traffic Data Analysis System, Big Data, Hadoop, MapReduce, Data Visualization

I. INTRODUCTION

In past few years, the expressway has formed a constantly expanding regional road network structure. Expressway information system has covered the main business areas of traffic management and accordingly precipitated large amount of traffic-related data such as videos, pictures, charts, and texts. These data are characterized by large quantities, high dimensions, multiple sources, and heterogeneous formats. Most of these data exists in several separate applications. These data are not fully integrated and utilized, which results in the fragmentation of traffic management.

Our existing expressway traffic data analysis system is built with relational database. The system has poor performance on real-time analysis and low efficiency for mass data processing. What makes it worse is the incapability to deal with the increasing data volume and problems like congestion and slow running. Traditional relational database is incompatible with various types of data. Furthermore, the lack of intuitive data visualization leads to tough decision-making. A brand-new system is urgently needed.

As for other existing systems, Guangzhou province has launched the construction of expressway information system to aggregate data from different departments, but the analysis and visualization of the data is still on the way[1]. Qi Shi and Mohamed Abdel-Aty introduce several models to measure and predict the congestion of expressway in Orlando [2]. Their work focuses on rear-end crashes only, and using single factor in the model leads to a bias towards the real situation. Ari Wibisono et al. use fast incremental model trees with drift detection to predict and visualize the traffic data during the 5-year in motorways in UK[3]. Their prediction is performed on a single server node, thus it's hard to scale up the computing power.

The Expressway Traffic Data Analysis System (ETDAS) is designed to reduce traffic jam, regulate the traffic flow and improve the road capacity. The main function of ETDAS is to collect, analyze and visualize the expressway traffic data. With big data analysis techniques like Hadoop, the system can find correlations across the massive data, excavate the value in the correlations, and predict the future trend of traffic situation. The statistics of traffic data like toll amount and traffic flow can be used to find the reason of traffic congestion and make suggestions to the actions to minimizing the impact of the congestion. The system visualizes the traffic data to display the traffic flow and ratio of different types of vehicles.

ETDAS should meet the requirement of real-time response, high performance, safety and reliability. The interface of the system should be user-friendly and easy to use. The modular structural design with high cohesion and low coupling will guarantee good extensibility and scalability. What is the most important for ETDAS is the ability to search speedily in

petabyte-level multi-dimensional data and organize the data into different kind of statistical results. The data contain structured, semi-structured and unstructured data such as toll collection data, surveillance video, government data and maintenance logs. There are traffic factors of congestion including the overall status, affected areas, temporal and spatial features, special sections like crossroads and frequently congested roads[4]. Considering the factors above, expressway staffs can get full knowledge of the expressway's status and take actions accordingly with the help of ETDAS.

II. System Architecture

ETDAS consists of reusable generic service components intended for big data analysis. The components support high availability, horizontal extension, and distributed big data storage. A variety of data analysis models are embedded, and multi-dimensional data visualization is supported. The system implements the separation of business procedure and data, which improves the reusability of the components, and satisfies analysis requirements of both transactional and analytical data[5].

ETDAS complies with relevant technical standards and specifications. The system is composed of data access layer, data storage layer, big data analysis layer, and application layer.

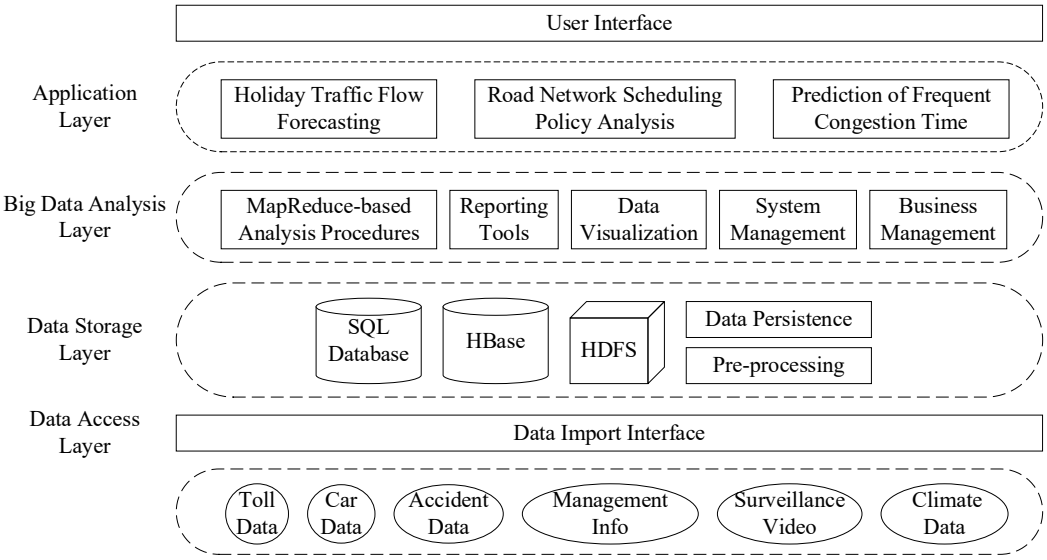


Fig. 1. System architecture of ETDAS .

Data access layer: this layer realizes the integration of various business data such as traffic flow, toll collection data, meteorological data, road events, road maintenance logs, etc. These business data are provided by different core business systems, such as the integrated traffic management platforms and the expressway toll systems. More and more data will be collected by the system. In addition, relevant data import interfaces are available for other external departments, such as transportation management departments, public security bureaus, medical system, and fire stations.

Data storage layer: this layer stores and manages data imported from the data access layer, including structured data like traffic flow data and unstructured data like surveillance videos. It provides support for massive data storage and performs data preprocessing and integration. Data are store in distributed clusters and can be retrieved by business logic and management needs.

Big data analysis layer: this layer consists of a big data analysis engine to provide distributed computing support for various types of data mining algorithms. It also provides common service components like reporting tools and visualizations to provide services for other components and layers. Expressway management components like system management and business management are provided, too.

Application layer: this layer provides multi-level, comprehensive data analysis application and decision-making services for expressway traffic big data. The data analysis application contains operations like statistics analysis, query, report, warning, forecast, optimization and decision-making on road network traffic data. Holiday traffic flow forecasting, road network scheduling policy analysis, and prediction of frequent congestion time is the key applications in the system. Furthermore, it provides a user-friendly interface for various end-user devices like computer, cellphone, etc.

ETDAS uses a variety of big data processing and visualization frameworks, such as Hadoop and ECharts. And we adapt these frameworks to meet the business needs of the system. To collect data, we adopt a data bus to covert different types of heterogeneous data into a unified form, and a distributed data input engine to solve the issues in dealing with high concurrent input. According to the characteristic of different types of data, we store them into different places. The collected data like traffic flow data and toll collection data is stored into HBase, because HBase can handle high concurrent input[6]. The data produced in analysis procedures is stored in the SQL database, which is convenient for the front-end to retrieve the result of analyses. Multidimensional data indexes are constructed to enable fast data query. Non-structured file data like videos and photos is directly stored in HDFS. For small-size files, they will

be merged when store into HDFS and HBase will maintain their file directories and indexes. Finally, we use the ECharts framework to visualize the analysis results.

III. DATASET

In this section we discuss the details of the dataset. The data in EDTAS has following characteristics:

- **Different data sources.** There are many types of data in EDTAS, such as structured data collected by toll systems, unstructured data like pictures and videos generated by road monitoring systems, in addition to details of facility maintenance events, traffic control, toll stations, sub-centers information and other information related to the highway.
- **Massive real-time data.** During the daily operation of the expressway, data of different update frequencies will be generated. For example, the update frequency of the toll gate data is 10 seconds/time, and the update frequency of the expressway event data is 10 minutes/time. The high update frequency of data and the large scale of data sources will further lead to a rapid increase in the amount of data. Toll stations can collect 840,000 vehicle information data items and 2.89 million of toll collection data items per month.
- **High efficiency.** The data analysis system must have high computational efficiency because traffic data preprocessing and congestion recognition must be done within a few hours so that expressway staffs can take action in time. All applications all have certain timeliness requirements [7]. Therefore, it is necessary to adopt big data frameworks to improve the speed of data collection, storage, transmission, processing, and analysis.

Aside from toll fee, the expressway toll stations also collect information from each passing car. The toll information system transmits the attribute of each passing car to the settlement center promptly through the private network. This data forms an expressway traffic information database. Also, data in the database is valuable for expressway management[8]. It contains large amount of information, such as traffic flow status, traffic volume ratio, axle load conditions, distribution of the traffic flow in each road, and total number of cars passing by the toll stations. There are nearly 60 fields in each data item of the raw toll collection data. The main fields include inbound/outbound stations, inbound/outbound times, vehicle type, axle weight, number of axes, driven distances, amount of toll fee, as well as ratio of overloaded cars. Based on these data, we can accurately calculate real-time traffic information data such as traffic flow, travel time, driving speed, OD traffic, congestion, traffic events, and characteristics of operation[9]. With the help of traffic flow allocation models, these data can be further analyzed[10]. The name, source and update frequency of traffic data are listed in the table 1.

TABLE I. NAME, SOURCE, AND UPDATE FREQUENCY OF THE TRAFFIC DATA

Name	Source	Update frequency
Expressway full road network data		
Traffic flow data	SQL database	Every five minutes
ETC/MTC data		Every day
Type of cars		Every day
Regional data		Every hour
Expressway toll station data		
Amount of toll fee	SQL database	Every five minutes
Type of cars		Every day
ETC/MTC data		Every day
Regional data		Every hour
Number of passing by cars	Access tables	Every ten seconds
Inbound/outbound car ratio	SQL database	Every five minutes
Information of roads	Excel files and SQL database	Every month
Information of sub-centers	Excel files and SQL database	Every month
Maintenance log	Excel files	Every day
Traffic event log	Excel files	Every ten minutes
Information of toll stations	Excel files	Every month

We take the ETC/MTC (Electronic Toll Collection/Manual Toll Collection) data as an example to illustrate the details of the data. The data records the information of ETC/MTC vehicles at each toll station of the expressway, including the name of toll stations, date and time, type of ETC/MTC (1 for ETC, 2 for MTC, and 3 for abnormal data), the traffic volume of different types, etc. These data can be used to find potential ETC users as well as the ETC/MTC usage of specific vehicles. The update frequency of the ETC/MTC data is once per day. Every month 28,412 data items are accumulated in total, involving 33 expressway toll stations.

The import and processing procedures of the ETC/MTC data of expressway toll stations are as follows：

1) Import data from SQL relational database into HBase;
2) Upload the config file of Hadoop;
3) Perform data preprocessing procedure;
4) MapReduce statistic procedure: According to the business needs of multi-dimensional statistics, various MapReduce jobs are performed to do traffic statistics, and the results are stored in the corresponding database of HBase;
5) Store the statistical results into SQL database： export the statistical result from HBase into SQL database.

Structural source data from toll system are stored in HBase, while non-structural data like videos and photos are stored into HDFS. Statistical data produced by analysis procedure along with data from other business systems are stored into SQL database. The data in SQL database can be retrieved by visualization modules to display the result of analysis to expressway staffs.

IV. DATA PROCESSING

A. Preprocessing

Data processing procedure mainly includes two parts: preprocessing and statistical analysis. There may exist problems such as invalid data, null data and redundant data in the expressway network operation data. Therefore, the data preprocessing module needs to determine whether the data is

invalid or not and attempt to correct the invalid part. If the data cannot be corrected, the module needs to log it for later analysis. The input of the data preprocessing module is the original source data, and the output is the data that can be used for further analysis after preprocessing.

During the procedure of judging whether the inbound/outbound datetime are valid or not, it is necessary to test whether the datetime is a null value or a unreasonable datetime (such as exceeding the maximum number of days in the month, ignoring the leap year, etc.). If the datetime is fault, the procedure will try to guess the correct datetime and set it right; if the datetime cannot be corrected, the error will be recorded in the log and the next item will continue to be processed. After getting the correct date and time, we also need to test whether the datetime is within the specified time range. Only the data in the specified time range will be stored in HBase. The preprocessing procedure can be illustrated by following pseudo code:

```
function DataPreprocessing
input: raw string of traffic data
output: the datetime, station and other values
function DataPreprocessing:
    extract datetime, station and data from raw string
    if both of incoming datetime and outbound datetime are
invalid:
        log the error and continue to deal with next item
    else:
        if one of inbound datetime and outbound datetime is
invalid:
            try to fix the invalid datetime
            if get error or the datetime is still invalid:
                log the error and continue to deal with next item
            end if
        end if
        if both the datetimes are in specified time range:
            write the datetime, station and value into database
        end if
    end if
end function
```

B. MapReduce-Based Statistical Analysis

The frequency of toll collection data of expressway is 10 seconds. There are totally 33 toll stations in the province, and the monthly data volume can reach 3 million. For these data generated in the expressway networking operation, MapReduce-based statistical analysis will efficiently calculate and store the result according to business needs. The mapper nodes extract data from the incoming string containing datetime, station name, and detailed traffic information. The datetime and station name are used as key, as well as the value of each pair is set as 1. Then the mapper nodes output key-value pair and pass to the reducer node. The reducer nodes receive the input key-value pair, count all pairs according to the key, and finally output the statistical result as a key-value pair to the persistence module[11]. The entire system has 14 categories of traffic data analysis procedures classified by stations, time periods, type of cars, normal/ETC/MTC, etc. The data obtained from these analysis procedures will eventually be stored in the SQL database for visualization. We take the analysis procedure of daily traffic statistic of ETC/MTC cars in the entire road network as an example to illustrate how the system process the data.

```
function Map:
input: raw string containing datetime, station and other data
output: the (key string, 1) tuple to the context
function Map:
    extract datetime, station from input value
    serialize the datetime and station as the key string
    output the (key string, 1) tuple to the context
end function
```

```
function Reduce:
input: the iterable (key string, 1) tuple lists
output: the (key string, sum) tuple to the context
function Reduce:
    sum := 0
    for each tuple in tuple lists:
        sum := sum + 1
    end for
    output the (key string, sum) tuple to the context
end function
```

We simulate the real situation on a cluster of 3 nodes, and each node is equipped with i7-6700K CPU and 12 GB memory. Currently, ETDAS based on MapReduce can perform statistical analysis of 50 million data on the cluster within 40 minutes. As for the old system, it needs a few hours to complete the statistical analysis. MapReduce on clusters is more efficient than traditional system for large-scale data.

C. Data Visualization based on ECharts

For expressway staffs, the analysis result need to be intuitively displayed for decision-making[11]. How to present the data on the end-user devices effectively is a challenging task[12].

Fig. 2. Regional traffic flow map .

ECharts is a visualization framework based on JavaScript. It provides rich visualization types, supports multiple data formats, and data can be directly displayed without conversion. With incremental rendering technique, ECharts can provide a real-time presentation of millions of data. With a multi-rendering solution, ECharts has good support for the cross-platform[13].

The visualization module will display the result produced by MapReduce-based on the front-end of the management system. The charts include line charts, bar charts, pie charts, maps, etc. There are currently more than fifty charts in the system. Take the regional traffic map as an example, as shown in Figure 2. The traffic flow of each region in a province is shown on the map. And the colors of the areas indicate the difference of traffic flow between the regions. Regions with extremely large amount of regional traffic flow are highlighted on the map, and expressway staffs can refer to the traffic flow of surrounding regions to analyze the reason of the extremely large amount and take actions accordingly.

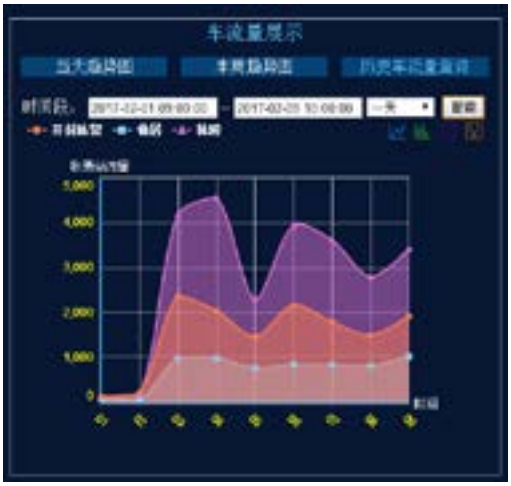


Fig. 3. Line graph to show the trend of traffic flow in multiple regions

The line graph can show the trend of traffic flow in the region over time. Users can input a time range to get the traffic flow of specified region in a certain time interval. And multiple regions can be selected so that their traffic flows are displayed in the same graph. This can be used to compare the similarities and differences among the regions, which are the reference when taking actions like congestion control and traffic control, as shown in Figure 3.

V. Conclusion

This paper provides a case study of big data in the expressway management. Because of the expanding scale of expressway road network and the increasing data volume, the total amount of traffic data has reached the terabyte level, even to the petabyte level. Big data frameworks are used to process the traffic data. With Hadoop, the system can find correlations across the traffic data, excavate the value in the correlations, and predict the future trend of traffic situation. Furthermore, the system can help to regulate the traffic flow and improve the road capacity. At present, the ETDAS has been officially deployed online and widely used.

Acknowledgment

This work is supported by Henan Provincial Department of Transportation Technology Project under Grant No.2016G5. The authors appreciate the help from Beijing Key Laboratory on Integration and Analysis of Large-scale Stream Data from North China University of Technology.

References

[1] Y. Du, and J. Jiang. "Big Data Background: Studies, Analysis and Forecasts of the Highway Toll System Data." Computer Knowledge and Technology 5X (2012): 3752-3754.

[2] Qi Shi, and Mohamed Abdel-Aty. "Big data applications in real-time traffic operation and safety monitoring and improvement on urban expressways." Transportation Research Part C: Emerging Technologies 58 (2015): 380-394.

[3] Ari Wibisono, et al. "Traffic big data prediction and visualization using fast incremental model trees-drift detection (FIMT-DD)." Knowledge-Based Systems 93 (2016): 33-46.

[4] P. Zhao, and K. Li. " A Theoretical Analysis for the Applications of Big-Data Methods for Traffic Congestion Relief." Modern Urban Research 10 (2014): 25-30.

[5] H. Zhang, X. Wang, J. Cao, and C. Zhu " Architecture of Intelligent Traffic Systems Based on Big Data." Journal of Lanzhou University of Technology 41.2 (2015): 112-115.

[6] A. B. Ayed, M. B. Halima, and A. M. Alimi. "Big data analytics for logistics and transportation." Advanced Logistics and Transport (ICALT), 2015 4th International Conference on. IEEE, 2015.

[7] S. Mark, *et al*. "Towards a multi-cluster analytical engine for transportation data." Cloud and Autonomic Computing (ICCAC), 2014 International Conference on. IEEE, 2014.

[8] H. Xu. " Discussion on Application of Big Data in Smart High Speed Traffic." China ITS Journal 03 (2016): 79-84.

[9] C. A. Quiroga. "Performance measures and data requirements for congestion management systems." Transportation Research Part C: Emerging Technologies 8.1-6 (2000): 287-306.

[10] Y. Jie, and J. Ma. "A big-data processing framework for uncertainties in transportation data." Fuzzy Systems (FUZZ-IEEE), 2015 IEEE International Conference on. IEEE, 2015.

[11] R. Yang, C. Lang, and W. Liu. " Current Status and Challenges of Highway Bigdata Processing." Computer Systems & Applications 23.9 (2014): 13-17.

[12] P. C. Wong, H. Shen, and C. Chen. "Top ten interaction challenges in extreme-scale visual analytics." Expanding the frontiers of visual analytics and visualization. Springer, London, 2012. 197-207.

[13] "Echarts". echarts.baidu.com, 2018, http://echarts.baidu.com/. Accessed 15 Apr 2018.

[14] H. Lu, Z. Sun, and W. Qu. " Big Data and Its Applications in Urban Intelligent Transportation System." Journal of Transportation Systems Engineering and Information Technology 15.5 (2015): 45-52.

[15] W. Qiu. " Intelligent Traffic Big Data Analysis Cloud Platform Technology." China ITS Journal 10 (2013): 106-110.

Approach the Cognitive Networks for Self-Adaptive Control Based on Service Awareness

Author: Mack J. Du

Address: Communication and Network Engineering Dept.
Shanghai Jianqiao University. Shanghai, China. 201306

Abstract: With internet technology quickly development, now information becomes explosive growth each day. Especially Internet of things applications is stepping into people work and life, the diversity services transmit on the communication network. A dynamic and smart network characteristic became a hot topic in academic field. This paper firstly studies multi services property by comparison way, secondly analyses the features for cognitive network and self-adaptive service control, then proposes the self-adaptive network architecture and intelligent service awareness model, by control theory finally designs a collaboration Mechanism in Self-Adaptive Control for service quality. The paper puts forward an innovative cognitive network and control mechanisms which not only intelligently adjust resource allocation but automatically adapt to a changeable network environment to ensure optimizing end to end network performance. This distinctive research idea and solution must be of useful reference significant to coming smart communication times.

Keywords: Cognitive network; Self-adaptive layered architecture; Service awareness model; Collaboration control mechanism; Service target.

References:

[1] Mack J. Du. Study for Information Connotation Evolution and Model, IEEE Xplore, 2017

[2] N. Economides, C. Himmelberg. Critical Mass and Network Evolution in Telecom. Telecommunication Policy Research Conference. 1994.

[3] C.Fortuna and M. Mohorcic. Trends in the Development of Communication Networks: Cognitive Networks. Computer Networks, 2009

[4] Clark D, Tennenhouse D. Architectural considerations for a new generation of protocols. Computer Communications Review,1990.

[5] J. Mitola and G. Maguire. Cognitive radio: Making software radios more personal, IEEE Personal Commun., vol.69, no.8, 1999.

[6] R. Thomas, Cognitive Networks. Blacksburg, VA: Virginia Polytechnic and State University, 2007.

[7] D. Clark, C. Partrige, J. Ramming et al., A knowledge plane for the Internet, in Proc.Conf. On Applications, Tech., Architectures, and Protocols for Computer. Commun. (SIGCOMM '03), New York, 2003.

[8] N.Baldo and M. Zorzi. Fuzzy logic for cross-layer optimization in cognitive radio networks, IEEE Commun. Mag., vol.46, no.4, 2008.

[9] M.Siebert. Self-X control in (future) mobile radio networks, Proc.European-Chinese Cognitive Radio Syst. Workshop, Beijing, 2008.

[10] F. Shao and Lifeng Wang "Cognitive network structure and approach based on cognitive level," J. Beijing Univ. Tech., vol.35, no.4, 2009.

[11] Thomas R, Friend D, Dasilva L, et a1. Cognitive networks adaptation and learning to achieve end-to-end performance objectives, IEEE Communications Magazine, 2006.

[12]Dakun, Dan. Cognitive network architecture and key technology, http://www.cnki.net/KCMS/detail/detail.aspx, 2012.

[13] M. Pitchaimani, B. Ewy, J. Evans. Evaluating Techniques for Network Layer Independence in Cognitive Networks, Proc.IEEE Int. Conf. on Commun. (ICC'07), Glasgow, 2007.

[14] T. Suda, T. Itao and M. Matsuo. The Biologically Inspired Approach to the Design of Scalable, Adaptive, and Survivable/Available Network Applications, The Internet as a Large-Scale Complex System, the Santafe Institute Book Series, Oxford University Press, 2005.

[15] B. Melcher and B. Mitchell. Towards an Autonomic Framework: Self-Configuring Network Services and Developing Autonomic Applications, Intel Technology Journal, volume 8, issue 4, November, 2004.

1 Introduction

Since IBM firstly providing new concept of Wisdom of the Earth, Internet technology and Internet of Things (IOT) technology get extremely fast development. Especially with people stepping into the twenty-one centuries, human absolutely comes into an information times. The exponential growth of ocean data and information explosion of society make scientists to unexpected. The theory research of modern communication network has lagged far behind times development and social requirement.

In particular, there are some obvious new features in network communication. 1) Diversity types of information. Today the multiformity of service information has covered various aspects, such as music, blog, searching, photo, television, news, game, e-mail, and film and so on. 2) Bearing objects variety. The transmission type on network includes very wide aspects, for example, picture, words, file, three division (3D), flash, video, multimedia, artificial reality (AR), and virtual reality (VR) etc. 3) Transmission quantity and time uncertainty. No one knows how many bit or Gage bit will produce in any line or section of network at any time. The transmission randomness and concurrency become normalcy in network [1].

All of above features in big data communication time have become a bottleneck to traditional network system. How to deal with this challenge in

existed network? How to promote the smart function in network that can automatically control transmission and management based on the service awareness? It has been key topic in the research field that needs to fix issues on modern network system. Up to now many researchers and scientists have stepped on this way for smart target.

This paper adopts comparative method to analyse and study this topic based on the cognitive characteristic of up-to-date communication network. Then it focuses on researching self-adaptive network architecture and intelligent service awareness model, Later the paper designs a coordinative control mechanism for quality of service (QoS). This paper proposes an innovative cognitive network and control mechanisms. It must be of important significance in theory reference and experimental application value.

2 Analysis for Network Feature and Smart Function

2.1 The existed network function

Generally, the existed communication network is composed of three layers architecture. This architecture system includes end-user layer, access layer and interactive network layer. The layer system diagram is as Fig.1. In this figure, end-user unit is information (i.e. signal) source and sink module group that is of sending and receiving two-way functions. Access layer unit is of information concentrating and disassembling features in wire and wireless network. These functions include but not limited to cell encoding and decoding, channel modem, and data encryption and decryption etc. In high layer, the interactive network represents data interactive transmission. This network can deal with real time and non-real time data service, work out the batch data transmission and cover long distance and wide space [2]. This interactive network could be but not limited to public telephone network, Internet, mobile network and so on. The operation and management (O&M) unit mostly focuses on maintaining functions such as accounting, configuration, fault, performance and safety management etc. All of these covers existed links, equipment, certain services and network daily maintenances.

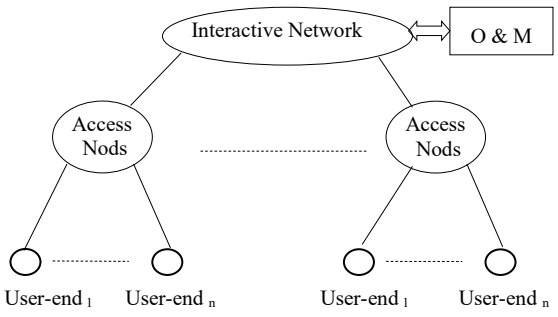


Fig.1 The existed network and management

As above section discussion, because of service type diversity and content quantity and time uncertainty, during stepping into IoT time, the existed network has several disadvantages:

- Difficultly to deal with service concurrency in dynamic situation.
- Without capability of known the service types and its features.
- Hard to adjusting traffic bandwidth based on actual application cases.
- Having rarely little perception ability to dynamic network environment.

And so on.

As C. Fortuna and his fellow discussion in Trends in the development of communication network on Cognitive Networks, at present, the network service types are varied, and network environments are complex and dynamic. Traditional end-to-end (e2e) insurance technology lacks intelligent inference and self-learning capabilities. Therefore, it cannot adapt to provide ideal service under dynamically changing network conditions [3].

2.2 The smart network function

From above section analysis, the most obvious issue is that the network and O&M system cannot recognize the much different service demands from multifarious end users. Furthermore, existed network cannot effectively and dynamically change service quality according to variations in the internal and external environment of the network system. Some academics have started to integrate cognitive elements from next generation networks (NGNs) into current networks in order to overcome these embedded defects. Consequently, the concept of cognitive networks (CNs) and smart function have arisen [4].

Because the mobile network and services have quickly infiltrated into every corner in recent decades, CN research is focused on the cognitive radio (CR). J. Mitola and his partner first put forward CR and the architecture of the cognitive ring. A CR system can obtain needed frequency spectrum through perception. It determines the reconstruction scheme of CR according to the optimization object and can adapt to changes in the frequency spectrum dynamical environment [5].

With further approaching on cognitive technology field, CNs based on CR is conceived by the Motorola and Virginia Tech. companies [6]. A CN has smart function for cognitive processes and perceives the current network condition. It perceives changes in itself and in the environment. It then makes plans and determinations and takes action based on these perceptions. The FOCALE architecture of dual close-loop control is also provided.

D. Clark and others proposed introducing Knowledge Plane (KP) to the Internet at CIGCOMM ‘03 [7]. The key point of this concept is that KP can perceive its own behaviour. It can analyse problems and adjust its operation to increase reliability and robustness.

In 2007, Baldo used fuzzy logic to effectively process modularization and inaccuracy in the CN [8].

Until 2008, one more detail smart function concept was provided by Siebert. He points out that the ability of a CN is greatly an important feature. The CN implement various tasks through autonomous self-management, self-optimization, self-monitoring, self-maintenance, self-protection, and self-healing functions [9]. That is also called as “6S “smart feature in field.

Later, C. Fortuna suggested that Thomas's definition of CN was incomplete. Knowledge expression and cognitive ring are the most important elements of the CN in 2009.

As advanced technology guide, the IEEE is currently discussing standardization on the integration architecture of isomerism wireless access networks. In these discussions, the concept of CN is applied. CN is considered as a new way of improving overall network and e2e system performance as well as simplifying network management. It is the main trend to the next-generation communication (NGC) [10].

From above discussions, there is network ecological ring from changing requirements to perceiving, from learning to response, final matching these demands. Thus, it is essentially necessary that the cognitive, perception, self-adjustment and feedback action functions are really a core of intelligent network.

2.3 The cognitive and self -adaptive characteristics

CN is a new research area and has just taken its first steps in China and in other countries. Therefore, relevant theories and techniques need to be further studied.

From network architecture point of view, the cognitive functions of a CN are implemented by distributed intelligent agents based on AI technology. Agents with learning and reasoning capabilities are deployed on each node in the network to monitor and collect environment information. These agents cooperate and exchange information so that the network can perceive its current status. End-to-end targets can be achieved based on the network status, and network resources can be evaluated, predicted, planned, adjusted, and allocated based on a knowledge library. As a result, the network has self-perception, self-learning, self-optimization, self-healing, and self-configuration capabilities. It can be measured, controlled, managed, and trusted.

As far as the service quality control architecture of the CN, this paper proposes one of key concept is the network's ability to perceive changes in the entire CN environment and automatically adjust itself in real time.

Hence the cognitive characteristics are briefly summary as following:

- With smart cell or brain to perceive dynamic environment.
- Being of capability to monitor and collect service and network data such as e2e service target, network parameter, and usable resources etc.
- Having exchange function in different cell nodes for various data.
- Possessing analysis ability for current statue and trend statue.
- Being able to automatically re-configure network and allocate resource.
- Various functions deploying in distributed nodes or layers of entire network.

Usually self-adaptive feature is as a key characteristic in cognitive cycle for conducting feedback action. In communication network, self-adaptive control technology can plan and allocate limited network bandwidth effectively so that network performance is improved [11]. The technology also manages and controls network traffic according to service features in order to improve the revenue of unit bandwidth. Therefore, intensive self-adaptive control is essential to solve network service quality problems in CNs.

In regard of self-adaptive meaning, that must cover the below several characteristics. 1) Learning the user's requirement, target and change, and adjusting relevant means. 2) Self analysing the various data from monitoring, and conducting corresponding action.3) For the variety and expansion of network types and topologies, allocating the resources to meet that in time. 4) Self configuring relevant parameters and protocols in dynamic traffic service situation. 5) Usually matching the trend statue based on the monitoring and awareness ability.

3 Research for Key Techniques of Cognitive Network on Service Target

3.1 Constructing for cognitive network architecture

In order to approach the cognitive network, this section focuses on the key technology of constructing network. The paper provides the concrete structure of a CN which mainly concerns "Target Cognitive and Smart Adjustment" based on the idea from R. Thomas. Furthermore, this architecture has distinctive point that more concentrates on dynamic perception, transition as well intelligent adjustment based on the target awareness. Hereon, the architecture of cognitive network consists of three layers as shown in Fig.2.

The first layer is target layer which is composed of many "e2e target "units. The target layer reflects the target demands which are put forward by the various applications, users, or resource needs and so on. The e2e target is designed the central service to meet the QoS requirements of each service. That drives the all behaviours in entire architecture.

By Cognitive Specification Language (CSL), the targets are mapped to specific mechanical demands and fed back to one or more relevant CN elements. In here, this CSL not only is adaptive for new network element, application program and target, but supports distributed and centralized operation. In this way, the concrete actual target requirements from end-user are transferred to NE target by standard CSL. This lays the essential foundation for cognitive feature in network.

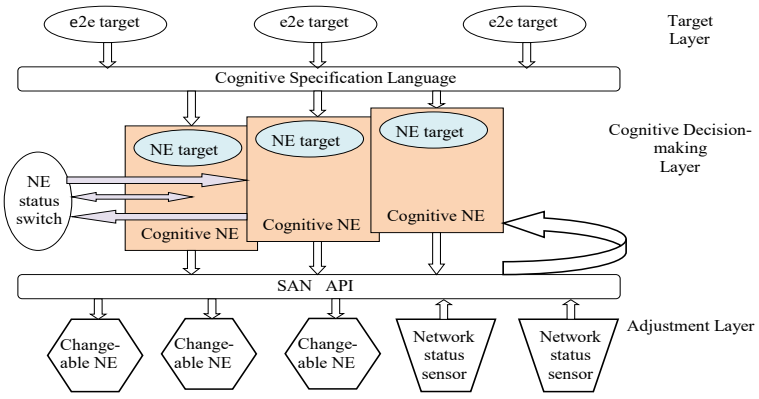


Fig.2 The architecture of cognitive network layers

The second layer is cognitive decision-making layer that contains a lot of cognitive NE. This layer has three main functions. Firstly, it perceives the current network status according to the demands of the target layer through CSL. Meanwhile it puts relevant NE target transferred from CSL into corresponding relevant cognitive NE. Secondly, it implements status switching of network elements (NEs) that match the target NE in dynamic service environment. Thirdly, it makes the decision for configuration and obtains the NE configuration through certain neural network method. This working mechanism is learning and reasoning process as well.

Between the cognitive decision-making layer and the adjustment layer, there is a smart middleware interface which includes application programming interface (API) in software adaptive network (SAN). SAN provides the suitable NE interface; furthermore, the API is of flexibility and expansibility. Meanwhile the SAN conveys the monitoring statue from the third layer to process layer. This API is mainly applied as information transferring interface.

The third layer is the adjustment layer. That is also called the adaptive network layer. The third layer consists of some changeable network elements, network statue sensors and SAN API etc. Its work procedure is these steps. 1) The decision from the cognitive decision-making layer is sent to the corresponding entity NE through API. 2) By adjusting the configuration of the NE, the entity NE becomes a changeable NE. Moreover, this changeable NE completely meets the demands from the target layer. 3) At the same time, the third layer feeds the network updated status through a monitoring sensor back to the second layer which is as base for next round judgment and decision. 4) It just is these accumulated and adjusted data forming rich database which makes the system has stronger smart function in dynamic learning and adjusting mechanism. It is noticed that statue sensors distribute in various nodes. These sensors work in coordinative form among them, and their collaboration degree is dynamically changeable with network environment.

By this cycle operation from demand, target, switching, decision as well as adjusting NE, consequently the architecture of cognitive network realizes the Self Adaptive Function.

3.2 Analysis for context perceiving technique

As far as cognitive function, the context perceiving is necessary to improve cognitive performance. It focuses on how to observe context information change, and conducts automatic reconfiguration based on these changes. The foundation of CN is the rapid perception of existed network environment such as QoS, traffic speed, route parameters and so forth. A CN needs to observe current network environment and information in appropriate time. The information is used for planning and decision making in later [12]. By this data, the CN determines whether the current network meets user demands and target. If yes, CN conducts relevant transmission and management for end-user application. Otherwise CN adopts a suitable reconfiguration method to adjust related specifications up to meeting user's requirement and target.

In term of information, environment information perceived by the CN includes network type, network topology, available resources, interface protocols, and network traffic etc. All of which affects e2e transmission performance.

This paper emphasizes the context perception is an important way to improve network intelligence. There are three working procedures. 1) It determines changes in context information then adjusts CN itself accordingly. 2) When the network environment dynamically changes, the network makes relevant self-adjustments. 3) This self-adjustment adopts a reflection mechanism and a policy mechanism. From the cognitive policy definition, the network can pre-define an adjustment method when the context changes including environment, network and information etc.

The feature function of context perceiving infiltrates in the various units in the cognitive decision-making layer and adjustment layer. That better consolidates the entire network smart feature.

3.3 Designing for cross-layer coordinative operation

Because of traditional network, such as OSI or TCP reference model, badly limiting non-interfacing layers communication, the essence of cross-layer coordinative operation is to break the frame of the current network defect to share parameters in different sub-layers. Thus, the coordinative operation merges and coordinates various useful data to maximumly optimize the property of entire network. In this operating design, the status parameters and QoS parameters in the system resources are transmitted in the protocol layer. As a result, a joint design combining various protocol layers is achieved. Finally, the system fully utilizes the resources in order to provide better target for end users.

Following above explanation, the work operating flow cover these procedures. The first step, that various parameters are conveyed to cross protocol layer stack. The CN, in next step, is to adjust the relevant NE protocol stack or protocol layer parameters on the basis of CN information. In this way, the CN ensures users receive high quality e2e service performance. The third step, the cognitive processing layer knows the status of network layers and determines proper actions according to an optimization algorithm. In the final step, the third network layer makes the reconfiguration of network parameters and protocol stacks to achieving e2e communication.

In this paper, it applies the explicit cross-layer design method that sets the two-way communication between interfacing layers and creates one-way transmission among non-interfacing layers. It is the most benefit that methods both considering the current network layer statue and converging cognitive network with coordinative function. As a result, system fully makes use of resources and realizes coordinative smart feature.

3.4 Studying for reconfiguration scheme

Because of the increasing complexity of environment in recent year, if the network is not meeting the e2e demands of users, the CN can focus on the relevant NE and adjusts its protocol stack parameters to meet these demands. The adjustment process is the reconfiguration of the network [13]. In entire cognitive architecture, the CN more emphasizes the e2e target, and it should provide e2e re-configurability. It is difference that the software radio technology is only limited to reconfiguring the terminal, while CN involves all layers of the NEs and protocol standards that a stream passes through. It is a scheme with foresight that ensures service quality targets are met. Hence much more factors are considered in e2e reconfiguration.

The cognitive function realization in CN is mostly based on reconfiguration of the NE. The reconfiguration process can also be implemented by software, but the technical level of this reconfiguration is higher. That covers many aspects such as terminal reconfiguration, network reconfiguration, protocol reconfiguration, resource recon- figuration and service reconfiguration and so forth. The configuration of these is not limited to a single node. Multiple NEs on the e2e path are covered. This is called End-to-end Reconfiguration (E2R). The complexity and importance of E2R is greater than terminal reconfiguration.

In this research, the adjustment layer in CN architecture achieves this reconfiguration function by SAN API unit and changeable NE unit.

4 CN Control Architecture Based on Service Awareness

4.1 Analysis and design for service awareness model

As section one discussion, in recent years, many new applications have emerged including peer to peer (P2P) networks, VoIP, streaming media, social media, interactive online games, and VR etc. All of these appearances in new services aspect are heavily impacting the traffic model, application mode and service model. In particular, the rapid development of P2P has caused explosive growth in traffic. Moreover, unlimited bandwidth usage has extremely increased the burden on the current network.

As a result, network congestion has become more serious. The simple network expansion does not meet the requirements of daily increasing services. Therefore, it is the best way to adopt CN technology for intuitive perception, analysis, determination, and control transmission service.

That means the service directly drives CN development. The network system intuitively perceives services on the network including end user service status and NE service status [14]. Hence the intuitive perception and classification based on the service stream becomes the foundation which is for service-centred resource configuration, route adjustment, and dynamic self-adaptive traffic control. In service-aware technology, before the CN is introduced, the traditional static port method, payload feature method, and stream statistical feature method are used. These methods are merely effective for perceiving regular services; however, they cannot perceive many new services accordingly.

In order to overcome these disadvantages, this paper focuses on cognitive function approach as above section introduction. The network has intelligence, analysis as well as decision-making capabilities. In this paragraph, this paper provides innovative Service Awareness Model (SAM) based on an integrated feature. This SAM is mainly for perceiving services intuitively and intelligently in real time. The relevant detail SAM construct diagram is shown as Fig.3.

In this SAM model, there are two layers that one is traffic identification layer and other is protocol analysis layer. The former is composed of various identification engines (IE), matching units and feature units. While IE again includes port matching IE, traffic feature IE, connection model IE, Topology feature IE and protocol analysis IE etc. All of these IEs bear to distinguish and judge the relevant situation and data in each specific service aspect. Similarly, the

matching and feature units cover a series of specific service models such as topology, known application, and matching etc. Their main functions are to conduct related action and adjustment according to the cognition of various service identification engines. The other layer includes different protocol analysis units for example data link layer protocol analysis (PA), IP layer PA, and TCP layer PA etc. which more concentrates on analysing the statues and details of different layer protocols.

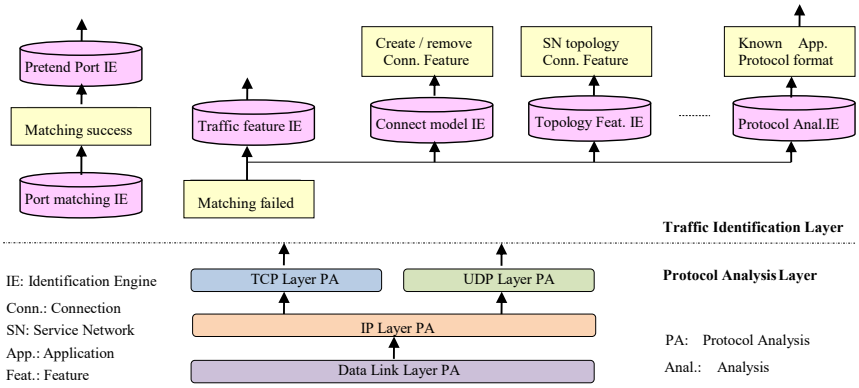


Fig.3 The intelligent service awareness and classification model

The model working mechanism is as following procedures. 1) After the model has obtained regular parameters of the CN, it constructs an integrated feature identification model which focusing on the traffic statistics feature, connection feature, topology feature, and content feature. The main purpose is to perceive and identify the various parameters as base for reallocating various resources accordingly. 2) The model sets an identification engine for each feature. It intelligently triggers and perceives different identification engines according to policies. 3) After above process, the model can accurately and efficiently identify these known or unknown, encrypted or plain text traffic. Finally, this intelligent cognitive model based on integrated feature accordingly
distinguishes known or unknown, encrypted or plain text services.

All of these lay the technical foundation based on service awareness for CN self-adaptive control architecture.

4.2 Constructing for layered service control architecture

From control point of view, in order to ensure conducting self-adaptive function in different aspects, this paper designs one layered scheme in entire system. The cognitive network with service decision-making and control function has a three-level structure. This structure is composed of NE (device) cognitive module, autonomous domain cognitive server, and central cognitive server as shown Fig.4. In these three layers, each part provides cognitive capabilities covering self-awareness, self-learning, and self-decision making.

The first, the NE (device) cognitive module is the basic unit of the CN for service awareness, analysis, and control purpose. It provides awareness and decision-making capability. Meanwhile this part can dynamically adjust NE

parameters or configuration. In here, several NEs and end-user devices (deployed with cognitive module) form a cognitive autonomous domain. Furthermore, this domain is hosted and configured by high level domain cognitive server. This domain is responsible for managing and controlling the NE device, service traffic, and network resources.

The second, the paper sets a central cognitive sever in the architecture. This server is main responsible for monitoring, awareness, and management on the running status in entire network. By this layered structure design, it is mainly to reduce the load on the central cognitive server. Even if the server temporarily fails in some time, that does not affect service QoS guarantee and management throughout the entire network.

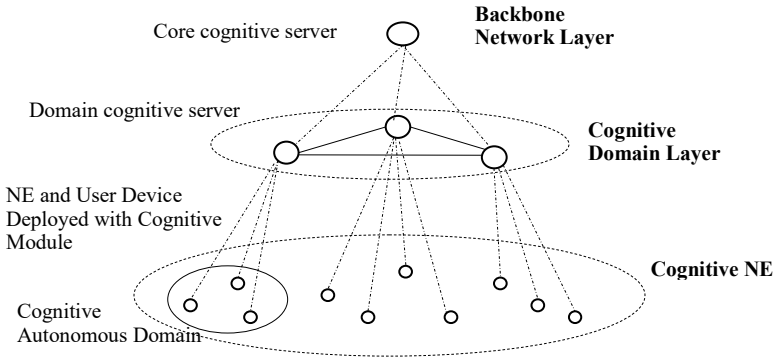


Fig.4 The self-adaptive control layered architecture of CN

The third, by distributing network way, it can realize real-time communication and exchange information between autonomous cognitive servers. The reason for using distributed management in the domain cognitive servers is to increase system reliability, flexibility, and expansibility. In an autonomous domain, the adjacent nodes can not only communicate among them, but complete distributed cooperative monitoring and self-adaptive processing.

This paper offers the three-layer architecture which integrates both the features of centralized construct and distributed processing technology. It has obvious advantage in networking and operating aspects.

4.3 Building the Collaboration Mechanism in Self-Adaptive Control

As above 3.1 section introduction, the NE is as important role in CN architecture. The cognitive NEs guarantee the e2e service quality in entire CN. Because there many cognitive NEs in cognitive layer, the working style of cognitive NEs is cooperative or independent [15]. The NEs firstly perceive the network condition in real time, then bring the trends together, and later analyse the current network environment. Finally, they configure themselves based on existing policies for achieving e2e service targets. In this way, it achieves the self-adaptive adjustment function.

In the following step, the paper focuses on setting combination control in path and port. Based on the service awareness, resource appointment concept

and control theory, the system integrates the service source-end QoS control and link QoS control in the CN. At same time, the paper proposes self-adaptive control mechanism in the collaborative port and path policy-based. This control mechanism of collaborative port and path mostly solve the issue of e2e quality guarantee for service traffic.

The first, the mechanism sends real-time network parameters to the autonomous domain server (or central cognitive server) through a feedback control. As a result, the self-adaptive service control mode is integrated into the terminal NE and routers. The second, that compares the history of network condition and the current condition to form a control policy to update the policy library through self-learning. At this point, the control policy gets optimal. Then the mechanism can ensure the normal operation of a single NE and has the features of CN. The mechanism uses relevant NE devices and reasonably allocates limited resources to improve e2e quality of element (QoE) and QoS. At the final, in this way, the mechanism optimizes the performance of the entire network. The following diagram shows the awareness-based service source end and distributed awareness-based link control layers system as Fig.5.

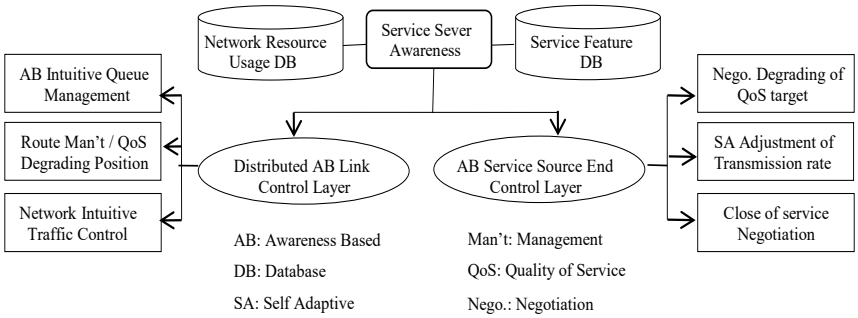


Fig.5 The collaboration mechanism in self-adaptive control

In this mechanism system, there are main three parts. The first part is service awareness sever with two databases of resource usage and service feature. The second part is the distributed awareness-based link control layer attached three function units covering queue management, degrading position and traffic control. The third part is the service resource end control layer with three function unites including degrading of QoS, adjustment of rate and close of service.

As far as source end, awareness-based service control is implemented through self-adjustment of the source-end transmission rate, intuitive closing of service and intuitive decrease of QoS target. Comparatively when the service source end launches a service in the traditional network, the current network condition is not considered. The CN service end has certain advanced cognitive functions. Thus, the cognitive information comes from the domain server or central control server.

In the Fig.5, the system can smartly work in different service requirement. 1) If certain conditions are met, for example, when bandwidth is sufficient, resources in the network can accept the access of other service traffic, and the service traffic can be transmitted to the peer end. 2) If high-priority users need to transmit services, but the current network does not provide sufficient resources as required by the user, the central cognitive server (or domain cognitive server) negotiates with users at the service source end. The result is alternative one. a) If the user accepts a reduction of QoS, the source end transmits the service traffic according to the negotiated results. b) If QoS requirements cannot be reduced, the cognitive server recycles the network resources being used according to the resource Distribution Policy or even forcibly closes certain low-priority services.

In regard of distributed awareness link, the link control is implemented through intuitive control of NE traffic, route management, QoS degradation positioning, and intuitive queue management. By perceiving the network and making decisions based on this information, these switches or routers with cognitive functions can intuitively control traffic of different services in the network. They also ensure the volume of trusted service traffic and key service traffic. Furthermore, they can limit the volume of unsafe traffic or non-key traffic as well.

Facing the dynamic network, the study of this paper focuses on the awareness and action function. 1) Usually service demands and network resources are changing in real time. By cognitive route management and QoS degradation positioning, the mechanism can automatically detect the bottlenecks or QoS-degrading parts of the e2e network. 2) The system can conduct analysis, decision making and re-route service traffic as well. 3) By adopting the intuitive queue management algorithm, the mechanism can fully determine the congestion in the CN. 4) Through negotiation method, the mechanism can effectively allocate resources for service target.

In here, awareness-based intuitive queue management is oriented to the server's collaborative driving policy. This policy is integrated into the intuitive queue management method to improve the resource appointment algorithm and router buffer management mode. As a result, the resources of router or end system can be completely reserved.

From above study and design, the system builds a new collaboration way for self-adaptive control mechanism. That further consolidates and realizes the entire intelligent function based on the service awareness in cognitive network.

5 Conclusions

In the existed network environment, there is comparatively way to go for realizing cognitive target. Owing to the complexity, isomerism, and ubiquity of the access mode and network applications, the current networks cannot meet the requirements of users' service quality. In academic field, many researchers pay efforts on studying this topic. Although up to now there is

not complete and perfect solution that could be applied in the existed network environment yet, there are still some bright researching achievements in different section of this field. The cognitive network has been considered as an innovative way for improving entire network performance and e2e system performance as well as simplifying network management. That is the prospective trend for next generation communication technology and network technology. The CNs is grandly important for ensuring performance in complex and isomerism networks environment

This work proposes a self-adaptive control feature for CNs. By service awareness, self-adaptive control can be implemented in a CN. This system is a new approach for solving the problem of NGN e2e service quality. Some techniques and methods have been applied in the experimental system in our joint key project" Developing smart techniques in network element model of NGN based on the service awareness ". Perception and adjustment functions are achieved and the collaborative control is applied for optimizing the service quality of network element. The technique demonstrates certainly stability and good performance.

This paper firstly discusses the diversity, burst nature and concurrency properties of current services in up-to-date communication. Meanwhile the paper analyses the difference of existed network and cognitive network by comparative way. Then it focuses on researching the key techniques and methods in cognitive network. Later the paper conducts relevant developing and puts forwards the layered architecture of cognitive network. At this basis, the paper provides the intelligent service awareness and classification model. Finally, the self-adaptive collaboration control mechanism is contributed as well. All of these achievements integrate different scientific disciplines and knowledge. It is believed that must be useful for the reference and application in cognitive network fields. Not only is it of important academic value but also useful practical significant.

Multi-Objective Service Composition by Integrating an Ant Colony System and Reinforcement Learning

Shunshun Peng

School of Computer Science and Engineering and Key Laboratory of Computer Network and Information Integration, Southeast University, China
E-mail: pengshunshun@seu.edu.cn

Hongbing Wang

School of Computer Science and Engineering and Key Laboratory of Computer Network and Information Integration, Southeast University, China
E-mail: hbw@seu.edu.cn

Qi Yu

College of Computing and Information Sciences, Rochester Institute of Tech, USA
E-mail:qi.yu@rit.edu

Abstract: Service composition uses existing services to deliver a new value-added service to achieve a business goal. A composition usually involves optimization of multiple and possibly conflicting objectives. Existing approaches cannot guarantee to effectively obtain solutions with a good trade-off among different objectives. In this work, we propose a hybrid approach based on an ant colony system and reinforcement learning for multi-objective service composition in a dynamic environment. It uses a set of ants to work in parallel by action choice and pheromone updating. The experimental results show that our approach is more efficiency than existing approaches.

Keywords: Multi-objective Service Composition; Dynamic Environment; Ant Colony System; Reinforcement Learning.

1 Introduction

Service composition provides a flexible means for leveraging existing services to deliver new value-added services, which help address the increasing complexity of modern software systems. In practice, many service composition problems involve dealing with multiple Quality of Service (QoS) parameters, which lead to multiple objectives. Achieving feasible

composite services hinges on satisfying multiple objectives simultaneously. Therefore, the optimization of service composition results in a multi-objective optimization problem.

Many approaches use a weighting mechanism to convert a multi-objective problem into a single-objective one to cope with multi-objective optimization, e.g., Peng et al. (2017); Wang et al. (2016). However, these methods suffer from two problems. First, it is difficult to assign weights to different QoS parameters. Second, more than one optimal solutions may be found when the relationship among multiple objectives is conflicting Van et al. (2014).

In this paper, we address these issues by proposing an approach based on ant colony system (ACS) and reinforcement learning (RL). ACS is a powerful combinational optimization method by pheromone-mediated indirect communication, which is inspired by the foraging behavior of ant colony. However, the pheromone trail guiding the action choice may be invalid in the dynamic scenarios, since the QoS parameters may change. RL can be leveraged to make optimal decisions for action choice in a dynamic environment by trial-and-error interaction, and learn the corresponding feedback on pheromone trail. In this work, we propose a new approach, named ACS-RL, which is based on ACS and RL. Our approach utilizes action choice and a pheromone updating strategy as well as the dominance relation to obtain a set of optimal solutions that balance multiple objectives. Besides, both the local and global QoS constraints are considered to reduce the search space by filtering unsatisfied component services and composite services. The experiments are presented to show the effectiveness and efficiency of the proposed approach. Our contributions are summarized as follows.

1. We propose ACS-RL, a new approach, which integrates ACS and RL, to effectively find a set of trade-off solutions.

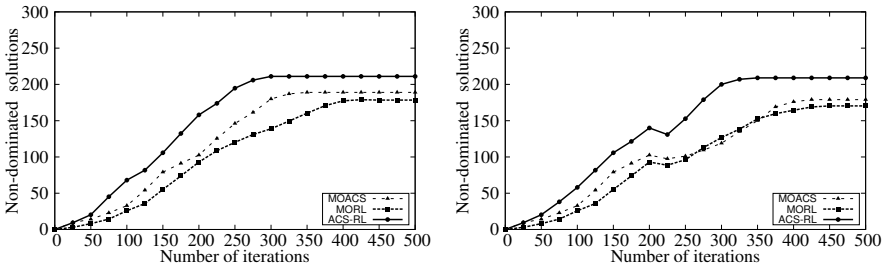
2. ACS-RL adopts learned multiple types of pheromone to make the optimal choice of action, and an enhanced pheromone updating strategy to speed up the optimization process and find optimal solutions.

2 The ACS-RL for Composition

In this section, we detail the proposed approach that integrates an ant colony system with reinforcement learning to effectively find optimal solutions of multi-objective service composition in a dynamic environment.

2.1 Action Choice

When using ACS for multi-objective service composition, we represent the solution construction as path building. In the process of building a path, the selection of each edge is described as action choice. The key challenge is that ACS makes action choice based on pheromone-mediated indirect communication. However, the pheromone density is related to the QoS parameter values of services, which change as services continue to evolve. This will make action choice problematic in a dynamic environment. To address this, we incorporate reinforcement learning with ACS to make optimal policies for action choice by trial-and-error interaction.



(a) Number of Candidate Services (b) Number of Abstract Services

Figure 1 Evaluation on covergence.

2.2 Pheromone Updating

After action choice, ants need to change the amount of pheromone on the execution path for guiding the action choice of other ants. The major problem of pheromone updating is that the amount of updates depends on the initial pheromone level, which is set as a certain amount. So the pheromone cannot well feedback the QoS parameter values of the service. In particular, in a dynamic environment, the pheromone updating strategy of adding new and better services is different from removing original services.

2.3 Non-Dominated Solutions Determining

After all ants construct paths, there are many solutions satisfying QoS constraints. Therefore, we need to confirm whether these paths are optimal solutions. We call the solutions satisfying QoS constraints as feasible solutions. The dominance relations can be used to select non-dominated solutions and hence the dominated solutions will be pruned.

3 Experiments

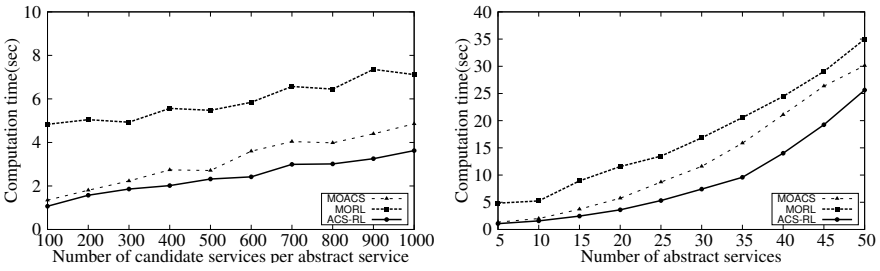
In this section, we carry out experiments to evaluate the efficiency of our approach.

3.1 Setup

The experiments are conducted on a HP PC with 3.40G Intel core i7-3770 CPU and 8GB RAM. We adopt the extended QWS dataset[1] Wang et al. (2014), which involves three QoS parameters, including ResponseTime, Throughput, and Availability. To evaluate the ACS-RL, we implement two approaches for multi-objective service composition for comparison purpose: multi-objective ant colony system (MOACS) and multi-objective reinforcement learning (MORL). The results are obtained by averaging over 50 runs of all these algorithms and the maximum number of iterations is set to 500 for each run.

3.2 Efficiency Evaluation

We evaluate the efficiency of the proposed approach by measuring the convergence efficiency in a dynamic environment and its execution time.



(a) Number of Candidate Services (b) Number of Abstract Services

Figure 2 Evaluation on the execution time.

3.2.1 Efficiency vs Convergence

In this experiment, we test the convergence to evaluate the impact of a dynamic environment on the efficiency of the case studies. Here, we set 5 abstract services and 500 web services. To reflect convergence efficiency in a dynamic environment, we set two scenarios by doing nothing and randomly changing %5 of QoS at the 225th iteration. Figure 1 shows the results in different scenarios. We can see that changes delay the convergence time. In Figure 1(a), ACS-RL converges at about 300th iteration, and MOACS converges at about 325th iteration with MORL at about 400th iteration. In Figure 1(b), convergence time postpones to 325th, 400th and 450th iterations for these three approaches. We can conclude that ACS-RL can adapt faster to the dynamic environment compared with MOACS and MORL.

3.2.2 Efficiency vs Execution Time

We also measure the execution time through two tests. In the first test set, the number of abstract services is fixed at 5 and the number of candidate services varies from 100 to 1000. The result is shown in Figure.2(a). In the second test set, the number of candidate services is fixed at 500 and the number of abstract services varies from 5 to 50. The result is shown in Figure.2(b). The result justifies the good efficiency of ACS-RL. This is because the pheromone updating strategy of ACS-RL can exploit the learned pheromone information to speed up the search process and the QoS constraints reduce the search space by pruning the infeasible services. In contrast, MORL and MOACS need more execution time because MORL is implemented by a single agent and the pheromone updating strategy of MOACS cannot well utilize the old pheromone information to accelerate search. In sum, the experiments prove the efficiency of ACS-RL compared with MOACS and MORL from the execution time perspective.

4 Related Work

Multi-objective service composition has recently received considerable attention in service computing. Gabrel *et al.* Gabrel et al. (2014) proposes a $0-1$ integer programming model to identify a composite service. In Klein et al. (2014), an extensible genetic algorithm is developed to obtain a near-optimal solution according to the fitness function, which synthesizes multidimensional QoS values. Wang *et al.* Wang et al. (2014) propose a hybrid approach, which integrates reinforcement learning with multi-agent techniques to find the optimal solution. However, these methods cannot accurately capture the user preference

over different QoS dimensions. Meanwhile, when the objectives are conflicting, a set of trade-off solutions should be found Van et al. (2014).

Different from existing efforts, we develop ACS-RL, which integrates ACS and RL to implement the multi-objective service composition in dynamic environments. It utilizes RL to make optimal policy of choosing action and leverages the improved pheromone updating strategy to find solutions. The dominance relation is applied to obtain a set of trade-off solutions.

5 Conclusion

In this paper, we addressed the problem of multi-objective service composition by proposing a new method, referred to as ACS-RL. It allows a set of ants to cooperate in parallel for a set of composite services, which have to satisfy both the global and local QoS constraints. The experimental results have shown a significant improvement over existing approaches.

References

Peng, Shunshun and Wang, Hongbing and Yuu, Qi. (2017) 'Estimation of Distribution with Restricted Boltzmann Machine for Adaptive Service Composition', *IEEE International Conference on Web Services (ICWS)*, pp.114–121.

Wang, Hongbing and Huang, Guicheng and Yu, Qi. (2016) 'Automatic Hierarchical Reinforcement Learning for Efficient Large-Scale Service Composition', *IEEE International Conference on Web Services (ICWS)*, pp.57–64.

Wang, Lijuan and Shen, Jun and Luo, Junzhou. (2014) 'Impacts of pheromone modification strategies in ant colony for data-intensive service provision', *2014 IEEE International Conference on Web Services (ICWS)*, pp.177–184.

Ardagna, Danilo and Pernici, Barbara. (2007) 'Adaptive service composition in flexible processes', *IEEE Transactions on Software Engineering*, Vol. 33, No. 66, pp.369–384.

Gabrel, Virginie and Manouvrier, Maude and Murat, Cécile. (2014) 'Optimal and automatic transactional web service composition with dependency graph and 0-1 linear programming', *International Conference on Service-Oriented Computing(ICSOC)*, pp.108–122.

Klein, Andreas and Ishikawa, Fuyuki and Honiden, Shinichi. (2014) 'SanGA: A self-adaptive network-aware approach to service composition', *IEEE Transactions on Services Computing*, Vol. 7, No. 3, pp.452–464.

Wang, Hongbing and Chen, Xin and Wu, Qin and Yu, Qi and Zheng, Zibin and Bouguettaya, Athman. (2014) 'Integrating on-policy reinforcement learning with multi-agent techniques for adaptive service composition', *International Conference on Service-Oriented Computing(ICSOC)*, pp.154–168.

Van Moffaert, Kristof and Nowé, Ann. (2014) 'Multi-objective reinforcement learning using sets of pareto dominating policies', *The Journal of Machine Learning Research*, Vol. 15, No. 1, pp.3483–3512.

A Service Annotation Quality Improvement Approach based on Efficient Human Intervention

Xuehao Sun[1,2], Shizhan Chen[1,2], Zhiyong Feng[1,3], Weimin Ge[1,2], Keman Huang[4,*]
[1]*Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin 300350, China*
[2]*School of Computer Science and Technology, Tianjin University, Tianjin 300350, China*
[3]*School of Computer Software, Tianjin University, Tianjin 300350, China*
[4]*Sloan School of Management, MIT, MA 02142, USA*
{*xuehaosun, shizhan, zyfeng, gewm*}*@tju.edu.cn, keman@mit.edu*

***Abstract*—Semantic Annotation plays an essential role in automatic service discovery and composition. However, existing approaches and tools cannot achieve high annotation quality to ensure the semantic service application. Meanwhile, the semi-automatic strategies for improving the annotation quality are time-consuming. To further improve the efficiency as well as the quality of the annotation, this paper presents an effective method involving human-computer interaction to further optimize the annotation procedure. Besides employing the feedback and propagation strategy to semi-automatically improve the annotation quality, the strategy to involve the manual annotation is developed when the efficiency of semi-automatically strategy is related low. To optimize the manual annotation procedure, a clustering based approach is presented to select the most impacted candidates to optimize the annotation improvement. In addition, to help the annotators to choose the correct annotation, the local ontology restriction based method is further designed to improve the recommendation performance. The experiments show that our approach effectively involving the human intervention can significantly improve the annotation quality, faster the quality improvement procedure and reduce the manual load by increasing the recommendation accuracy.**

***Keywords*-Annotation Quality for Web Services; Human-Computer Interaction; Efficiency of Quality Improvement; Effective Annotation Object; Local Ontology Restriction;**

I. Introduction

With the rapid advancement of Service Computing techniques, more and more reusable software services have been published to the Internet on a daily basis. Semantic annotation of web service is particularly meaningful in the automatic service application [1]. There are many approaches and tools that leverage semantic web technology [2]–[4] in last decades. The annotation language is usually LOD (Linked Open Data) [5], or the domain ontology bootstrapping from the service description [6]. However, the annotation procedure and the quality are becoming the essential concerns when implementing the semantic web service based applications.

Many of the approaches are either not validated or the validations are lacks of credibility [7], which means the Quality of Semantic Annotation (QoSA) [8] is not ensuring. A verification framework based on software testing technology has been proposed in [9]. However, a high degree of similarity does not mean the correctness of semantic annotation. Our previous work [8] has developed a technique to incrementally assess and correct the inaccurate annotations based on invocation. The invocation results and optimization logs [10] are learned to enhance the semantic annotations for other services.

However, the convergence process of automatic annotation improvement is related slow. In addition, in the context of micro-services in the business environment, each service invocation will require cost, including time or even monetary, from the annotators. This indicates the importance to accelerate the annotation improvement procedure. Furthermore, part of the annotations cannot be corrected by the automated process. Therefore, it is necessary to introduce the manual annotation into the quality improvement procedure. While the following challenges are needed to be solved:

1) *When to involve the human intervention can efficiently optimize the convergence process, including speed up the convergence and improve the final quality of annotation?* We calculate the optimizing efficiencies for operation and parameter propagation. An integrated efficiency threshold is set to terminate the automatic process and transfer to the manual annotation procedure.

2) *How to improve the performance of human intervention?* In another word, how to efficiently select the operations so that their manual annotations can significantly optimize the annotation quality improvement process? The idea here is that if we can identify the operation which has a large number of similar ones, then correcting its annotations can help to correct others through the automatic procedure. Hence, we develop the spectral clustering algorithm to identify the most efficient operations when involving the manual annotations.

3) *How to reduce the manual annotation load on the basis of selected operation?* Straightforwardly, the candidate annotations are recommended to assist annotators. Furthermore, we design the restrictive conditions to classify the local instances to further improve the recommendation performance, in turn to reduce the human burden during

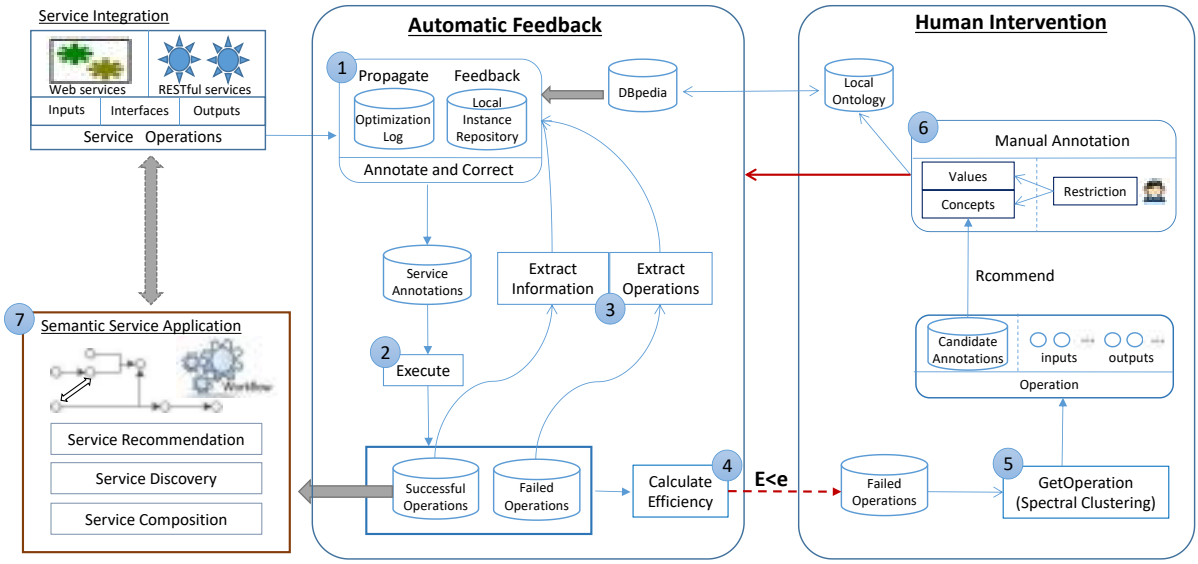


Figure 1. Semi-Automatic Iteration Annotation Framework for Web Services.

annotations as well as improve the success rate of automatic annotation.

The introduction of the efficiency threshold to transfer to the manual annotation benefits us significant time reduction, only 1/3 of the automatic iteration is needed to reach convergence. More importantly, the human intervention during the annotation improvement procedure gains significant QoSA improvement, reaching 89.20% which is 39.86% improvement comparing with the original strategy without involving the manual annotation. In addition, the spectral clustering strategy to identify the most effective operations for manual annotation decreases the needed manual involvement from the annotators: only 4.27% human intervention is required. Finally, the designed local ontology restriction further increases the recommendation performance, with 22.75% improvement in F-Measure.

Therefore, the main contribution of this paper is the effective human intervention strategy to improve the annotation quality improvement procedure, consisting of:

- The design of efficiency threshold to transfer to manual annotation;
- The spectral clustering approach to identify the effective annotated operations;
- The local ontology restriction based recommendation approach to reduce the manual annotation burden.

The rest of the paper is organized as follows. Section II introduces the semi-automatic feedback annotation framework. Section III details the human intervention mechanism. Section IV reports the implementation and the experiment results. Section V reviews the related work. Section VI concludes the paper.

II. Framework

Many approaches and tools for service annotation have been developed over these years [11]–[15], while few are verified to be efficient and accurate. This section gives an overview of semi-automatic feedback annotation framework for web services illustrated in Figure 1. This structure includes service integration, the automatic feedback process, human intervention mechanism and semantic service application.

A. Service Integration

In this paper, the SOAP and Restful services are both considered during the semantic annotation procedure. The differences between these two types of services are the description languages and invocation ways. We encapsulate the interface, input parameters and output parameters as an operation unit. The parameters are associated with ontology concepts and instances to realize the function of service operations. The interfaces are used for invocations to verify the semantic annotations.

B. Automatic Feedback Annotation

Each operation is executed to evaluate the QoSA. The local instance repository [8] is generated from the successful invocations, and feedback to the parameters which are equivalent. The optimization logs [10] are established to propagate to correlated operations. Therefore, the successful operations are continuously increased.

C. Human Intervention Mechanism

The human intervention is introduced to optimize the convergence process of automatic iterations, and to improve

the final quality. For each round, we calculate the efficiency of feedback and propagation strategy. The strategy for involving the manual annotation is developed when the efficiency is lower than a given threshold. The spectral clustering algorithm is adapted to identify the operation with the most significant contribution in next feedback process. On the basis of recommended semantics, we add the restrictive conditions to provide more accurate annotations. Meanwhile, the local ontology restriction is established by self-learning to refine concepts and classify instances. The successful invocation will trigger the automatic feedback process again.

D. Service Application

The process of our feedback and propagation annotation work is also the realization of service interaction. The semantic service system is established for the service recommendation, discovery and composition.

In our framework, the automatic feedback annotation is proposed in our previous work [8], [10]. In this paper, we will focus on the human-computer interaction.

III. Human Intervention

A. When should we involve the human intervention procedure?

After multiple rounds of automatic annotation modification, the effect of improvement is decreasing. It is because the effective information available for modification reduces in each circulation. Therefore, we should involve the human intervention procedure when necessary to improve the performance of each automatic circulation.

For each iteration, we extract the related operations to annotate and correct. The numbers of failed verifications and successful verifications are recorded, and we can calculate the efficiency of the operation verifications as follows:

$$E_o = \frac{|\{O_{successful}\}|}{|\{O_{successful}\} \cup \{O_{failed}\}|} \quad (1)$$

where $O_{successful}$ represents the successfully verified operations, and O_{failed} represents the failed ones. It can be seen that E_o is the fraction of the successful verifications in all extracted operations.

Note that the verifications of semantic annotations are in the operation level, whereas the modifications are in the parameter level. Therefore, the validity of modification evaluation cannot be judged only by invocation results. As shown in Figure 2, the semantic annotations (SA) in successful operation O_1 are propagated to the input parameters inp_{21} and inp_{22} of failed operation O_2 in round 1. The PCA is the pool of candidate annotations for each input parameter. The PCA for input inp_{21} is updated successfully in round 1. And the PCA for input inp_{22} is updated successfully by operation O_3 in round 2. Therefore, the successful invocation of O_2, in round 2, is on the basis of the successful modification of input inp_{21}. It means that the modification is effective even if the invocation is failed in round 1. Thus we consider the efficiency of feedback not only in the operation level but also in the parameter level.

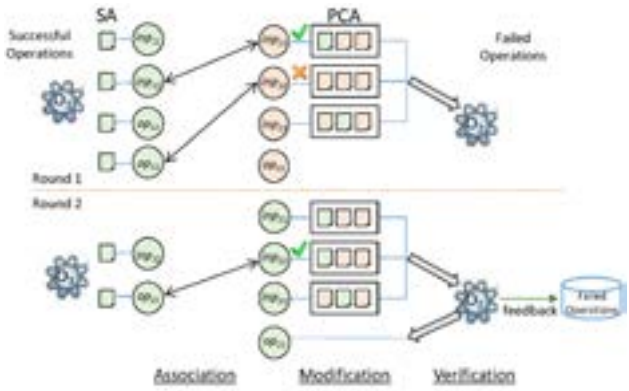


Figure 2. the Process of Modification and Verification Phase.

For each iteration, the parameters associated with the successful operations are extracted, such as input parameter inp_{21} and inp_{22} in round 1. However, the PCA for inp_{22}, which is more similar than the semantic annotation of the parameter op_{12}, cannot be updated in round 1. Therefore, the efficiency of parameter modification is defined as follows:

$$E_p = \frac{|\{p_{updated}\}|}{|\{p_{associated}\}|} \quad (2)$$

where, $p_{associated}$ represents the parameters which are associated with the successful operations. And $p_{updated}$ represents the parameters whose PCA are updated successfully. It can be seen that E_p is the fraction of the successfully updated parameters in all associated parameters. For example, in Figure 2, the efficiency of parameter modification is $\frac{1}{2}$ in round 1.

In this paper, we defined $E = E_o \times E_p$ to evaluate the efficiency of the feedback and propagation in quality improvement. If this efficiency is lower than a given threshold, then we need to involve the manual annotation to avoid the invalid and continue failure invocations. We will discuss the selection of this threshold during the experiment in Section IV.

B. Which service operation should be selected to annotate by the human?

Human intervention needs to efficiently select the operations so that their manual annotations can significantly optimize the annotation quality improvement process. The operation is an appropriate option if the corrected annotation of it can adaptively propagate to a large number of similar ones in the automatic modification process. In fact, if the two operations are very similar, the adaptive modification will be propagated from one to another. Therefore, the similarity based clustering algorithm is developed to identify the biggest cluster of similar operations.

1) Operation Similarity: The operation names are always composed of some words, such as 'GetWeatherByTown'. Therefore, it is not suitable to calculate the similarity by operation names. It is obvious that the parameters for each operation can realize the function. So we defined the operation similarity between operation O_1 and operation O_2 as follows:

$$os_{12} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} sim(p_{1i}, p_{2j})}{m \times n} \quad (3)$$

where p_{1i} refers to the parameters in operation O_1, p_{2j} refers to the parameters in operation O_2. $sim(p_{1i}, p_{2j})$ is the similarity between parameter p_{1i} and parameter p_{2j}. m and n separately refer to the number of parameters in operation O_1 and operation O_2. It can be seen that the operation similarity is the average similarity of every pair of parameters. Thus, os_{12} lies in the interval [0,1]. We specify $os_{ii} = 0$ to eliminate its own impact.

2) Spectral Clustering: The spectral clustering algorithm is developed to identify the operation which has the most similar ones. We first reduce the dimension of operation similarity matrix by the technique of Laplacian matrix to provide the indicators. Then we use K-Means based algorithm to cluster the failed operations in k subsets. The details of Spectral Clustering (SC) are described as follows:

1) Form the adjacency matrix $W = \{w_{ij}\} \in \Re^{n\times n}$ where $w_{ij} = os_{ij}$.
2) Calculate the diagonal matrix $D = \{d_{ij}\}$, where $d_{ii} = \sum_{j=1}^{n} w_{ij}$, and $d_{ij} = 0$ if $i \neq j$.
3) Construct the Laplacian matrix L by $L = D - W$.
4) Calculate the k largest eigenvectors of L, and form the matrix $X = [x_1, x_2, \ldots, x_k] \in \Re^{n\times k}$ by stacking the eigenvectors in columns.
5) Treat each row of X as a list of indicators of every operation, cluster them into k clusters by K-Means(X, k).

The k subsets are obtained by the SC above. Then we get the operation correlated with the most operations in the biggest subset.

Algorithm 1 Get Operation(GetO)

Input: $\{O_{failed}\}$: the failed operations;
Output: O: the operation selected to manually annotate;
1: $OS \leftarrow \{os_{ij}|i, j \in \{O_{failed}\}\}$;
2: $\{S_t|t \leq k\} \leftarrow SC(OS, k)$;
3: $\{l_t\} = \{length(S_t)\}$;
4: $t \leftarrow \{t|max\{l_t\}\}$;
5: $S = S_t$;
6: $O \leftarrow \{O_i|max\{\sum_j os_{ij}\}, j \in S\}$;
7: **return** O;

Algorithm 1 details the process of getting operation. Line 1 gets the operation similarity matrix for the failed operations. Line 2 gets the k subsets of operations by SC. Lines 3∼5 get the biggest subset S. Finally, return the operation O with the maximum similarity summation in S (Line 6∼7).

C. Recommendation and Annotation

As shown in Figure 2, the invocation of operation O_2 is failed in round 1, whereas the candidate semantic annotations for the parameters inp_{21} and inp_{23} all include the correct annotations. Straightforwardly, the PCA is recommended to assist annotators as shown in Figure 3. In fact, we only need to annotate the input parameter inp_{22} and select the correct annotations for the other parameters.

The semantic annotations are recommended, including concepts and instances. There are many instances responding to one concept. For example, the concept 'Town' has a large number of instances. However, the parameters in different operations are limited to different conditions. For example, the operation named 'GetUkLocation' shows that its restrictive condition is the UK. Therefore, we can only execute the operation by selecting the instances of town in the UK, which increases the workload and reduces the precision of recommendation. In order to refine the semantic annotations, we design the annotation concept with a restrictive condition:

$$ac =< rIns, bc > \quad (4)$$

where $rIns$ refers to the restrictive condition, such as 'UK'. bc refers to the basic concept, such as 'Town'. If there is no restrictive condition on a basic concept, the value of $rIns$ is all.

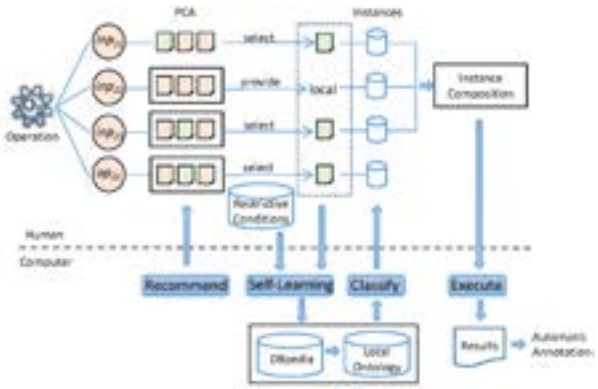


Figure 3. the Process of Recommendation and Annotation.

As shown in Figure 3, the restrictive conditions, provided for semantic annotations of prameters, can be learned from DBpedia to generate more similar concepts, such as 'USA Town', 'China Town', etc. The instances, classified with restrictive concepts, are saved in Local Ontology. Thus, the details of local ontology learning from DBpedia are reported in Algorithm 2.

Line 1 gets the concept of $rIns$ in DBpedia. Lines 2∼3 get all instances of the concept rc and generate restrictive concepts combined with bc. Lines 4∼7 classify the instances

of bc by the restrictive conditions and put them into the local ontology base. For example, we first get all countries by the restrictive condition 'UK'. Then the similar annotation concepts, such as 'USA Town', are generated. Next we adopt the operation 'GetTownByCountry' to classify all towns by countries. Finally, the towns are stored by different countries.

Algorithm 2 Local Ontology Learning(LOLearning)

Input: $ac =< rIns, bc >$: the concept annotated by human; DBpedia : DBpedia base;
Output: LO: Local Ontology generated by self-learning;

1: $rc \leftarrow getConceptByInstance(rIns, DBpedia)$;
2: $\{rIns_i\} \leftarrow getInstanceByConcept(rc, DBpedia)$
3: $AC \leftarrow \{ac_i | ac_i =< rIns_i, bc >\}$
4: $O \leftarrow getOperation(bc, rc)$;
5: **for** each $ac_i \in AC$ **do**
6: $\quad Ins = \{ins_j\} \leftarrow Execute_O(rIns_i)$;
7: $\quad LO \leftarrow LO \cup \{< Ins, instanceof, ac_i >\}$
8: **end for**
9: **return** LO;

Therefore, the semantic annotations, including concepts and instances, are more accurate to reflect the semantics of parameters. Meanwhile, the connections of operations are more accurate. For example, the operation 'GetWeatherByTown' limited in the USA will be not connected with the operation 'GetUKLocation' by parameter 'Town'. That is why some of the parameters are obviously equivalent but the instances cannot be successfully propagated. As a result, the recommendations are more accurate, and the success rate is higher in automatic annotations. They both contribute to reducing the manual load.

D. Human-Computer Interaction

For each round of automatic iterations, if the efficiency is related low, the human intervention will be involved. And the operation is identified by clustering algorithm. Finally, we annotate the operation on the basis of recommended semantics. Through the human-computer interaction, the process of annotation improvement iterations can be continuously and efficiently performed.

Algorithm 3 details the process of the semi-automatic feedback annotation. Line 1 sets a flag to perform the whole semi-automatic annotation iterations. Lines 2∼10 modifies the semantic annotations for the failed operations by the information of successful operations and local ontology when the efficiency is higher than e. Line 11 gets the operation in failed when the efficiency is related low. Line 12 annotates the operation by annotation concept and instance for each parameter on the basis of the recommended semantics. Lines 13∼14 update the successful operation and failed operations for next round. Line 15 generates the local ontology by self-learning. The efficiency is set as 1 so as to continue the automatic annotation iteration process. The whole semi-automatic feedback annotation process is terminated when the efficiency of manual annotation is also lower than e. It can be seen that the annotator's work is only to annotate the operation as line 12.

Algorithm 3 Semi-Automatic Feedback Annotate

Input: $O_{successful}$: all successful operations in last round; O_{failed}: all failed operations in last round;
Output: OL: Optimization Log; LIR: Local Instance Repository; LO: Local Ontology;

1: **while** $flag$ **do**
2: $\quad$ **while** $E > e$ **do**
3: $\quad\quad \{OL, LIR\} \leftarrow$ extract by $O_{successful}$ and LO;
4: $\quad\quad$ AutomaticAnnotate(O_{failed}) $\leftarrow \{OL, LIR\}$;
5: $\quad\quad$ Update($O_{successful}, O_{failed}$);
6: $\quad\quad$ Update($p_{updated}, p_{associated}$);
7: $\quad\quad E_o \leftarrow \frac{|\{O_{successful}\}|}{|\{O_{successful}\} \cup \{O_{failed}\}|}$
8: $\quad\quad E_p \leftarrow \frac{|\{p_{updated}\}|}{|\{p_{associated}\}|}$
9: $\quad\quad E = E_o \times E_p$;
10: $\quad$ **end while**
11: $\quad O \leftarrow GetO(O_{failed})$;
12: $\quad \{< p, ac, ins > | p \in O\} \leftarrow$ manual annotation;
13: $\quad \{O_{successful}\} \leftarrow O$;
14: $\quad \{O_{failed}\} \leftarrow \{O_{failed}\} - \{O\}$;
15: $\quad LO \leftarrow LOLearning(ac)$;
16: $\quad E \leftarrow 1$;
17: **end while**

IV. Experiments

A. Data

In this section, we perform the semi-automatic feedback annotation framework to prove the effectiveness. First, we employed the dataset consisting of 115 real-world web services proposed in [10]. As real-world web services growing fast, we have added new data from 894 web services which are tested to be available. The services include SOAP services and RESTful services in seven domains: currency, weather, traffic, geography, date, hotel and holidays. Therefore, the dataset in our system is summarized in table 1.

Table I
Benchmark Dataset

Data	Services	Operations	Parameters
Original	115	790	14326
New Added	894	4459	54947
Total	1009	5249	69273

Note that our framework is generic and it can be further extended in the following aspects: 1) the knowledge base can be switched by OpenCyc [16], or workflow provenance [15]

etc; 2) the dataset can be substituted by other web services which can be encapsulated as operation units.

B. Efficiency Threshold to Involve Human Intervention

To determine the efficiency threshold, we observe the change of the feedback efficiency in the automatic modification process. As shown in Figure 4, the efficiency is between 0.09 and 0.34 in first several iterations. However, it is a continuous decline when it is lower than 0.09. To further determine the value of e, we report the QoSA by different threshold as shown in Figure 5.

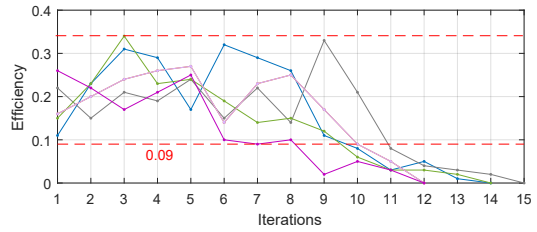


Figure 4. Efficiency of Feedback Propagation.

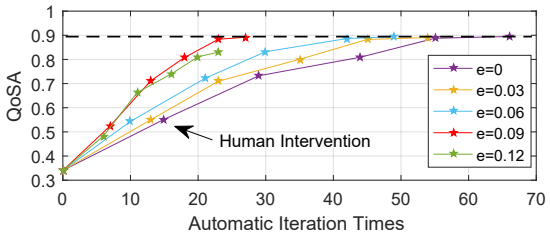


Figure 5. Effectiveness of Human Intervention for Different e

The horizontal axis represents the number of automatic iterations, and the points of the pentagram indicate the time point of human intervention. It can be seen that when the value of e is lower than 0.09, the final QoSA will all reach 89%. However, the higher the e is, the few iterations need to perform. But when the e is 0.12, the QoSA is lower because the automatic modification is terminated at the effective time. It is obvious that the convergence of quality improvement procedure is faster when the efficiency threshold is 0.09, only 1/3 of the automatic iteration is needed.

C. Operation Selection

In order to evaluate the effectiveness of Spectral Clustering method for selecting the operation, we considered the following methodologies:

- **Original Automatic Feedback** (OAF) : all feedback strategies are performed by automatic process without human intervention.
- **Random Selection** (RS) : the operation is selected by random when the human intervention is occurred.
- **Spectral Clustering Selection** (SCS) : the operation is selected by algorithm of spectral clustering when the human intervention is occurred.

Because of the lots of new data, we count the QoSA improvement after 20 times of human intervention as a round. Figure 6 reports the results of QoSA for each methodology. It can be seen that our method is 39.68% improvement for QoSA comparing with OAF. It is obvious that the human intervention plays an important role in the process of automatic feedback optimization. To remove the random effect of RS, we ran ten times to get the average result for each round. It is shown that the final QoSA of SCS reaches 89.20% in five rounds. To further compare the performance of the SCS and RS, we have compared the number of rounds and the final QoSA by t-test as shown in Figure 7,8. It can be seen that $p = 0.000 < 0.01$ and the mean difference is -0.153 for the test results of QoSA, which means that the final QoSA of the method SCS is significantly larger than the method RS. Likewise, the test results of rounds show that the convergence of the method SCS is significantly faster than the method RS.

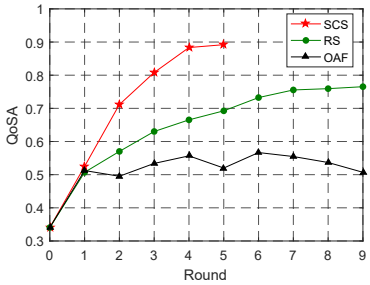


Figure 6. Effectiveness of Operation Selection Method.

	Test Value = 0.8920					
	t	df	Sig. (2-tailed)	Mean Difference	95% Confidence Interval of the Difference	
					Lower	Upper
QoSA	-11.625	9	.000	-.15325	-.1831	-.1234

Figure 7. T-Test Results of Convergence QoSA.

	Test Value = 5					
	t	df	Sig. (2-tailed)	Mean Difference	95% Confidence Interval of the Difference	
					Lower	Upper
Round	3.857	9	.004	4.50000	1.8608	7.1392

Figure 8. T-Test Results of Rounds.

D. Recommendation Evaluation

To evaluate the performance of the recommendation, we compared the number of recommended semantic annotations with two methodologies as follows:

- **Original Manual Annotation** (OMA): all parameters are annotated by original recommended semantic annotations without restrictive conditions.
- **Annotation with Restrictive Conditions** (ARC): all parameters are annotated by recommended semantic annotations and restrictive conditions.

To remove the effect from different parameters, we select the same operation for each round. As shown in Figure 9, we only compare the change of the difference between the two methods. The method ARC has a significant decline compared with OMA after three rounds. Finally, the number of recommended semantic annotations by ARC is only 1/6 of the original in the eighth round.

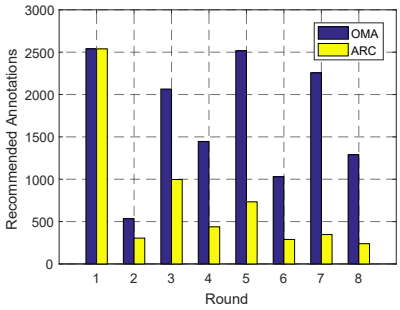


Figure 9. the Number of Recommendation Annotations comparison.

In order to evaluate the effectiveness of recommended semantic annotations on the basis of the significant decline, we use three measures, including precision, recall, and F-Measure. Formally:

$$P = \frac{|\{L_{sa}\} \cap \{R_{sa}\}|}{|\{L_{sa}\}|} \tag{5}$$

$$R = \frac{|\{L_{sa}\} \cap \{R_{sa}\}|}{|\{R_{sa}\}|} \tag{6}$$

$$F - Measure = \frac{2P * R}{P + R} \tag{7}$$

where L_{sa} is the list of recommended semantic annotations. R_{sa} is the list of right semantic annotations. The experiment results are shown in Figure 10 and 11.

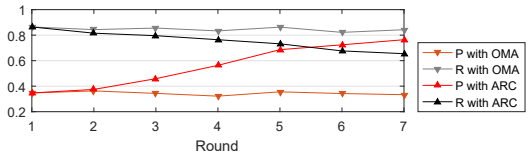


Figure 10. Precision and Recall in Recommendations.

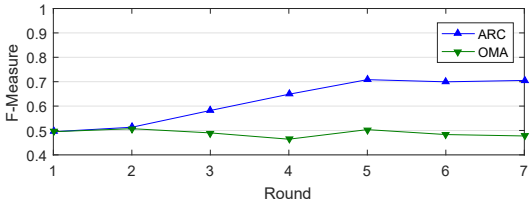


Figure 11. Comparison of F-Measure in Recommendations.

It is obvious that the precision with ARC is higher than OMA after several rounds. However, the value of recall is lower than OMA because the decline of recommended semantic. Results of integrated evaluation F-Measure reported in Figure 11 show that the performance of ARC is better than OMA, with 22.75% improvement.

V. Related Work

Semantically annotating web services is an important aspect to support the automatic matchmaking and composition of web services [7]. Early manual annotation of web services was extremely cumbersome, which required professional knowledge, the specific ontology, and the specific function. Then there are many types of research on the automatic semantic annotation. Most are classified into three categories: annotation by schema matching techniques [11], [12], annotation by machine learning algorithms [13], [14], and annotation by data-driven workflows [15]. However, the real invocation and interaction of the service are the standard to mark the correctness of the annotation quality.

Belhajjame [9] first proposes the verification framework based software testing technology. The ontology knowledge and manual effort are adapted to provide test cases for inputs. The invocation results are recorded to analysis the correctness of the annotations. Chen [8] extends to generate the local instance repository, from invocation results to feedback, to modify the annotation. Manual annotation is needed to give a successful invocation when there is no correct annotation.

It is obvious that if the instances are accurate to invoke the operation, the concepts corresponding to the instances are correct to reflect the semantics. Segev [6] introduces the domain ontology bootstrapping from the web service description. Ontology adaptive learning has become the focus of the research [17], [18].

These proposals provided the direction for improving the quality of semantic annotations for web services. However, none of them considered the efficiency of human-computer interaction. Additionally, none of them considered how to construct the ontology which is more suitable for web service semantics. From this perspective, this paper introduces the human intervention mechanism and local ontology to provide the high efficiency and quality for annotating the web services.

VI. Conclusion

In this paper, we present an effective method involving human-computer interaction to further optimize the annotation quality improvement procedure. The efficiency threshold is introduced to transfer to the manual annotation when the feedback and propagate strategy cannot effectively improve the annotation quality. We adopt the technique of spectral clustering to identify the operations which have a large number of similar ones, so that the manual annotation for the operations can significantly optimize the annotation quality improvement process. Furthermore, we design the

restrictive conditions to classify the local instances to further improve the performance of the recommendation. Meanwhile, the success rate of automatic modification is also improved to reduce the human load.

The experiments show that our method needs only 1/3 of the original automatic iterations to reach the convergence, which will significantly reduce the time and resource cost for the annotation quality improvement. Moreover, combing the human intervention and the clustering method for identifying the operation make a significant QoSA improvement, reaching 89.20% which is 39.80% improvement compared with the original strategy. The final QoSA of clustering method is significantly larger than random selection method, and the convergence is significantly faster. Only 4.27% human intervention is required in our experiments. Finally, the recommendation performance is increased with 22.75% improvement in F-Measure by using the local ontology restriction.

In the future, we will further our framework to construct the service ecosystem with the semantic web services and their interaction. Moreover, we will further extend the method of adaptively learning the relationships for semantics in the knowledge base to provide more accurate information.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China grants 61502333, 61572350 and 61672377. Keman Huang is the corresponding author.

REFERENCES

[1] J. Cardoso and A. P. Sheth, *Semantic Web Services, Processes and Applications*. Springer US, 2006.

[2] A. V. Paliwal, B. Shafiq, J. Vaidya, H. Xiong, and N. Adam, "Semantics-based automated service discovery," *IEEE Transactions on Services Computing*, vol. 5, no. 2, pp. 260–275, April 2012.

[3] D. Repchevsky and J. L. Gelpi, "Bioswr–semantic web services registry for bioinformatics," *Plos One*, vol. 9, no. 9, p. e107889, 2014.

[4] S. N. Han, G. M. Lee, and N. Crespi, "Semantic context-aware service composition for building automation system," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 752–761, Feb 2014.

[5] H. Q. Yu, X. Zhao, S. Reiff-Marganiec, and J. Domingue, "Linked context: A linked data approach to personalised service provisioning," in *2012 IEEE 19th International Conference on Web Services*, June 2012, pp. 376–383.

[6] A. Segev and Q. Z. Sheng, "Bootstrapping ontologies for web services," *IEEE Transactions on Services Computing*, vol. 5, no. 1, pp. 33–44, Jan 2012.

[7] D. Tosi and S. Morasca, "Supporting the semi-automatic semantic annotation of web services: A systematic literature review," *Information & Software Technology*, vol. 61, no. C, pp. 16–32, 2015.

[8] J. Chen, Z. Feng, S. Chen, K. Huang, W. Tan, and J. Zhang, "A novel lifecycle framework for semantic web service annotation assessment and optimization," in *2015 IEEE International Conference on Web Services*, June 2015, pp. 361–368.

[9] K. Belhajjame, S. M. Embury, and N. W. Paton, "Verification of semantic web service annotations using ontology-based partitioning," *IEEE Transactions on Services Computing*, vol. 7, no. 3, pp. 515–528, July 2014.

[10] K. Huang, J. Zhang, W. Tan, Z. Feng, and S. Chen, "Optimizing semantic annotations for web service invocation," *IEEE Transactions on Services Computing*, vol. PP, no. 99, pp. 1–1, 2017.

[11] F. A. Musyaffa, L. Halilaj, R. Siebes, F. Orlandi, and S. Auer, "Minimally invasive semantification of light weight service descriptions," in *IEEE International Conference on Web Services*, 2016, pp. 672–677.

[12] X. Hu, Z. Feng, and S. Chen, "Augmenting semantics of web services based on public open ontology," in *2013 IEEE International Conference on Services Computing*, June 2013, pp. 304–311.

[13] E. S. Chifu and I. A. Letia, "Unsupervised semantic annotation of web service datatypes," in *Proceedings of the 2010 IEEE 6th International Conference on Intelligent Computer Communication and Processing*, Aug 2010, pp. 43–50.

[14] K. Lerman, A. Plangprasopchok, and C. A. Knoblock, "Automatically labeling the inputs and outputs of web services," in *National Conference on Artificial Intelligence*, 2006, pp. 1363–1368.

[15] K. Belhajjame, S. M. Embury, N. W. Paton, R. Stevens, and C. A. Goble, "Automatic annotation of web services based on workflow definitions," *Acm Transactions on the Web*, vol. 2, no. 2, p. 11, 2008.

[16] C. Matuszek, J. Cabral, M. J. Witbrock, and J. Deoliveira, "An introduction to the syntax and content of cyc," pp. 44–49, 2006.

[17] P. Minervini, V. Tresp, C. D'amato, and N. Fanizzi, "Adaptive knowledge propagation in web ontologies," *ACM Trans. Web*, vol. 12, no. 1, pp. 2:1–2:28, Aug. 2017.

[18] S. Mokarizadeh, P. Küngas, and M. Matskin, "Ontology learning for cost-effective large-scale semantic annotation of web service interfaces," in *Knowledge Engineering and Management by the Masses*, P. Cimiano and H. S. Pinto, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 401–410.

[19] V. Saquicela, L. M. Vilches-Blzquez, and scar Corcho, "Semantic annotation of restful services using external resources," 2010.

[20] B. Liu, K. Huang, J. Li, and M. Zhou, "An incremental and distributed inference method for large-scale ontologies based on mapreduce paradigm," *IEEE Transactions on Cybernetics*, vol. 45, no. 1, pp. 53–64, Jan 2015.

The effectiveness research on O2O service recommendation strategy

Shuai Huangfu, Xiao Xue

School of Computer Science, Henan Polytechnic University, Jiaozuo, China

Abstract: In the information era, the recommendation platform needs to filter out the most effective information for people. O2O service recommendation demands offline resources support. In the case of large user base, traditional service recommendation methods will cause many users to adopt the same recommendation at the same time. Thus, it will be so crowded at the specific service point because of the limited service resources. It may be useless information and bad experience for users to get a delayed recommendation. So, the game among users may affect the performance of service recommendation. How to solve the effectiveness of O2O service recommendation is the important problem in the filed. Based on the interactive mechanism of cyber-social, this paper puts forward a set of O2O service recommendation strategies and conducts performance comparison analysis of these strategies by using computational experiment method. Finally, the optimized O2O service recommendation strategy is identified.

Keywords: Online To Offline; collaborative filtering; game theory; service recommendation strategy; computational experiment; Cyber-Social

1 Introduction

The resources in the traditional recommendation model are unlimited such as news recommendation. The characteristics of the demand side are always emphasized and the characteristics of the supply side are often ignored. But, in O2O service recommendation, the factors of both sides need to be considered, including the location of the user, weather, time (demand-side) [1-3] and the attributes of service providers (supply-side).

Taking the catering industry as an example, according to traditional service recommendation results, a large number of users will adopt the recommendation information at the same time and go to the same service point in a short period of time. The supply side is inefficient and the development is unbalanced. Customers wait too long to be satisfied. When the user drives to the appropriate service point, a similar problem occurs because of the game among customers, which causes congestion at an

intersection. Therefore, the interaction mechanism of supply-demand[4-5] has an important influence on the performance of O2O service recommendation strategy. How to integrate this interaction mechanism in O2O service recommended strategy has become the key problem to be solved in the field.

2 Design of O2O Service Recommendation Strategy

2.1 Strategy 1: Recommendation algorithm for demand side (User-CF)

According to user similarity based on user's base information to calculate the user's comprehensive situation similarity:

$$sim\left(u_x,u_y\right)=a\times sim(u_x\left(c_t\right),u_y\left(c_t\right))+b\times sim(u_x\left(b_n\right),u_y\left(b_n\right)) \quad (1)$$

Here, $a\geq 0;b\geq 0;a+b=1$. After obtaining the user's comprehensive situation information similarity, the potential preferences of the non-scoring items is calculated:

$$r_{xjt}^{'}=\overline{r_x}+\frac{\sum_{u_y\in\bar{u},(j,t,n)\in\bar{s}_{xy}} sim\left(u_x,u_y\right)\times\left(r_{\left(u_y,i_j,c_t,b_n\right)}-\overline{r_{u_y}}\right)}{\sum_{u_y\in\bar{u}} sim\left(u_x,u_y\right)} \quad (2)$$

Then, by predicting the user's interest preference value, the Top N items in the preference values list are recommended to the target users.

2.2 Strategy 2: Recommendation algorithm for supply side (Service-CF)

By means of adopting formula(2), the sequence of Top-N service items Q_a are recommended. According to $P_x=1-\frac{N}{M}$, the current service capability of each service item P_x is identified. The comprehensive list of service items $L_a=\alpha\times Q_a+\beta\times P_a$, $\alpha\geq 0;\beta\geq 0;\alpha+\beta=1$.

Here, P_x represents the current service capability value of the service item, N represents the number of customers at the serving point, and M represents the maximum number of customers at the serving point.

2.3 Strategy 3: Consider connection path between supply side and demand side (O2O-CF)

When the starting point R and the terminal point S is determined, the amount of virtual resources in the a-th section of the k-th candidate path is calculated:

$$x^{'}{}_a=\sum_r\sum_s\sum_k g_k{}^{rs}*\delta^{rs}{}_{a,k} \quad (3).$$

Here, g_k^{rs} represents the virtual resource ownership of this k-th path between R and S. $\delta^{rs}_{a,k}$ is a path correlation variable, if the k-th path contains the section a, then, $\delta^{rs}_{a,k}$ =1, otherwise, $\delta^{rs}_{a,k}$ =0.

Then, the travel time of the k-th candidate path is calculated:

$$v_k^{rs} = \sum_a t_a(x'_a)\delta^{rs}_{a,k} \quad (4)$$

Here, $t_a(x'_a)$ is the travel time of the section issued by service provider, which is a monotonically increasing function of the total amount of virtual resource x'_a. Finally, the recommendation list of candidate paths are recommended to users based on the value of v_k^{rs} in order (from big to small).

Table1. The specific steps of three strategies

User-CF	Service-CF	O2O-CF
Input: "Target user (U)-service (S)- situation (C)"Multidimensional scoring matrix; **Output:** Top-n services in the recommendation list;	**Input:** "Target user (U)-service (S)- situation (C)"Multidimensional scoring matrix; **Output:** Top-n services in the recommendation list;	**Input:** "Target user (U)-service (S)- situation (C)"Multidimensional scoring matrix; **Output:** Top-n services in the recommendation list;
Step 1: Calculate the comprehensive situation similarity $sim(u_x,u_y)$ based on formula (1). **Step 2:** Find the top N of $sim(u_x,u_y)$ with the highest score, as the nearest neighbor to the target user u_x based on formula (2). **Step 3:** Generate recommendations. The top N items with the highest preference value are predicted as the recommended items for the target user u_x.	**Step 1:** Calculate comprehensive situation similarity $sim(u_x,u_y)$ based on formula (1). **Step 2:** Find the top-N highest score of $sim(u_x,u_y)$ as the nearest neighbor of the target user u_x based on formula (2). **Step 3:** Get the list of recommended Top-N service items. **Step 4:** Rebuild the new list L_a and recommend.	**Step 1:** Calculate comprehensive situation similarity $sim(u_x,u_y)$ based on formula (1). **Step 2:** Find the top-N highest score of $sim(u_x,u_y)$ as the nearest neighbor of the target user u_x based on formula (2). **Step 3:** Get the list of recommended Top-N service items, rebuild the new list L_a and recommend. **Step 4:** After user selection, the shortest path between supply and demand is determined by the Dijkstra algorithm. **Step 5:** Determine whether the virtual resource occupancy exceeds the congestion threshold according to formula (3), and if not, step 3 is recommended. **Step 6:** If the threshold is exceeded, recommend

		according to formula (4).

3Result of computational experiment

3.1Initial setting of computational experiment

Table 2 the setting of experiment parameter

Name	Remark
Weather	Bounded random in the range {0(bad), 1(general), 2 (good) .
Demand of user	The customer demand obeys normal distribution $\gamma \sim N(\mu,\sigma^2)$, the parameter size of μ determines the amplitude of customer demand; The parameter size of σ^2 determines the volatility of market demand. There, $\mu = 65, \sigma$ is large.Bounded random in the range{50，80}.
The cost of service	Bounded random in the range {20, 60}.there are 3 types {Low cost: 20-30; Medium cost: 31-50; High cost: 51~60}.
Price of service	Bounded random in the range {50,130}. Corresponding price cost： {Low price: 50-70; Medium cost: 71-110; High cost: 111~130}.
The profits of service	$\Pr ofit(t+1) = \Pr ofit(t) + (Price - cost)*CurUser(t)$
Maximum capacity	Bounded random in the range{5,10}.
Current capability	$Capacity(t) = \begin{cases} MaxUser - NewUser(t); & MaxUser \geq NewUser(t) \\ 0; & MaxUser < NewUser(t) \end{cases}$

3.2 Evaluation experiment of service strategy

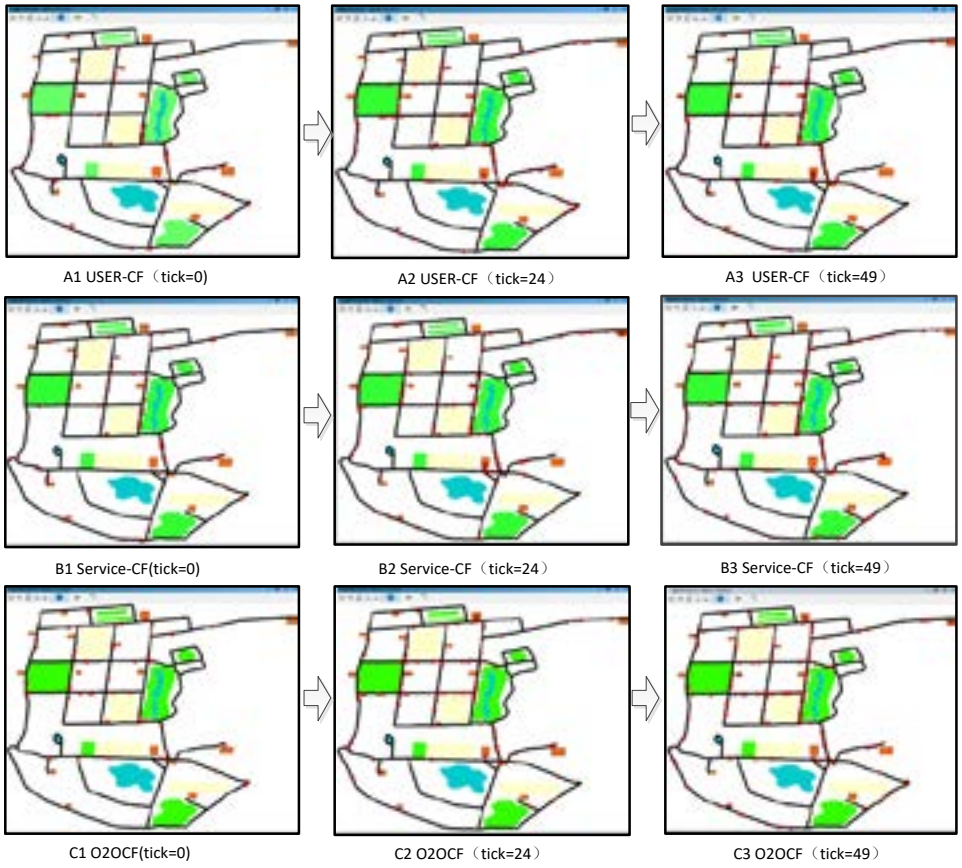
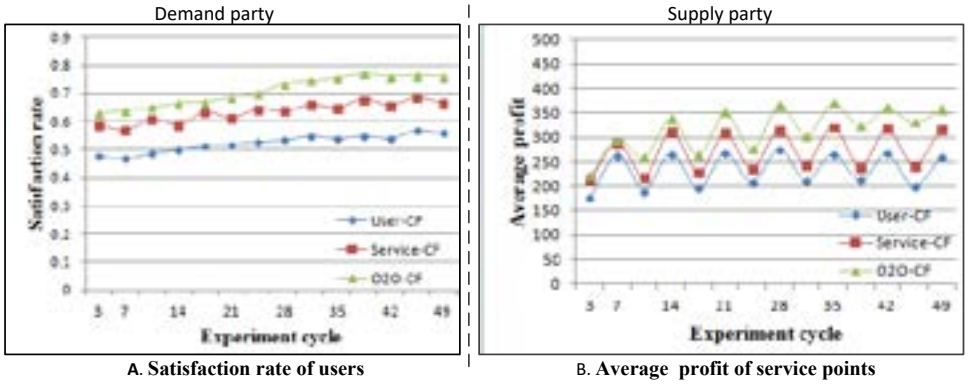


Fig.1The evolution results of three service recommendation strategies

The experimental interface is to simulate the real condition of a city through GIS map. The main roads and different service points (represented as the orange box in the picture) are distributed in various regions. According to the service strategy, these services can be recommended to customers (red pentagram).



A. Satisfaction rate of users　　B. Average profit of service points

Fig.2 The performance comparison of different service recommendation strategies

It can be concluded from Fig.2: because of the occurrence of service congestion, the performance of User-CF is the worst in three strategies: satisfaction rate of users is lower and the business profit is less. In turn, the performance of service-CF is relatively growing until the road congestion occurs between supply side and demand side. In addition, the performance of O2O-CF is growing more. The experimental results show that the O2O-CF service recommendation strategy is the best one, which can improve the effectiveness of service recommendation.

References:

[1] Bin Li, Bo Zhang, Xuejun Liu, Wei Zhang. Collaborative filtering recommendation algorithm based on Jaccard similarity and location behavior. Computer Science, 2016,(43)12,200-205.

[2] Kaiji Liao，Jiewen Ouyang，Yunjiang Xi. A collaborative filtering algorithm based on location information. Systems Engineering, 2015(12):121-125.

[3] Xiao Xue, Hongfang Han, Shufang Wang, and Chengzhi Qin. Computational Experiment-based Evaluation on Context-aware O2O Service Recommendation [J]. IEEE Transactions on Services Computing, 2016.

[4] Jing Zeng; Laurence T. Yang; Man Lin; Huansheng Ning; Jianhua Ma. A survey: Cyber-physical-social systems and their system-level design methodology. Future Generation Computer Systems, 2016.

[5] Pingyu Jiang, Kai Ding, Jiewu Leng. Towards a cyber-physical-social-connected and service-oriented manufacturing paradigm: Social Manufacturing. Manufacturing Letters, 2016, 7:15-21.

An approach for service selection based on the records of request/matching

Rong Yang, Dianhua Wang, Shuwen Deng

School of Computer Science and Technology, Hubei University of Science and Technology, Xianning 437100

Abstract: As the emergence of more and more Web services, it becomes more difficult that finding the candidate atomic services, and composing them to form a matched service process, where not only time is strictly limited, but user demand must be satisfied. In order to solve this problem, this paper presents a service process selection method, which is driven by historical successful service request /matching scheme. Firstly, the system framework model of this paper's method is analyzed, and several concepts about service selection and combination are formalized. Then, the method model and algorithm used in this paper are analyzed in detail. Finally, through a set of experiments, the effectiveness and robustness of our approach are evaluated.

Keywords: Web Service; Service Selection; Service Composition; Service Process

1 Introduction

Now, Web services are becoming more and more popular, and almost all software companies, governments, and even some individual applications are deployed using Web services. In particular, with the emergence of the Internet of things and cloud computing , it is possible that everything is as a service. For some complex applications, a single service cannot meet the functional requirements, so multiple atomic services need to be composed to meet the user's special functional requirements and non-functional requirements (QoS: quality of service).

Considering the similar functionalities of services, quality-aware services composition has received great attention nowadays[1-5]. However, it is a NP hard problem to select the best atomic services at each stage and combine them to form the optimal composite service process, which must meet user's constraints. [6] uses historical information to optimize composite services. In [7], in order to improve the efficiency of multi-agent service combination, it tried to explore those agent patterns with high frequency. For reusing, [8] mined the composite service flow pattern from the execution log. [9] also explored how to reuse service process fragments, but only considering the benefits of both parties , i.e. service providers and service users. For reducing the response time for new service requests, [10] recommended composite service by calculating the matching probability among historical user requests and solutions .

In this paper, we study how to make full use of prior knowledge to improve the efficiency of service composition, which is based on our previous work [11](i.e. the SCKY method) and is very close to [10].

2 System framework and formal description

2.1 System framework

Figure 1 is the overall framework model of the system. As shown in Fig. 1, it firstly cluster records of user service requests. Specifically, it put those similar service requests to a same class. So, for some new user request, the system can response quickly by means of matching the similarity among users. Over time, all the service composition schemes become the valuable resources. In this paper, we use SCKY to store and retrieve service composition processes. Furthermore, by means of SCKY, we can reuse any granularity of service process fragments[11], where hit probability of each service process fragment can be calculated by frequency statistics.

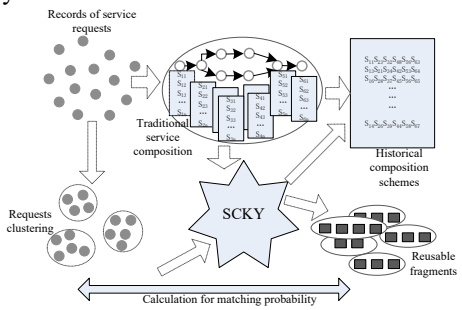


Figure 1. System framework model

2.2 Formal description

It is assumed that one abstract composite service process (ASP:Abstract Service Process), which satisfying some function, consists of m activities, simply described as $ASP = \{a_1, a_2, \ldots, a_m\}$. For each activity a_i, it owns n candidate atomic services in the service library, which can perform its function. Concretely, these atomic services can be formally defined as $S_i = \{s_{i1}, s_{i2}, \ldots, s_{in}\}$. Without the loss of generality, we assume an atomic service has d-dimensional non-functional properties, formally described as $Q = \{q_1, q_2, \ldots, q_d\}$.

A service request User_Requirement is a 2-tuple User_Requirement =< *FR*, *NFR* >. *FR*=<*Acti*, *Stru*>, and *NFR*=<*Cons*, *Pref*>, where *Acti* is the set of activities and *Stru* denotes the structure information, but *Cons* provides constraints of users and *Pref* means user's special preference information, respectively. Using selection algorithms, each task of an *ASP* can be allocated to one actual service, such that the QoS of the *ASP* is optimized. Therefore, a *CSP*(*CSP*: Concrete Service Process) is a process that consists of concrete Web services. It can be formally defined as $CSP = \{ s_{1i}, s_{2j}, \ldots, s_{mk} \}$.

By the QoS aggregation model detailed in [11], we can get the quality of *CSP*. Finally, we maximize the following objective function

$$\sum_{i=1}^{d} w_i * (CSP_{QoS}^{i}) \qquad (1)$$

Where w_i is the user's preference weight for i[th] non-functional attribute, and $\sum_{i=1}^{d} w_i = 1$.

3 Algorithm description

Algorithm 1(i.e. SPFSelect) shows the concrete steps for the composite service selection. Given a service request *SR*, we judge which cluster *SR* belongs to, or which cluster it is closest to(line 5). Ideally, there may be some cluster Clu_i. As shown in Fig.1, we have established the mapping relationship between the cluster *Clu* of service requests and the reusable process fragments library. So, we can find one optimal service process fragment *sp* from the corresponding fragment set(lines 6-10), which can meet the functional and non-functional requirements. However, in the worst-case scenario, one service request maybe has no similar items among the historical service requests. In this case, we must search from the original service library. So long as user's requirements are satisfied, the shorter the length of service process is, the better the result is. Actually, we search service process fragments according to the directions on Fig.3. If under some granularity, we get a service process fragment then return immediately (lines 11-18). Line 19 implies that there is no appropriate service process fragment.

Algorithm 1. SPFSelect

```
1.  Input: service request SR, service library SL, request clusters Clu
2.  Output: a concrete service process sp
3.  Begin
4.  sp = ϕ
5.  Find SR in which cluster of Clu
6.  If SR in Clu_i then
7.      Find corresponding reusable service process set sps
8.      Find the most matching service process sp from sps
9.      Return sp
10. End If
11. i=1
12. While i ≤ Length_max do
13.     If ssp match SR and ssp with length i then
            Assign ssp to sp
14.         Return sp
15.     Else
16.         i++
17.     End If
18. End While
19. Return sp
20. End Begin
```

4 Experimental analysis

4.1 Experiment setup

In this section, we will use some experiments to evaluate our approach. Based on GP[13] and ABC[14-15], [10] proposed the improved PM-GP and PM-ABC. Similarly, we present SPF-GP and SPF-ABC in this paper, which is a perfect combination of GP, ABC and SCKY. Next, we will compare them. We employ the QWS dataset [16] which includes a set of 2,507 Web services. For each Web service, there are nine observed QoS attributes. In this paper, we choose five most commonly used QoS attributes, i.e., response time, availability, throughput, successability, and reliability. Because the dataset is small, the number of candidate services divided equally to each activity is naturally small. We have randomly generated a dataset with 30,000 candidate services, whose QoS values are within the {max,min} QoS value ranges of the QWS dataset.

All experiments are implemented in Java. The hardware environment is a machine with the Intel(R) Core(TM) i5 CPU 760, 2.80 GHz, and 4 GB RAM running Windows 7 (64-bit).

4.2 Experiment results

In the first set of experiments, we focus on the optimality of SPF-GP and SPF-ABC(i.e., given a query condition, we calculate the ratio of current solution relative to the quality of global optimal solution), where the number of tasks for each composite service is 10, and there are 20 historical service request clusters. Under different conditions, we compare the scalability of the above four algorithms. We aim to find out the performance variation as the size of candidate service set gradually increases. As we can see from Fig. 4, all optimality of the four algorithms grows when the size of candidate service set varies from 200 to 1000. However, the improvement of SPF-GP is better than PM-GP, and SPF-ABC better than PM-ABC.

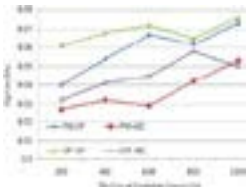


Fig. 4. Optimality Comparison

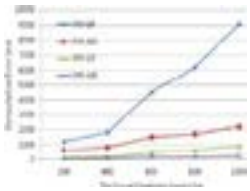


Fig. 5. Time cost Comparison

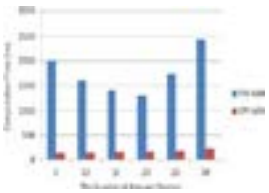


Fig. 6. Time cost VS request clusters

Then, still utilizing the above experimental setting, we compare the time cost of the algorithms. As showed in Fig. 5, the computation time of both SPF-GP and SPF-ABC reduce significantly comparing with the PM-GP and PM-ABC. It achieves considerable improvement by 65.28% and 52.38% respectively.

Finally, we compare the time performance for SPF-ABC and PM-ABC under the experimental setting that number of tasks 10, size of candidate service set 1000, and history solutions 1000. From Fig. 6, we can infer that the computation time of SPF-ABC grows slowly and significantly less than PM-ABC's(all below 250 milliseconds). Moreover, when the number of request clusters varies from 5 to 30, the time performance of PM-ABC is unstable.

5 Conclusion

In this paper, we first dissect and find the distribution of customer requests and service solutions. Then, based on our previous work, we propose a method for searching service process fragments, which contains the request solution mapping relationships between request clusters and service patterns based on statistical method. Experiments show that our method performs well.

References:

[1] X. Chen, Z. Zheng, X. Liu, Z. Huang, and H. Sun, "Personalized qos aware web service recommendation and visualization," IEEE Transactions on Services Computing, vol. 6, no. 1, pp. 35–47, 2013.

[2] Q. Z. Sheng, X. Qiao, A. V. Vasilakos, C. Szabo, S. Bourne, and X. Xu, "Web services composition: A decade's overview," Information Sciences, vol. 280, pp. 218–238, 2014.

[3] S.-Y. Hwang, C.-C. Hsu, and C.-H. Lee, “Service selection for web services with probabilistic QoS,” IEEE Transactions on Services Computing, vol.8, no.3, pp. 467–480, 2015.
[4] W. Ahmed, Y. Wu, and W. Zheng, “Response time based optimal web service selection,” IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 2, pp. 551–561, 2015.
[5] Z. Zheng, H. Ma, M. R. Lyu, and I. King, “Qos-aware web service recommendation by collaborative filtering,” IEEE Transactions on Services Computing, vol. 4, no. 2, pp. 140–152, 2011.
[6] L. Wenmin, D. Wanchun, L. Xiangfeng, and J. Chen, “A history recordbased service optimization method for QoS-aware service composition,” in Proc. of the 2011 IEEE International Conference on Web Services(ICWS 2011), July 2011, pp. 666–673.
[7] X. Wang, W. Niu, G. Li, X. Yang, and Z. Shi, “Mining frequent agent action patterns for effective multi-agent-based web service composition,” in The 7th International Workshop on Agents and Data Mining Interaction (ADMI 2011), May 2011, pp. 211–227.
[8] B. Upadhyaya, R. Tang, and Y. Zou, “An approach for mining service composition patterns from execution logs,” Journal of Software: Evolution and Process, vol. 25, no. 8, pp. 841–870, 2013.
[9] Yang R, Li B, Wang J, et al. Reusing Service Process Fragments with a Consensus between Service Providers and Users[J]. Chinese Journal of Electronics, 2016, 25(CJE-4): 648-657.
[10] L. Ruilin, X. Xiaofei, W. Zhongjie, et al, “Probability matrix of request-solution mapping for efficient service selection,” in Proc. of the 2017 IEEE International Conference on Web Services(ICWS 2017), June 2017, pp. 444–451.
[11] Yang R, Li B, Wang J, et al. SCKY: A Method for Reusing Service Process Fragments[C], in Proc. of the 2014 IEEE International Conference on Web Services(ICWS 2014), June 2014, pp.209-216.
[12] E. Al-Masri and Q. H. Mahmoud, “Investigating web services on the world wide web,” in Proc. of the 17th International Conference on World Wide Web (WWW 2008), April 2008, pp. 795–804.
[13] L. Zeng, B. Benatallah, A. H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, “Qos-aware middleware for web services composition,” IEEE Transactions on Software Engineering, vol. 30, no. 5, pp. 311–327, 2004.
[14] X. Xu, Z. Liu, Z. Wang, Q. Z. Sheng, J. Yu, and X. Wang, “S-ABC: A paradigm of service domain-oriented artificial bee colony algorithms for service selection and composition,” Future Generation Computer Systems, vol. 68, pp. 304–319, 2017.
[15] X. Wang, X. Xu, Q. Z. Sheng, Z. Wang, and L. Yao, “Novel Artificial Bee Colony Algorithms for QoS-Aware Service Selection,” IEEE Transactions on Services Computing. [Online]. Available: http://dx.doi.org/10.1109/TSC.2016.2612663.
[16] E. Al-Masri and Q. H. Mahmoud, “Investigating web services on the world wide web,” in Proc. of the 17th International Conference on World Wide Web (WWW 2008), April 2008, pp. 795–804.

An approach to the mobile social services recommendation algorithm based on association rules

Mingjun Xin*, Wenfei Liang and Jie Shu

School of Computer Engineering and Science

Shanghai University

Shanghai, 200444, China

Email: xinmj@shu.edu.cn

Email: wenfei@shu.edu.cn

Email: sujay@t.shu.edu.cn

*Corresponding author

Abstract: With the continuous development of social networks, a lot of research hot spots around it have been studied in depth. A social networking recommendation has always been a research hotspot. Time and location are intrinsic factors of the data. Adding a certain tense constraint and describing the geographical location when mining the association rules will make the rules better describe the objective reality and thus be more valuable. In this paper, it studies the association rules in the mobile social network recommendation algorithm. By introducing the mobile user's location information to the collaborative filtering recommendation process, the association rules between the items are minded. Then the association rules are filtered and split, which is integrated into the similarity matrix to make recommendation algorithm based on location information combined with association rules. The experiments show that it performs well.

Keywords: association rules; Similarity; Collaborative filtering; Recommendation algorithm.

Biographical notes: Mingjun Xin is a Full Professor in the School of Computer Engineering and Science, Shanghai University. He received his PhD degree from School of Computer Science and Technology, Northwestern Polytechnical University, Xi'an China in 2001. His research interests are mobile service recommendation, LBSN, web service computing, group social computing intelligence and software engineering et al.

Wenfei Liang is an MS student in the School of Computer Engineering and Science, Shanghai University. He received his BS degree from the School of Computer Science and Technology, Nanchang University, China in 2015. His research interests are recommendation system, mobile platforms.

Jie Shu is an MS student in the School of Computer Engineering and Science, Shanghai University. He received his BS degree from the School of Computer Science and Technology, Anhui Polytechnic University, China in 2016. His

research interests are recommendation system, mobile platforms.

1 Introduction

With the rapid development of network, social network has attracted more and more attention from researchers. Shopping online becomes more popular with the development of e-commerce, such as the foreign famous amazon.com, domestic taobao.com, jd.com, etc. However, it is difficult for users to make a choice facing the large amount of data. The appearance of recommendation system solves this problem. It makes the user spend less time when faced with a wide variety of goods. At the same time, it improves the shopping experience.

In a social network, the general users have their own social circle of friends, mostly the friends of realistic society or the people with the same interests, such as online shopping experience, users can refer to a friend's shopping for your next shopping choices. However, in the current recommendation algorithm, the recommended accuracy is still to be improved for the accuracy cannot fully meet the needs of people. Therefore, the analysis of users' preferences, shopping's behavior and so on makes more accurate recommendations and becomes the hot spot of current research. In the recommendation of shopping, the association rule is an important recommendation algorithm. Because of the inconsistency of the project distribution in the transaction database, the accuracy of this algorithm needs to be improved.

Among the researches on recommendation in the social network, Saida et al. [1] thoroughly analyzed the advantages and disadvantages of the Apriori algorithm. Besides, they analyzed the function of the Hadoop and MapReduce processes and compared the performance of the algorithms with the experimental results of different data sets applied to traffic accidents. A Dahbi et al. [2] proposed a method based on the K-means algorithm to classify and store association rules, they divide the association rules into K separate clusters, and then classify the resulting clusters from the best to the worst. Experiments on numerous datasets show that the proposed method has significant performance and effectively classifies association rules. Y Wang et al. [3] discussed and studied a personalized recommendation system based on sequence association rule mining and proposed a time association rule algorithm. Taking the online shopping customer sales data as an example, the system was empirically analysed and the algorithm is also applicable. GCLan et al. [4] proposed a new rule by introducing a new concept of temporal

association rule mining, which adopts the hierarchical structure of temporal particles, and the life of the project is also considered. The main goal of V Radhakrishna et al. [5] was to find a temporary abnormal pattern from a time database in a single database scan, without multiple scans of the database. Given the reference support time sequence for a finite time slot in the study, the goal is to find all temporal patterns that differ from the reference time vector, which reduces the space and time complexity compared to the traditional method.

Among the above researches, although many improvements are put forward in different aspects, the accuracy is not high when making recommendation in the social network. In this paper, the main research work is listed as follows: first of all, based on the definition of association rule, a weighted association rules mining model is used in the experiment, and the way of mining frequent itemsets is improved according to the project and time. Secondly, it proposed an algorithm of weighted association rules which solve the problem of data sparsity by association rule filtering and resolution. Besides, the location information is added to the score matrix used in collaborative filtering algorithm to form a three-dimensional matrix. Finally, the experiments are conducted to demonstrate the effectiveness of our recommendation algorithm.

2 Algorithm design and implementation

2.1 A new weighted formula based on time span.

According to the time, the movie data item is divided into several parts *t1... tn*, for each time period, to give its corresponding initial weights *gi gi* range between 0 and 1, so that it can be used as a weight to calculate a project's support, the specific formula is as follows:

$$\sup(X) = \frac{\sum_{i=1}^{n}(gi*Xi)}{N} \quad (1)$$

In the above formula (1), sup(X) is the support degree of a project, *Xi* is the specific item *X* in each time period of segmentation, and *gi* is the corresponding weight value, and *N* is the total number of projects in the database.

If the user for certain types of film score interval is always stable in a certain range, then the stability of this kind of movie is very strong, so a time span of weighted formula design, make the transaction in the database project can stability according to their respective time interval for different weights. The specific formula is as follows:

$$E = \frac{t}{n} \quad (2)$$

$$G = \frac{\sum_{i=1}^{n} e-(e-(di+1-di))}{n*e} \quad (3)$$

2.2 Association rule correction score matrix.

In this paper, the association rules are used to mine the many-to-one intrinsic links between the projects. Firstly, the new strong association rules are obtained in the process of association rule mining, and then the association rules are filtered and split. Since *IM*'s similarity is one-to-one, and the final project (user) is expected to be an indicator, there are only one-to-many (or one-to-one) relationships. Therefore, it is necessary to break up the association rules of multiple pairs (or one-to-many) into the corresponding rules of the corresponding N multiple pairs, and the splitting method is as formula (4) :

$$X \to Y = \begin{cases} x1, x2, x3, \dots, xm \to yi & Y \in \text{Multiple sets}, \ i \in [1, N] \\ x1, x2, x3, \dots, xm \to Y & Y \in \text{Multiple sets} \end{cases} \quad (4)$$

Finally, strong association rules are integrated into *IM* to construct a modified project - project similarity matrix *CM*. At the same time, the total number of *IM* columns needs to be extended to maintain a many-to-one relationship. It uses the hyperbolic tangent function to integrate the association rules into a similar matrix, and the degree of support and reliability is the main criterion to measure its similarity. *Xi* represents a single project of the lead *X*, and t represents the sum of each rule support and confidence. The AveItem(xi) represents the average preference value of the project *xi*, and the AveItem'(X) represents the average preference value of the association rule *X*. The calculation method is defined as follows:

$$sim'(X, Y) = \begin{cases} \frac{\sum_{xi \in X} sim(xi, Y)}{\sum_{xi \in X}(pearson)} * \left(1 + \frac{e^{t}-e^{-t}}{e^{t}+e^{-t}}\right) & X \in \text{Association Rules} \\ sim(X, Y) & X \in \text{Items} \end{cases} \quad (5)$$

$$AveItem'(X) = \frac{\sum_{xi \in X} AveItem(xi)}{\sum_{xi \in X}(pearson)} \quad (6)$$

$$sim(Ii, Ij) = pearson(Ii, Ij) = \frac{\sum_{k \in UAi,j}\left(Wk,i-\overrightarrow{Ii}\right)*(Wk,j-\overrightarrow{Ij})}{\sqrt{\sum_{k \in UAi,j}(Wk,i-\overrightarrow{Ii})^{2} * \sqrt{\sum_{k \in UAi,j}\left(Wk,j-\overrightarrow{Ij}\right)^{2}}}} \quad (7)$$

UAi,j represents the user collection with preference for both project *Ii* and project *Ij*; Sim(Ii,Ij) represents the similarity between project *Ii* and

project *Ij*; $\underset{\text{Ii}}{\rightarrow}$ represents the average preference values of Ii; $\underset{\text{Ui}}{\rightarrow}$ represents the average preference value of user *Ui*.

Figure 1 shows the conversion chart from IM to CM. The figure shows the new similarity weights between projects. Since the filtered association rules have two forms: one-to-one and many-to-one, the similarity in IM is also needed synchronization update. Itemn+1 represents the newly generated combination items, for example, Itemn+1 is combined by Item2 and Item3, and W'1,n+1 represents the similarity of Item1 to Item2 and Item3.

$$\begin{matrix} \text{Item1} & \text{Item2} & \dots & \text{Item n} \end{matrix} \quad \begin{matrix} \text{Item1} & \dots & \text{Item n+i} \end{matrix}$$

$$\begin{bmatrix} \text{W1,1} & \text{W1,2} & \dots & \text{W1,n} \\ \text{W2,1} & \text{W2,2} & \dots & \text{W2,n} \\ \dots & \dots & \dots & \dots \\ \text{Wn,1} & \text{Wn,2} & \dots & \text{Wn,n} \end{bmatrix} \rightarrow \begin{bmatrix} \text{W}'1,1 & \dots & \text{W}'1,\text{n}+\text{i} \\ \text{W}'2,1 & \dots & \text{W}'2,\text{n}+\text{i} \\ \dots & \dots & \dots \\ \text{W}'\text{n},1 & \dots & \text{W}'\text{n},\text{n}+\text{i} \end{bmatrix}$$

IM CM

Figure 1. transformation of IM to CM.

According to the CM matrix and Pearson similarity measure method, the final project - project similarity matrix FM can be calculated. Then，the ratings *Ij* is predicted based on the target user behaviour records the history of the *Ui*. Finally to score matrix in location information fusion filter, *Pi*, *j* said to predict the *Ui* of the *Ij* score, computation formula is listed as follows:

$$\text{Pi,j} = \text{AveItem(Ij)} + \frac{\sum_{\text{Ik}\in\text{sims}} \text{sim}'\text{(Ij,Ik)} * \left(\text{Wi,k} - \text{AveItem(Ik)}\right)}{\sum_{\text{Ik}\in\text{sims}} \text{sim}'\text{(Ij,Ik)}} \tag{8}$$

The score Pi,j is a predicted value which is depends on the similarity and weight between the items.

2.3 The scoring matrix based on u-i-l

It is a blend of location factors in the collaborative filtering recommendation, blend in geography information matrix, forming a user - based project-position (U-I-L) of the three-dimensional matrix, each latitude in their respective vector to represent the attribute value. Although the third dimension is introduced, it is beneficial to alleviate the problems such as large amount of similarity calculation and cold start-up in the process of reducing dimension.

The calculation formula of *rui*, which represents the preference of user u in a geographic location for project I with geographical location within *dexp*, is listed as follows:

$$\text{rui} = (\widehat{\text{rui}} + 1 - \frac{\text{dui}}{\text{dexp}})\text{p} \tag{9}$$

Including dui u said users and project the actual location of distance, I dexp said user u submit the location of the information, in particular geographical location distance dexp u submit the area near the location of the project sets prefiltering, dexp is an experience. The value is always be set as 500 meters, 1000 meters or 1500 meters, etc. Actually, it is related to the intensity of the project in the area and the sensitivity of users to distance. P is to revise the range of the prediction score to the distribution range of the historical score. In the experiment of the later data set, the value of *p* is 5/6.After the location distance is added, the final score prediction of project I by user u will be arranged in descending order to form the recommended list of TOP-N projects according to the project score.

3 Experimental analysis

3.1 Experimental data set

In this paper uses two data sets to combine the simulated experimental data sets.

(1) The Movie Lens data set that is provided by the Group Lens research team of the University of Minnesota. It includes 943 users with 100, 000 scores for 1682 movies. Every piece score of the movie contains the following fields:

(2) The sample data of all GPS data which is generated by 12,000 taxis in Beijing in one month. Every piece of the GPS data contains the following fields:

3.2 Experimental procedure and verification

This experiment is conducted on a combinative data set. Two different methods are compared:

CF (collaborative filtering algorithm): It makes recommendation according to a set of similar users or items.

ARRA (the proposed recommendation algorithm): It improves the method of association rules when calculating the value of support. Based on the improved association rules, the CF algorithm is modified to make the recommendation more accurate.

3.3 Evaluation benchmark

In this paper, the calculation formula is adjusted as follows:

$$\text{Accuracy} = \frac{Correct}{M} = \frac{\sum_u |R(u) \cap T(u)|}{\sum_u |R(u)|} \quad (10)$$

$$\text{Recall} = \frac{Correct}{N} = \frac{\sum_u |R(u) \cap T(u)|}{\sum_u |T(u)|} \quad (11)$$

Among them, R(u) represents the collection of N items recommended by users, and the collection of items that users like on the test set is T(u).

3.4 Experimental result

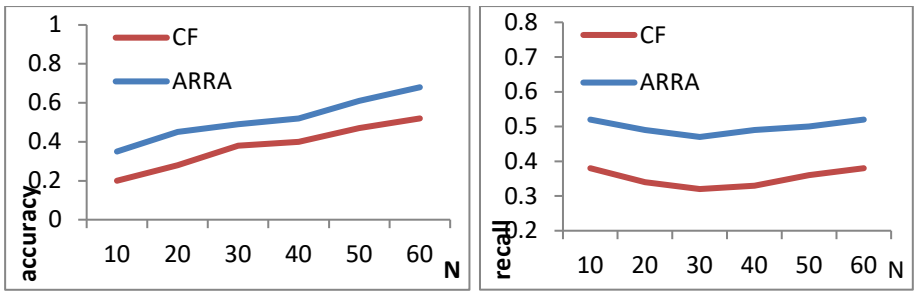


Figure 2 the accuracy and recall of the algorithms

Figure 2 shows that the algorithm proposed in this paper has better accuracy and recall than the CF. The performance of ARRA is ideal when making recommendation in the social network.

4 Conclusion

In this paper, it proposed a mobile social services recommendation algorithm based on association rule mining and collaborative filtering. A new model of association rules mining is put forward in this algorithm. Specifically, the transaction database is divided into interval segmentation database, at the same time the weight proportion is introduced and the new weight formula is put forward. The data items are two iterative weighted through association rule filtering and resolution, which alleviate the sparse matrix data problem. In addition, the location information is taken into the score matrix so that more relevant projects were excavated through the improved collaborative filtering algorithm, which further improves the precision of the recommendation.

In the future work of our research, more information will be analyzed such as user preferences, the categories of items. By analysing more factors, the more meaningful association rules can be explored which can be used to improve the performance of recommendation.

Reference

[1] Hmina N, Hmina N, Hmina N. Association rules mining on MapReduce[C]// International Conference on Big Data, Cloud and Applications. ACM, 2017:58.

[2] Dahbi A, Mouhir M, Balouki Y, et al. Classification of association rules based on K-means algorithm[C]// IEEE International Colloquium on Information Science and Technology. IEEE, 2017:300-305.

[3] Wang Yonggang.Sequential Association Rules Based on Apriori Algorithm Applied in Personal Recommendation. International Journal of Database Theory and Application Vol.9, No.6, 2016:257-264

[4] Lan G C, Hong T P, Wu P S, et al. Mining hierarchical temporal association rules in a publication database[C]// IEEE International Conference on Cognitive Informatics & Cognitive Computing. IEEE, 2013:503-508.

[5] Radhakrishna V, Janaki V, Janaki V. Mining Outlier Temporal Association Patterns[C]// International Conference on Information and Communication Technology for Competitive Strategies. ACM, 2016:105.

[6] Yu X, Miao S, Liu H, et al. Association Rule Mining of Personal Hobbies in Social Networks[J]. International Journal of Web Services Research, 2017, 14(1):13-28.

[7] Gonsalves B, Patil V. Improved web service recommendation via exploiting location and QoS information[C]// International Conference on Information Communication and Embedded Systems. IEEE, 2016:1-5.

[8] Rong Y, Wen X, Cheng H. A Monte Carlo algorithm for cold start recommendation[C]// International Conference on World Wide Web. ACM, 2014:327-336.

[9] Yin Z, Cao L, Han J, et al. Geographical topic discovery and comparison[C]// International Conference on World Wide Web. ACM, 2011:247-256.

[10] Ye M, Yin P, Lee W C, et al. Exploiting geographical influence for collaborative point-of-interest recommendation[C]// International Acm Sigir Conference on Research & Development in Information Retrieval. ACM, 2011:325-334.

MTransD: A Dynamic Relationship Construction based Approach for Multi-lingual Knowledge Graph Embedding and Alignment

Huijie Liu

School of Computer Science and Technology,
Harbin Institute of Technology Shenzhen Graduate School,
Shenzhen, China
E-mail: chaser_j@163.com

Xiaofeng Zhang

Computer Science and Technology,
Harbin Institute of Technology Shenzhen Graduate School,
Shenzhen, China
E-mail: zhangxiaofeng@hitsz.edu.cn

Yuxing Fei

School of Computer Science and Technology,
Harbin Institute of Technology Shenzhen Graduate School,
Shenzhen, China
E-mail: 2718816726@qq.com

Abstract: Recently, knowledge graph already plays an important role in various artificial intelligence applications such as service computing and question & answering. Most existing approaches are proposed to build the mono-lingual knowledge graph but not the cross-lingual knowledge graphs. In this paper, we propose the MTransD model for this issue. The proposed model makes transitions for each embedded semantic vectors to its cross-lingual space to achieve the cross-lingual knowledge graph alignment. Experiments are evaluated on the $WK3l$ data set and several state-of-the-art models are performed for model performance comparison. The promising experimental results demonstrate the superiority of the proposed MTransD model for the cross-lingual knowledge graph extraction.

Keywords: Knowledge Graph; Embedding Model; Cross-Lingual Alignment.

1 Introduction and Related Works

Recently, knowledge graphs such as WordNet(Miller (1995)) and DBpedia(Lehmann (2015)) have been proposed for various artificial intelligence applications, such as service

computing and Question and Answering. In general, knowledge graph encodes structured information of entities and relations, and a typical knowledge graph usually contains millions of entities and billions of relations. In the past decades, many research works have been proposed which are mainly based on symbol and logic inferring to build a complete knowledge graph. However, these works can not work well to build a large-scale knowledge graph. Recently, an approach is proposed for this task which embeds an element (entity and relation) into a low-dimensional embedding vector space, called knowledge graph embedding. On top of this work, TransE (Bordes et al. (2013)) is proposed which learns vector embedding for both entities and relations. However, TransE can only cope with the 1-to-N mapping situation. Consequently, the TransH (Wang et al. (2014)) is proposed to allow an entity to have different representation forms if involved in more than one relation. All these approaches embed the entities and relations into the same vector space, which ignore the diversity of entities. Lin et al. (2015) proposed TransR which models the entities and relations into two separate vector space and later TransD (Ji et al. (2015)) is proposed. Unfortunately, the focus of most existing approach is how to build mono-lingual knowledge graph. In era of big data, it is urgently needed to address the following issues such as how to align cross-lingual knowledge graphs. For all these tasks, the alignment and fusion of cross-lingual knowledge graphs is the basic one. The existing difficulties might be as follows. It is required to cope with more personalized multi-lingual knowledge graphs. The knowledge base could be used to construct different language knowledge graphs which are hard to acquire. Due to the diversified expression manner, the knowledge bases of different languages may be quite different which makes the model learning process very hard. In this paper, we propose a cross-lingual knowledge graph alignment and fusion model called MTransD consisting of two parts: knowledge embedding and alignment model.

2 The Proposed MTransD Approach

2.1 Problem Formulation

Let $2n$ vectors represent the entity $e_{L_1}, e_{L_2}, ..., e_{L_n}$ and relation $r_{L_1}, r_{L_2}, ..., r_{L_n}$, having the same meanings in different languages $L_1, L_2, ..., L_n$. Among them, let **e** or **r** denote the semantic vector in each language, representing the common knowledge shared by a set of entities or relations. The rest n vectors $e_{p_1}, e_{p_2}, ..., e_{p_n}$ or $r_{p_1}, r_{p_2}, ..., r_{p_n}$ is called space vector corresponding to the language used to construct the vector relation space. Let KB denote the knowledge base, L denote a language, and G_L denote the knowledge graph extracted for language L. E_L and R_L respectively denote the entity set and relation set. In the knowledge graph G_L, a piece of knowledge is represented by a triplet $T = (h, r, t), T \in G_L, h, t \in E_L, r \in R_L$, where h, t are entities and r is the relation.

In MTransD model, Both entities and relations are represented in two separate vectors. The corresponding vector of $T = (h, r, t)$ is $\boldsymbol{h, h_p, r, r_p, t, t_p}$, where $\boldsymbol{h, r, t}$ are called semantic vectors which are then used to represent the semantic meanings of entity h, t and relation r. $\boldsymbol{h_p, r_p, t_p}$ are called space vectors which are used to dynamically build spatial relation between vectors. $\boldsymbol{h, h_p, t, t_p} \in R^n$ are actually embedded in the entity space, and $\boldsymbol{r, r_p} \in R^m$ are embedded in the relation space.

For each triplet (h, r, t), $\boldsymbol{M_{rh}}$,$\boldsymbol{M_{rt}} \in R^{m \times n}$ project the entity embedding vectors $\boldsymbol{h, t}$ into the relation space. The projection matrix $\boldsymbol{M_{rh}}$,$\boldsymbol{M_{rt}}$ are calculated as follows

$$M_{rh} = r_p h_p^T + I^{m \times n} \tag{1}$$

$$M_{rt} = r_p t_p^T + I^{m \times n} \tag{2}$$

Given a language knowledge graph G_L, its loss function if computed for multi-lingual knowledge graph can be defined as,

$$\ell_{K_L} = \sum_{(h,r,t) \in G_L} f_r(h,t) \tag{3}$$

2.2 Align Multiple Knowledge Models

After acquiring multiple knowledge models, we need to align these models in a cross-lingual manner. Given a pair of languages (L_1, L_2), $\delta(L_1, L_2)$ is defined as its alignment set containing all the aligned triplets $\{(T_{L_1}, T_{L_2}) | T_{L_1} \in G_{L_1}, T_{L_2} \in G_{L_2}, T_{L_1} \longleftrightarrow T_{L_1}\}$ between two knowledge graphs G_{L_1} and G_{L_2}. For example, $((China, capital\_city, Beijing), (\text{"中国","首都","北京"})) \in \delta(English, Chinese)$ is a triplet in the alignment set between English and Chinese. For a triplet (T_{L_1}, T_{L_2}) in the alignment set, we have $T_{L_1} = (h_{L_1}, r_{L_1}, t_{L_1}),\ T_{L_2} = (h_{L_2}, r_{L_2}, t_{L_2})$. The semantic vector $\boldsymbol{h_{L_1}, t_{L_1}, r_{L_1}, h_{L_2}, t_{L_2}, r_{L_2}}$ and the space vector $\boldsymbol{h_{pL_1}, t_{pL_1}, r_{pL_1}, h_{pL_2}, t_{pL_2}, r_{pL_2}}$ are mapped to each other. Consequently, the corresponding alignment equations could be written as,

$$r_{pL_1}(h_{pL_1}^T H_{L_1} - t_{pL_1}^T t_{L_1}) \approx r_{pL_2}(h_{pL_2}^T H_{L_2} - t_{pL_2}^T t_{L_2}) \tag{4}$$

$$(r_{pL_1} h_{pL_1}^T - r_{pL_2} h_{pL_2}^T)\overline{h} \approx (r_{pL_1} t_{pL_1}^T - r_{pL_2} t_{pL_2}^T)\overline{t} \tag{5}$$

The loss function of the alignment model is defined as follows,

$$\ell_{A_{\{L_1,L_2\}}} = \sum_{(T_{L_1}, T_{L_2}) \in \delta\{L_1,L_2\}} \ell_a(T_{L_1}, T_{L_2}) \tag{6}$$

2.3 Multiple Knowledge Models Fusion

Ideally, entities or relations with the same semantics in multi-lingual knowledge graphs should have the same semantic vectors in the embedding vector space. However, it is hard to train the unbiased cross-lingual knowledge graphs as well as the matching semantic vectors. Therefore, similar vectors might be fused together to approximate the matching semantic vectors in the cross-lingual knowledge graphs. In the proposed MTransD model, for entities e_{L_1}, e_{L_2} in the knowledge graphs of two languages L_1, L_2, the relation between the entities should be first aligned according to their similarity between the semantic vectors. Let $sim(e_{L_1}, e_{L_2})$ denote the similarity of the semantic vectors of entity e_{L_1}, e_{L_2}, which is calculated as,

$$sim_1(e_{L_1}, e_{L_2}) = \frac{e_{L_1} \cdot e_{L_2}}{\|e_{L_1}\|_2 \|e_{L_2}\|_2} \tag{7}$$

$$sim_2(e_{L_1}, e_{L_2}) = \|e_{L_1} - e_{L_2}\|_2 \tag{8}$$

$$sim_3(e_{L_1}, e_{L_2}) = \|e_{L_1} - e_{L_2}\|_1 \tag{9}$$

where $sim_1(e_{L_1}, e_{L_2})$ adopts the cosine distance to measure the similarity. The greater the value of the function, the higher the similarity of the entities. The $sim_2(e_{L_1}, e_{L_2})$ and $sim_3(e_{L_1}, e_{L_2})$ measure the similarity of semantic vectors using Euclidean distance and Manhattan distance, and its value falls into $[0, 2]$. The smaller the value of the function, the higher the similarity of the entities.

3 Experimental Results

Experiments will be performed to evaluate the effect of cross-lingual alignment. The data set used in the experiments is a public one called $WK3l$. $WK3l$ contains knowledge graphs of English, French and German in the dbo:Person domain of DBpedia. There are are 2,534,869 triplets in the data set. And a labeled cross-lingual alignment is provided for some of the cross-lingual triples, such as English and French, English and German alignment set. These data can be seen as the training data set with its total number is 193,846. In addition, the data set also provides a test set for model evaluation which contains 97,414 aligned entity pairs. The test set includes the aligned cross-lingual entity pairs, which are divided into four groups, i.e., English to French, French to English, English to German, and German to English. In this experiment, both the MTransD model and several multi-lingual knowledge graph embedding and alignment models are evaluated such as **MTransE** by Chen et al. (2017), **CCA** by Faruqui and Dyer (2014), **LM** by Mikolov et al. (2013) and **OT** by Xing et al. (2015). The dimension of space vector is set as $m, n = 75$, and the learning rate and epoch is set to 0.01 and 400. The evaluation criterion is the widely adopted "Hits@10". There are five scoring functions defined in the original MTransE model, and three scoring functions for the proposed MTransD. The mean and variance value of aligned entities calculated using each evaluated model are recorded in the table.

Table 1 Comparison Results by using criterion Hits@10

Model	English-French	French-English	English-German	German-English
LM	12.31	10.42	22.17	15.21
CCA	20.78	19.44	26.46	22.30
OT	44.97	40.92	44.47	49.24
$MTransE - Var_1$	51.05	46.64	48.67	50.60
$MTransE - Var_2$	45.25	41.74	46.27	49.00
$MTransE - Var_3$	38.64	36.44	50.82	52.16
$MTransE - Var_4$	59.24	57.48	66.25	68.53
$MTransE - Var_5$	59.52	57.07	60.25	66.03
$MTransD - Var_1$	54.38	50.09	56.16	60.55
$MTransD - Var_1$	64.62	**68.93**	**72.80**	73.10
$MTransD - Var_1$	**70.53**	66.67	70.69	**75.39**

In table 1, the last three columns are the proposed MTransD model with different scoring functions where Var_i denotes the i-th scoring function. A higher Hits@10 value means the calculated results are more accurate. From the results in table 1, it is obvious that the proposed MTransD model can achieve the best model performance. Among all evaluated models, $MTransD - Var_2$ performs the best in two cross-lingual alignment task, whereas the proposed $MTransD - Var_3$ performs the best in the rest two cross-lingual alignment task. However, for "French-English" and "English-German" alignment tasks, the results of the $MTransD - Var_3$ are comparable to the proposed $MTransD - Var_2$ which indicates that the effect of these two loss function are interchangeable and is better than the first loss function. On the average, the original MTransE is better than the rest three models and is 8.97% lower than the proposed MTransD in this task.

4 Conclusion and Future Work

Nowadays, knowledge graph already plays an important role and is widely adopted by various research applications. However, most knowledge graph models are built for mono-lingual language and thus is not suitable for the cross-lingual situation. To this end, we

proposed the MTransD model, a novel multi-lingual knowledge graph embedding and alignment model, in this paper. Particularly, we employ two separate vectors (semantic and space vectors) to represent entities and relations, and propose three loss functions to measure the discrepancy between several mono-lingual knowledge graphs and align these models accordingly. In the experiments, we extensively evaluate the proposed MTransD with several state-of-the-art models. The promising results have demonstrated the superiority of the proposed model in terms of mean, variance and Hits@10 criteria.

References

Miller, G.A. (1995) 'WordNet: a lexical database for English', *Communications of the Acm*, Vol. 38, No. 11, pp.39–41

Lehmann, J. (2015) 'DBpedia: A large-scale, multilingual knowledge base extracted from wikipedia', *Semantic Web*, Vol. 6, No. 2, pp.167–195

Bordes, A., Usunier, N., Garcia-Duran A., Weston, J. and Yakhnenko, O. (2013) 'Translating Embeddings for Modeling Multi-relational Data', *Proceedings of the 26th International Conference on Neural Information Processing Systems*, pp.2787–2795

Wang, Z., Zhang, J., Feng, J. and Chen, Z. (2014) 'Knowledge graph embedding by translating on hyperplanes', *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, pp.1112–1119

Lin, Y., Liu, Z., Sun, M., Liu, Y. and Zhu, X. (2015) 'Learning entity and relation embeddings for knowledge graph completion', *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pp.2181–2187

Ji, G., He, S., Xu, L., Liu, K. and Zhao, J. (2015) 'Knowledge Graph Embedding via Dynamic Mapping Matrix', *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp.687–696

Chen, M., Tian, Y., Yang, M. and Zaniolo, C. (2017) 'Multilingual Knowledge Graph Embeddings for Cross-lingual Knowledge Alignment', *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp.1511–1517

Faruqui, M. and Dyer, C. (2014) 'Improving Vector Space Word Representations Using Multilingual Correlation', *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pp.462–471

Mikolov, T., Le, Q.V. and Sutskever, I. (2013) 'Exploiting Similarities among Languages for Machine Translation', *Computer Science*

Xing, C., Wang, D., Liu, C. and Lin, Y. (2015) 'Normalized Word Embedding and Orthogonal Transform for Bilingual Word Translation', *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp.1006–1011

A Topic-Enhanced Recurrent Autoencoder Model for Sentiment Analysis of Short Texts

Shaochun Wu, Ming Gao, Qifeng Xiao, Guobing Zou

School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China

Abstract: This paper presents a topic-enhanced recurrent autoencoder model to improve the accuracy of sentiment classification of short texts. First, the concept of recurrent autoencoder is proposed to tackle the problems in recursive autoencoder including "increasing in computation complexity" and "ignoring the natural word order". Then, the recurrent autoencoder model is enhanced with the topic and sentiment information generated by Joint Sentiment-Topic (JST) model. Besides, in order to identify the negations and ironies in short texts, sentiment lexicon is utilized to add feature dimensions for sentence representations. Experiments are performed to determine the feasibility and effectiveness of the model. Compared with recursive autoencoder model, the classification accuracy of our model is improved by about 7.7%.

Keywords: short texts, sentiment analysis, word embedding, recurrent autoencoder, recurrent neural network, recursive autoencoder, joint sentiment-topic model

1 Introduction

In this paper, we present a hybrid approach which combines machine learning and lexicon-based methods and utilizes the topic information on short texts to achieve a better performance in sentiment classification.

2 Related Work

Based on the recursive autoencoder model, this paper proposes the recurrent autoencoder model to learn word embeddings according to the natural word order and reduce the computation complexity. Furthermore, the recurrent autoencoder is improved by learning the word embedding with the supervision of topic and sentiment information. Besides, we add sentiment feature dimensions for sentence representations with lexicon to identify negations and ironies so that the accuracy of sentiment classification can be improved.

3 Topic-Enhanced Recurrent Autoencoder Model

In order to solve these two issues of Socher's recursive autoencoder model: "increasing in computation complexity" and "ignoring the natural word order", we first introduce traditional recurrent neural network, and then put forward the recurrent autoencoder model to learn the word embedding. Finally, we discuss how to improve the recurrent autoencoder model with topic information.

3.1 Traditional Recurrent Neural Network

A recurrent neural network and the unfolding in time of the computation involved in its forward computation are shown in Figure 1.

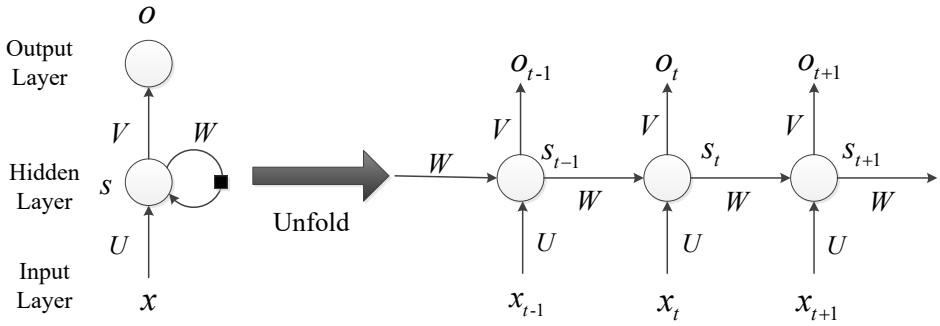


Fig. 1 A recurrent neural network and the unfolding in time of the computation involved in its forward computation

3.2 The Concept of Recurrent Autoencoder Model

A recurrent autoencoder model and the unfolding in time of the computation involved in its forward computation are shown in Figure 2.

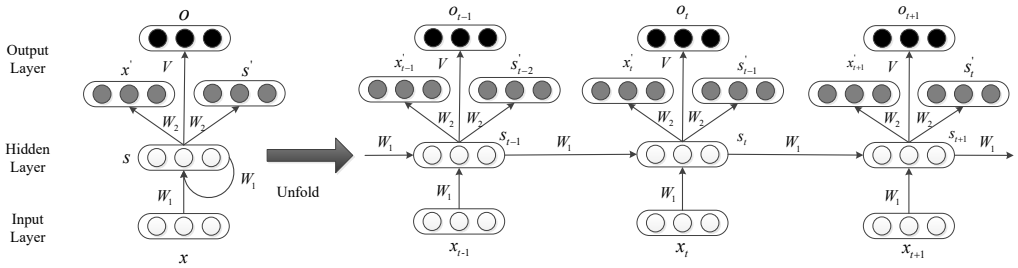


Fig. 2 A recurrent autoencoder model and the unfolding in time of the computation involved in its forward computation

3.3 Topic-Enhanced Recurrent Autoencoder Model

We design a topic-enhanced recurrent autoencoder (TRAE) model for sentiment analysis of short texts. First, we use JST model to generate the topic-sentiment combined distribution, and then utilize it to conduct the supervised learning of word embeddings.

3.3.1 Calculate the topic-sentiment combined distribution with JST model

In this paper, we use the JST model to generate the topic-sentiment combined distribution.

3.3.2 Learning word embeddings over topics and sentiments

After the topic-sentiment combined distribution for each document is obtained, we use it to conduct the learning of word embeddings. In our model, there is a softmax layer as output over each hidden layer.

We can back propagate errors to influence the model parameters and learn word embedding. When the model parameters are stable, the representation of a sentence will be obtained.

4 The Union Model for Sentiment Analysis of Short Texts

The last merged word embedding generated by TRAE model, namely the sentence representation, can be effectively used for sentiment analysis. However, to further improve the performance, we design a union model for sentiment analysis of short texts.

4.1 Advanced sentence representation combined with sentiment lexicon

In order to identify negations and ironies of short texts effectively, we improve the sentence representation by adding feature dimensions with sentiment lexicon.

4.2 Naive Bayes classifier

Naive Bayes classifier, a classical algorithm for classification, has been widely used in many fields such as image processing, data mining and so on. Especially, it has a great performance in the text classification. In this paper, we use it to classify the sentiment polarity of the advanced sentence representation.

5 Experimental Design and Resultant Analysis

5.1 Data Preparation

In the experiment, we adopted two publicly available datasets: Stanford Twitter Sentiment(STS) and SemEval-2013~2015 [11].

5.2 The Design of Experiments

In order to test the performance of the topic-enhanced recurrent autoencoder in sentiment analysis of short texts, we adopted a series of contrast experiments.

5.3 Result Analysis

5.3.1 Time complexity

In the experiment, we compared the time complexities of recursive autoencoder, recurrent neural networks and recurrent autoencoder, respectively.

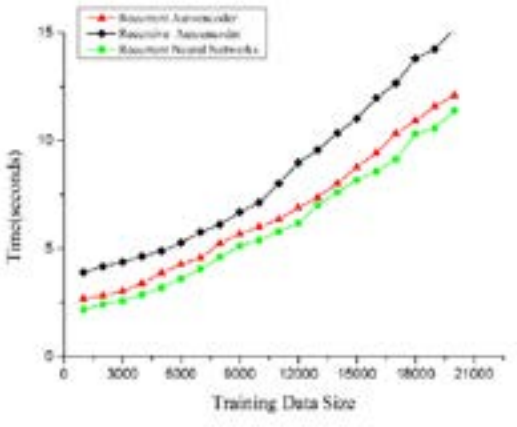


Fig. 5 Time complexities of different models corresponding to the training data size

5.3.2 The accuracy of sentiment classification

In order to prove that the sentence representation built by the recurrent autoencoder model has a higher quality than that of the recursive autoencoder and the RNNs, we used the test sets with different n-gram length to take a contrast experiment of the classification accuracy.

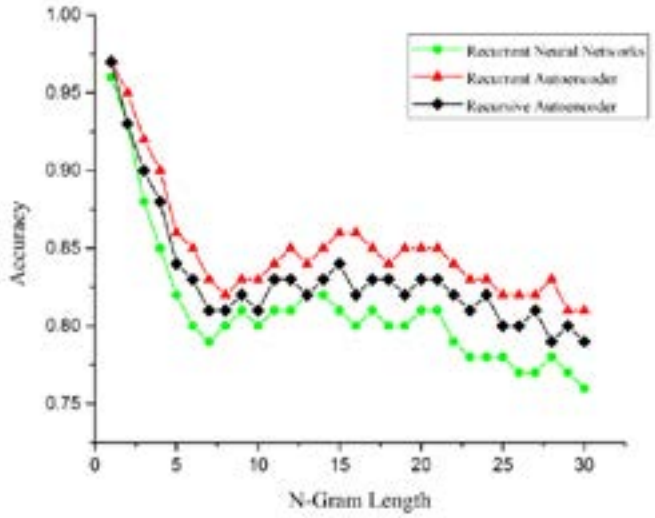


Fig. 6 The accuracies of different models with different n-gram length

6 Conclusion

In this paper, a topic-enhanced recurrent autoencoder model for sentiment analysis of short texts is proposed to improve the accuracy of sentiment classification.

Acknowledgement

The authors thank to all anonymous reviewers for their insightful comments and useful suggestions.

References

[1] Giachanou A, Crestani F. Like it or not: A survey of twitter sentiment analysis methods[J]. ACM Computing Surveys (CSUR), 2016, 49(2): 28.
[2] Wu W, Li H, Wang H, et al. Probase: A probabilistic taxonomy for text understanding[C]//Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data. ACM, 2012: 481-492.

A Construction and Self-Learning Method for Intelligent Domain Sentiment Lexicon

Shaochun Wu, Qifeng Xiao, Ming Gao, Guobing Zou

School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China

Abstract: A new method of building intelligent sentiment lexicon based on LDA and word clustering is put forward in this paper. In order to make seed words more representative and universal, this method uses LDA topic model to build the term vectors and select seed words. The improved SO-PMI algorithm has been used to calculate the emotional tendency of each sentiment word. In addition, the domain sentiment lexicon's automatic extension and update method is designed to deal with dynamic corpus data. Experiments show that the proposed method can build the sentiment lexicon with higher accuracy, and can reflect the change of words' emotional tendency in real time. It is proved in this paper that this method is more suitable for processing a large number of dynamic Chinese texts.

Keywords: Sentiment Lexicon, SO-PMI algorithm, seed words, LDA Topic Model, word clustering, incremental text processing

1 Introduction

During the past decade, with the widespread use of web sites and social media, a large number of comment texts are generated. These comments reflect the commenters' feelings about certain person, event and commodity. They can exert the orientation of public opinion clearly. Sentiment analysis of these text data is of great scientific and commercial value. Sentiment analysis is refered to the analysis and reasoning of subjective text with emotional content.

The most common way to build a sentiment lexicon is to calculate the PMI (Pointwise Mutual Information) between unknown emotional words and known words and then using

the calculated PMI to measure the emotional tendency of each unknown emotional word. Using this method to construct the sentiment lexicon can match the basic demand for sentiment analysis. Yet there are some problems which could always lead to low accuracy such as seed words selection strategies, the complex association relationship between words, data sparse and so on. Furthermore, the traditional methods of building sentiment lexicon are mostly used to train samples in static historical data monolithically to obtain emotional information of unknown sentiment words.

To overcome these problems, in this paper, we provide a construction and self-learning method for intelligent domain sentiment lexicon. First, the seed words set is constructed by word clustering methods based on LDA topic model. Second, synonym group mining is imported to improve the accuracy of traditional SO-PMI algorithm. Then the initial domain sentiment lexicon is constructed. And as the expanding of corpus, the lexicon can constantly adapt to the new text data, maintaining high coverage and accuracy.

2. Related Work

Lexicons play an important role in the study of textual sentiment analysis.The researchers who studied sentiment lexicons mostly focused on calculation of words' sentiment tendency. Most of them did not improve sentiment lexicon's ability of reflecting the change of sentiment words in dynamic corpus, and the specific sentiment tendency of uncommon sentiment words in certain domain areas.

In order to handle these shortcomings, in this paper, we presents a construction and self-learning method of sentiment lexicon for dynamic data and specific domain. In the case of high accuracy, self-learning and self-expansion are implemented for dynamic data and in specific domain language environments.

3. Extraction of Seed Words based on Word Clustering

Seed words refers to several words with strong commendatory or derogatory tendency in known vocabulary. Seed words are often very representative among words of the same emotional tendency. A word set's coverage, emotional intensity and distinguishing ability must be considered before being selected as seed words set. A seed words set must have some characteristics including a wide range of uses, small similarity between different members and strong emotional tendencies [1].

In order to make sure that the seed words have a very obvious commendatory or derogatory tendency and strong representative characteristics without affecting the co-occurrence statistics, we use word clustering to cluster the known sentiment words, and select several clusters' center as seed words.

We use the K-Means ++ algorithm to select the center of each cluster. K-Means++ algorithm mainly focuses on the improvement of the clustering center initialization. It has a more scientific method to select the cluster center instead of random selection. Since we choose k cluster centers as k seed words, we only need to complete the process of clustering center initialization of K-Means ++ algorithm, then we can get k positive seed words, and construct a positive seed words set.

4. Construction of Initial Sentiment Lexicon Based on Improved SO - PMI Algorithm

After selecting enough reasonable seed words, we need to use the SO-PMI algorithm to calculate the emotional tendency of unknown sentiment word.

The SO-PMI algorithm mainly evaluates the similarity between 2 words by calculating the PMI value between them, and evaluates the emotional tendency of the candidate word by whether the PMI between it and the commendatory is larger than that between it and the derogatory seed words.

However, the accuracy of the result is not satisfactory enough in practice. The reason is that the degree of contact between 2 words cannot be determined only by whether they appear in a single sentence together. In actual calculation process, we often face the problem of data sparseness, where we use the co-window and synonym group to improve the accuracy of PMI calculation.

Not only that, the co-distance between two words can reflect the degree of correlation between those two words, so this paper imports co-window and co-distance into SO-PMI algorithm to improve the accuracy.

We use synonyms to improve the lexicon construction method. For the unknown sentiment word w, whose occurrences are fewer than the threshold m, we find the synonym group of w in the synonym sources, and convert the SO-PMI calculation of w to the SO-PMI calculation of its synonym group.Because the construction of the synonym group is very similar with the word clustering process, we generate the word eigenvector with the LDA "word - topic" matrix. Then we use cosine similarity to calculate the similarity between different words, and select the n words most similar with each word as its synonym group.

5. The Self - learning Process of Intelligent Domain Sentiment Lexicon

In order to make the Domain Sentiment Lexicon (DSL) be able to adapt to the rapidly changing language environment in the big data environment, the Intelligent Domain Sentiment Lexicon (IDSL) needs to be able to perceive the language changes from the newly loaded new corpus data [2].

During the update process, assume that there are n new documents updated, we preprocess these n documents, statistic candidate words' frequency and update co-occurrence information between candidate words and seed words. While the data update is completed, the emotional polarity of the candidate words is updated by the following formula to update the emotional tendencies in the sentiment lexicon:

$$PMI(w_1, w_2) = \log \frac{(N + \Delta N) \times (C(w_1 \& w_2) + \Delta C(w_1 \& w_2))}{(C(w_1) + \Delta C(w_1)) \times (C(w_2) + \Delta C(w_2))}$$

where ΔN represents the number of new text, $\Delta C(w_1)$、$\Delta C(w_2)$、$\Delta C(w_1 \& w_2)$ are the growth of candidate words, seed words and the co-occurrences of them, respectively.

In the process of updating the sentiment lexicon, not only the emotional tendency of the words in the emotional lexicon will change with the change of the time node, the construction of the sentiment lexicon will encounter a variety of new words, or some cold

old words whose frequency would improve greatly because of the emergence of new interpretations of these words.

6. Experimental results and evaluation

6.1. Experimental data

The experimental data of this paper is crawled from douban film. More than 520,000 film review text data are crawled for the experiment.
In this paper, we use 60 artificial tagged network buzzwords and 40 professional words for film critics. Words those appeared more than 20 times in the film review data are randomly selected into three groups as a test set. The composition of sentiment lexicon test set can be seen in Table 1.

Table 1 composition of the sentiment lexicon test set

Set	Positive	Negative	Total
Common Test Set 1 (CT_1)	2150	1875	4025
Common Test Set 2 (CT_2)	1465	1962	3427
Domain Test Set (DT)	40	40	80

6.2. Experimental methods and results

6.2.1. Comparison of seed words selection methods

In this paper, we use the word clustering method based on LDA topic model and K-Means ++ algorithm to select seed words.Table 2 gives the results of the experiment. The last column in the table gives the weighted average accuracy for each method.
1)Select both 40 positive and negative words with highest frequency of in the corpus respectively;2)Search for the terms in Google. Select both 40 positive and negative words with highest result respectively; 3)According to the word clustering method proposed in this paper, we select both 40 positive and negative words respectively;

Table 2 Contrast of Selection Methods of Seed Words

Method	CT_1	CT_2	DT	AP
M_1	0.786	0.792	0.788	0.789
M_2	0.766	0.800	0.863	0.783
M_3	0.776	0.809	0.900	0.792

6.2.2. Improved SO-PMI algorithm performance test

In this paper, in order to solve the problem of data sparse, the original SO-PMI algorithm is improved by using the LDA-based synonym group construction and co-distance.
The average accuracy of improved SO-PMI on the construction of emotional lexicon has a certain degree of improvement, which proves that the proposed method of SO-PMI can actually improve the performance of sentiment analysis by experiments.

6.2.3. Domain Sentiment Lexicon Expansions and Emotional Tendency Changes

In order to test the process of IDSL's ability of self-expansion and adjustment, we divide the film review data into two parts according to the date of film and comment, and uses the film and television comment data before 2013 as the initial evaluation data set, which is used to construct the initial lexicon, data after 2014 are divided into 3000 texts per batch for experiment.
In the experimental results, we can find some words such as "小鲜肉","情怀","直男","好人"and so on. As time goes on, the emotional tendencies of these words tend to be neutral and even derogatory from the previous commendation.

7. Conclusion

In this paper, LDA is used to construct the eigenvector of seed words, and then the seed words are selected by the cluster center selection of K-Means ++ algorithm. Not only that, the original SO-PMI algorithm is improved by introducing the co-distance and the synonym group mining method based on LDA word similarity calculation. The initial domain sentiment lexicon is constructed with the improved SO-PMI algorithm and corpus. And the DSL's self-learning method is designed for the dynamic data. The experiment is conducted based on the film and television evaluation data from www.douban.com.
The experimental results show that the proposed method has a great improvement on the DSL construction, and has good processing ability for dynamic data. In the future work, we will improve the existing IDSL construction method, in order to achieve a fuzzy function, and the output of specific emotional values, and import the concept of time window, making the sentiment lexicon keen to reflect the change of current words' emotion.

Reference

[1].Laohakiat S, Phimoltares S, Lursinsap C. A clustering algorithm for stream data with LDA-based unsupervised localized dimension reduction[J]. Information Sciences, 2017, 381: 104-123.
[2].Deng S, Sinha A P, Zhao H. Adapting sentiment lexicons to domain-specific social media texts[J]. Decision Support Systems, 2016.

Learning context-dependent word embeddings based on dependency parsing

Ke Yan[1]，Jie Chen[1]，Wenhao Zhu*[1]，Baogang Wei[2]
[1]Shanghai University, Shanghai, China
[2]Zhejiang University, Zhejiang, China
Correspondence: No. 99, at Shangda Road, Shanghai University, Shanghai, China;
Tel: 02166133878; E-mail: whzhu@shu.edu.cn

Abstract: Word embedding is the basic method of text representation and it has proven helpful in solving various text processing tasks. In natural language, the contextual information of a text has a crucial influence on the semantics of word representations. According to current research, most training models are based on shallow textual information and do not fully exploit deep relationships in sentences. Aiming at overcoming this problem, this paper proposes the dependency-based continuous bag of words (DCBOW) model. This model integrates the dependency relationship between words and sentences into the context in the form of weights, which increase the influence of specific context information on the prediction of target words. The experimental results show that relative to syntactic similarity, the proposed method highlights the semantic relations and improves the performance of word embeddings.
Keywords: word embedding, context-dependent, dependency

1 Introduction

With the rapid development of the Internet, the amount of data on the Internet has grown dramatically. Processing natural language by using computers has become an important means for acquiring and mining knowledge from Internet text data. As an important process in natural language processing (NLP), text representation can map words, sentences, etc. to the numerical space so that the computer can perform numerical calculations and processing. Due to the characteristics, text representation is widely used in various NLP tasks such as text classification [1], information retrieval [2], sentiment analysis [3], dialogue system [4] and machine translation [5].

With the rise of deep learning, word representations that are based on neural networks are widely used. Most existing models use context to predict target words. The skip-gram model uses the word embedding of a word in the context as the context representation. The CBOW model uses the average of contextual word embeddings to represent the context [6], and NNLM uses the word embeddings of the first n-1 words of the target word to represent the context [7]. However, most training models are based on shallow textual information and do not fully exploit deep relationships in sentences. For example, the PAS model [8] considers only the positional information or part-of-speech tagging. In addition, the method of learning word embeddings from dependency relations ignores the effect of other unrelated words [9].The fundamental essence of semantics is relationships. In a sentence, many types of relationships exist among words, such as the relative positions of words in the PAS model. However, it is a relatively weak semantic representation.

Aiming at the above problems, this paper proposes a method of learning context-dependent word embeddings based on dependency parsing. Based on the study of distributed word representations, dependency parsing is introduced to obtain the dependency relations between words. The experimental results show that the word

embedding generation model with dependency weights that is proposed in this paper further enhances the effect of the word representation method and greatly improves the performance of word embeddings.

2 Related Work

The strategy of using neural networks to train language models was first proposed by Xu Wei in 2000 [10]. Subsequently, Bengio et al. proposed a neural probabilistic language model [7]. While optimizing the language model, the word representation is also obtained through the mapping matrix. After training, each word in the text is represented as a continuous low-dimensional dense real number vector, which is called a word embedding. All these vectors form a word vector space, and each vector is a point in the space. In this space, we can determine their semantic or grammatical similarity based on the distance between word correspondence vectors.

However, most models consider only part of the contextual information when learning words and fail to introduce the other contextual information. The method that was proposed by C&W [11] uses the local context to train the word vector but ignores the influence of the full text on the word vector. Based on the work of C&W, EH. Huang proposed a method in 2012 that uses global information to optimize the word vector [12]. Next, in a more representative study, the CBOW and skip-gram models were proposed by Mikolov [6]. These two models can efficiently learn high-quality word vectors from large-scale unstructured text data.

Qiulin et al. [8] observed that the CBOW model assigns the same weight to each word in the window. Thus, even if the order of the words were reversed, the CBOW model still produced the same result. Therefore, Qiulin et al. proposed a PAS model that was based on the position information of the words to obtain a word-sensitive higher-quality word vector. Levy, Goldberg et al. [9] also proposed an improvement to the skip-gram model. Instead of using a linear word-bag context, their improved model used dependency-based contexts, and they found that different types of contexts produced markedly different embeddings.

3 Dependency-based Continuous Bag of Words (DCBOW)

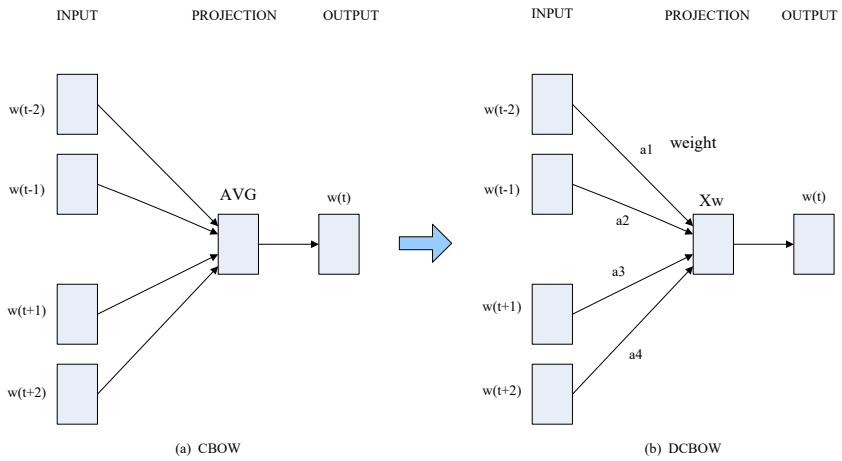


Figure 1 DCBOW model

For the CBOW model, as shown in Figure 1(a), the mapping process from the input layer to the projection layer is performed by averaging of the word vectors of all

the words in the context. The formula is as follows:

$$X_w = \frac{1}{2k}\sum_{i=1}^{2k} v(Context(w)_i) \in R^m \quad (1)$$

The modified model structure is shown in Figure 1(b). The difference is that the mapping from the input layer to the projection layer increases the influence of various types of contextual information on the prediction of target words, which is reflected in the form of weights. Then, the projection layer is obtained by weighted summation. The formula is as follows:

$$X_w = \sum_{i=1}^{2k} a_i \cdot v(Context(w)_i) \in R^m \quad (2)$$

where *m* represents the length of the word vector and a_i is the weight of the context word when it is projected.

The model predicts the probability of the current word in the context *c* of the current word *w*:

$$p(w|c) = \frac{\exp(e'(w)^T X_w)}{\sum_{w' \in V} \exp(e'(w')^T X_w)} \quad (3)$$

$$c = w_{t-k}, w_{t-(k-1)}, \dots, w_{t-1}, w_{t+1}, w_{t+2}, \dots, w_{t+k} \quad (4)$$

The optimization goal of the DCBOW model is to maximize the following log-likelihood function:

$$L = \sum_{(w,c)\epsilon C} logp(w|c) \quad (5)$$

where *C* represents the entire corpus and we uses the stochastic gradient ascent technique to obtain the optimal parameters.

By analysing the contextual information, we introduce dependency parsing to obtain the weights. When using the dependency parser to parse sentences, the valid pairs of dependent words can be extracted. The dependent pairs consist of the dominant word and the subordinate word. For example, the parsing result of the sentence “Australian scientist discovers star with telescope.” is shown in Table 1.

Table 1 Dependency pairs

Dependency	amod	nsubj	ROOT	dobj	case	nmod:with
Dominant	scientist-2	discovers-3	ROOT	discovers-3	telescope-6	discovers-3
Subordinate	Australian-1	scientist-2	discovers-3	star-4	with-5	telescope-6

For each sentence, according to the word pair and window size, the context word of each predicted word is tagged, the dependent word is marked with *d*, and each unrelated word is no mark required. We used the context that is obtained by above method as input to the DCBOW model. The word weight with the label *d* is set to *a*, and the weight of the ordinary word is set to *b*, where $i * a + j * b = 1$ (the sum of *i* and *j* is the window size). When the objective function *L* in formula (5) is maximized, the parameters *a*, *b*, and a_i in formula (2) are adjusted.

4 Experiments

We evaluate the effectiveness of the DCBOW model through comparison with NNLM, CBOW and Skip-gram models. We conduct experiments on text corpora that were created from Wikipedia documents. For model evaluation, we mainly use two evaluation methods: word similarity and word analogy tasks. The window size in this experiment is set to 10 and the dimension of vectors is set to 200.

1) Word Similarity

WordSimilarity-353[13] and Mtruk-287[14] are manually labelled datasets that are often employed to evaluate the quality of word vectors. Spearman correlation coefficient is used to measure the accuracy. The experimental results show that DCBOW model achieves the highest accuracies of 0.661 and 0.548, which shows that the dependency relationship reflects the deep relationship in the sentence and can better integrate the semantic information into the word vector.

Table 2 Spearman coefficients of the models on various test sets

Model	NNLM	CBOW	Skip-gram	DCBOW
WordSimilarity-353	0.572	0.604	0.625	**0.661**
Mtruk-287	0.456	0.512	0.531	**0.548**

2) Word Analogy

The Semantic-Syntactic test set contains 5 types of semantic relations and 13 types of syntactic relations. We evaluated the performance on all categories, which makes the deep exploration possible.

Table 3 Accuracies of the CBOW model and the DCBOW model on the Semantic-Syntactic test set

Group	Domain	CBOW	DCBOW
-	Semantic	**34.74% (3081/8869)**	**53.24% (4722/8869)**
1	Common capital city	68.18% (345/506)	91.90% (465/506)
2	All capital cities	35.52% (1607/4524)	56.15% (2540/4524)
3	Currency	4.04% (35/866)	8.08% (70/866)
4	City-in-state	31.82% (785/2467)	54.56% (1346/2467)
5	Man-woman	61.07% (309/506)	59.49% (301/506)
-	Syntactic	**45.79% (4888/10,675)**	**48.55% (5183/10,675)**
6	Adjective to adverb	15.32% (152/992)	13.10% (130/992)
7	Opposite	16.01% (130/812)	16.26% (132/812)
8	Comparative	75.15% (1001/1332)	73.05% (973/1332)
9	Superlative	29.23% (328/1122)	27.54% (309/1122)
10	Present participle	36.27% (383/1056)	41.48% (438/1056)
11	Nationality adjective	70.04% (1120/1599)	83.61% (1337/1599)
12	Past tense	41.73% (651/1560)	47.69% (744/1560)
13	Plural nouns	54.58% (727/1332)	58.63% (781/1332)
14	Plural verbs	45.51% (396/870)	38.97% (339/870)
-	Total	**40.77% (7969/19,544)**	**50.68% (9905/19,544)**

Table 4 Accuracy in the Semantic-Syntactic test set

Model	NNLM	CBOW	Skip-gram	DCBOW
Semantic	23.51%	34.74%	51.93%	**53.24%**
Syntactic	41.41%	45.79%	44.84%	**48.55%**

According to Table 3, in the semantic part (groups 1-5), the accuracy of the DCBOW model with the dependency weights is higher than that of the CBOW model; the total accuracy increases by 18.5% and 15.85%, respectively. In the syntactic section (groups 6-14), the accuracy of the DCBOW model increases by only 2.76% and 4.26%. According to Table 4, the total accuracy of the DCBOW model is higher than that of other models regardless of the semantic or syntactic

questions. This result shows that the word vectors that are obtained by the DCBOW model perform well and yield improved results in terms of semantics.

5 Conclusions

Based on the research on the distributed word representation method, this paper proposes a DCBOW model, which is based on the context's contribution to the prediction of target words. This model uses parsing to obtain contextual relationships to parameterize weights and uses these weights to optimize the construction of word representation. The experimental results show that the proposed method obtains improved performance by identifying the deep relationships between words in sentences; furthermore, the word embeddings that are learned from the DCBOW model can better capture the semantic relations.

References

[1]. Zhu L, Wang G, and Zou X. (2017) 'Improved information gain feature selection method for Chinese text classification based on word embedding'. International Conference on Software and Computer Applications. ACM, pp.72-76.

[2]. Zamani H and Croft W B.(2017) 'Relevance-based Word Embedding'. International ACM SIGIR Conference on Research and Development in Information Retrieval. pp.505-514.

[3]. Santos C N D and Gattit M.(2014) 'Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts'. International Conference on Computational Linguistics. pp.69-78.

[4]. Ryu S, Kim S, Choi J, et al.(2017) 'Neural sentence embedding using only in-domain sentences for out-of-domain sentence detection in dialog systems'. Pattern Recognition Letters, Vol. 88, No. C, pp.26-32.

[5]. Zhang J, Liu S, Li M, et al.(2012) 'Bilingually-constrained Phrase Embeddings for Machine Translation'. National Key Laboratory of Pattern Recognition, Vol. 59, No. 13, pp.111-121.

[6]. Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[J]. Computer Science, 2013, pp.1-12.

[7]. Bengio Y, Schwenk H, Senécal J S, et al.(2006) 'Neural Probabilistic Language Models'. Springer Berlin Heidelberg, pp.137-186.

[8]. Qiu L, Cao Y, Nie Z, et al.(2014) 'Learning word representation considering proximity and ambiguity'. Twenty-Eighth AfAAI Conference on Artificial Intelligence. pp.1572-1578.

[9]. Levy O and Goldberg Y. (2000) 'Dependency-Based Word Embeddings'. Meeting of the Association for Computational Linguistics. pp.302-308.

[10]. Xu W and Alex R. (2000) 'Can Artificial Neural Networks Learn Language Models?'. In Sixth International Conference on Spoken Language Processing. pp.202-205.

[11]. Collobert R, Weston J, Karlen M, et al.(2011) 'Natural Language Processing (Almost) from Scratch'. Journal of Machine Learning Research, Vol. 12, No. 1, pp.2493-2537.

[12]. Huang E H, Socher R, Manning C D, et al. (2012) 'Improving word representations via global context and multiple word prototypes'. Meeting of the Association for Computational Linguistics: Long Papers. Association for Computational Linguistics. pp.873-882.

[13]. Finkelstein R L. (2002) 'Placing search in context:the concept revisited'. Acm Transactions on Information Systems, Vol. 20, No. 1, pp.116-131.

[14]. Radinsky K, Agichtein E, Gabrilovich E, et al.(2011) 'A word at a time: computing word relatedness using temporal semantic analysis'. International Conference on World Wide Web.pp.337-346.

A CNN-based Temperature Prediction Approach for Grain Storage

Chen Caiyuan
School of Software and Microelectronics
Peking University
Beijing, China
thuchencaiyuan@126.com

Li Yiyu
School of Software and Microelectronics
Peking University
Beijing, China
yiyuli@pku.edu.cn

Mo Tong
School of Software and Microelectronics
Peking University
Beijing, China
motong@ss.pku.edu.cn

Weiping Li
School of Software and Microelectronics
Peking University
Beijing, China
wpli@ss.pku.edu.cn

***Abstract*—Temperature prediction has a pivotal role in the grain storage phase. Accurate prediction results can optimize the effect of ventilation decisions and reduce the losses of stored grain. Most existing studies have only focused on layer temperature predictions whose predict particle size is very large. In contrast, this paper attempts to use Convolutional Neural Network (CNN) to predict the point temperature of grain piles. The CNN-based approach uses multiple convolution kernels that share weights to capture the characteristics of grain temperature at different locations, which make full use of the temperature information around the target point. Experiments on real business data show that compared to other conventional algorithms, CNN has the best prediction effect on point temperature prediction problems.**

Keywords—Grain Storage, Temperature Prediction, Convolutional Neural Network, Point Prediction

I. INTRODUCTION

Grain storage crucial for people's livelihood. However, eight percent of the grain is lost during the storage each year according to the report of FAO (Food and Agriculture Organization of the United Nations,2013). It has been well accepted that the losses of grain are mainly caused by mildew, pests and respiration of grain itself. Moreover, the temperature of grain piles largely determines what kind of ventilation scheme should be used to solve these problems. Therefore, temperature prediction plays a critical role in the maintenance of grain storage. Thus, accurate predictions of grain temperature can be used to develop ventilation strategies that help eliminate mildew and killing pest.

Normally, as shown in Fig 1, the temperature sensors are deployed in a three-dimensional distribution way inside the grain piles. It is easy to find that the temperature of some points is significantly higher than the ambient temperature. If the high temperature points can be discovered in advance and be measured in a timely fashion, it can save much energy and reduce the losses of stored grain. However, most researches have only focused on layer temperature prediction without further on the point temperature. At the same time, there is a certain degree of thermal conductivity between grain particles and the air in the grain piles. If the temperature information around the target point can be used, the prediction result will be more accurate.

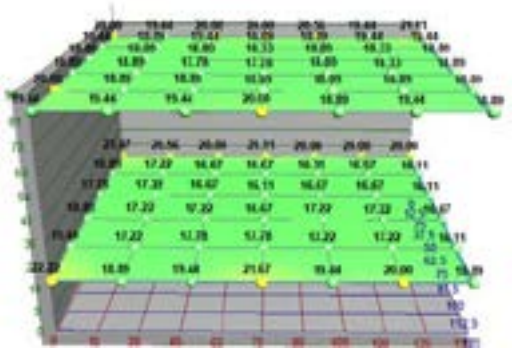


Fig. 1. Temperature distribution map of a grain warehouse (partial)

With the data analysis we found that there is a certain correlation between the temperature of the predicted point and the temperature of the surrounding points. However, this correlation has no significant relationship with location distribution. Naturally, we propose to using Convolutional Neural Network (CNN) to learn features from the correlation. CNN is a machine learning algorithm that has a high degree of invariance in spatial transformations such as translation and rotation.

In this paper, we propose a CNN based method for predicting the point temperature in grain storage. Through multiple convolution kernels sharing weights, the temperature features of different positions in the grain pile are captured. In this way the ambient temperature information is fully utilized to improve the prediction results. We conducted several sets of experiments aiming to find appropriate parameters and summarized the regular patterns of CNN in grain temperature prediction. We further compared our experimental results with other conventional algorithms on point temperature prediction.

The rest of this article is organized as follows. The second section introduces the state-of-the-art of grain temperature prediction and the related applications of CNN. In the third section, a grain temperature prediction method based on CNN is proposed. The fourth part gives the experimental results. Section V concludes the paper with some discussions.

II. RELATED WORK

Temperature prediction in the grain storage The existing literature on temperature prediction is extensive and many different methods have been used to study this issue. Lei Menglong et al. (2014) used a multivariate linear regression function to predict the temperature of the grain storehouse and achieved a good forecast effect. Support Vector Regression (SVR) can be used to deal with various regression prediction problems. Ni Fan (2017) constructed models of the grain ventilation process using SVR regression theory. The study suggested that the optimized prediction model has a better result especially when the sample size is limited. The Back Propagation (BP) artificial neural network has strong self-learning capabilities and the application field is very extensive. It is especially suitable for solving nonlinear problems and a number of studies have confirmed the effectiveness in the direction of temperature prediction. Surveys such as that conducted by Gao Song et al. (2015) and Shi Ruihua (2015) supposed a BP neural network forecasting model and used actual monitoring data to simulate on MATLAB platform. However, such studies remain narrow in focus only on layer temperature without further prediction of point temperature.

CNN related applications CNN are widely used in the field of image processing such as image classification and face recognition(Zhou Feiyan et al.2017). A seminal study in this area is the work that LeCun et al. (1998) designed CNN using the BP algorithm based on Fukushima's research work and the model is called LeNet-5. In a follow-up study, Krizhevsky et al. (2012) supposed a network structure used CNN for image recognition in the LLSRC-12 competition. This network structure is referred to as AlexNet and has achieved the best classification result. Compared to the traditional CNN, The AlexNet uses ReLU instead of saturation nonlinearity function, which reduces the computational complexity of the model. At the same time, the model becomes more robust and reduces over-fitting of fully connected layers through the dropout technique. In addition, CNN has also made achievements in audio retrieval. Abdel-Hamid et al. (2014) established the CNN model for speech recognition combined with Hidden Markov and conducted experiments on the standard TIMIT speech database. It is demonstrated that CNN model can improve speech recognition accuracy. In the same vein, various studies have assessed the efficacy of CNN in the field of short text clustering (Xu, J et al.2015), visual tracking (Gao Junyu et al.2016), image fusion (Li Hong et al.2016), etc.

III. CNN-BASED TEMPERATURE PREDICTION APPROACH

The temperature of target point will be affected by the temperature of surrounding points because there is heat exchange inside the grain pile. This characteristic makes point temperature prediction different from the layer temperature prediction. In order to study the effect of surrounding points temperatures on the prediction results of the target point temperature, we calculated the Pearson correlation coefficient (PCCs) r, between the target point temperature and the other points temperatures.

$$r = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2}} \quad (1)$$

The range of r is: $r \in [-1, 1]$. When $r = 0$, it means that there is no correlation between positive and negative values, and the greater the absolute value, the stronger the correlation.

Fig. 2. PCCs of temperature prediction from other points to the target point

For example, the correlation of all points temperature in a layer with the target point temperature is illustrated in Fig 2. What can be clearly seen is the positive correlation between the temperature of the target point and the temperatures of the other points in the layer. However, there is no obvious regularity between the intensity of correlation and the position of points. If linear regression is used to predict the temperature of the target point, the weights of results learned from each point will change with the positions of these points exchanged. That is, although the spatial relative positions of these points are the same, the original learning weights will be invalid after spatial changes. It will waste part of the spatial information and reduces the final prediction accuracy.

Therefore, we thus attempt to use CNN, which has invariance characteristics under translation and rotation, for point temperature prediction. Considering CNN can use the remaining points temperatures information as features to learn, it is clear that better prediction results would be achieved by sharing weighted convolution kernels to capture the patterns in different positions. The specific network structure is shown in Fig 3.

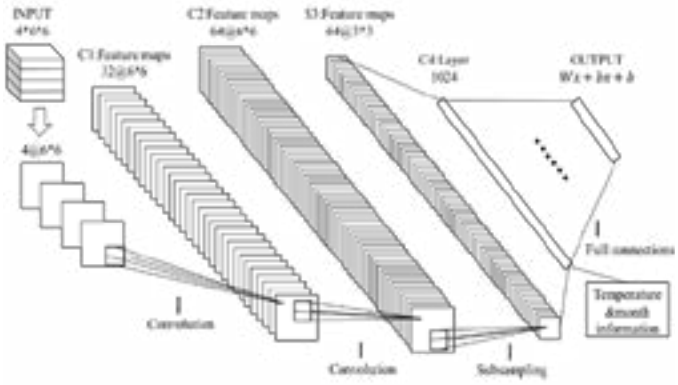


Fig. 3. CNN network structure for point temperature prediction

This section describes the architecture of the network structure in more detail. DoMapN is the size of each output feature map in each subsampling layer and oMapN is in convolution layer. Where $DWindow$ is the size of the subsampling kernel, $iMapN$ represents the size of each input feature map, $CWindow$ is the size of the convolution kernel, and $CInterval$ represents the sliding step of the convolution kernel above it.

$$\text{oMapN} = \left(\frac{iMapN - CWindow}{CInterval} + 1\right) \quad (2)$$

$$\text{DoMapN} = \left(\frac{oMapN}{DWindow}\right) \quad (3)$$

The input is the entire sensor temperature values of the entire granary and the size is 6*6*4, that means 4 layers from top to bottom and 6*6 points per layer. Each layer is similar to each feature map of pictures as LeNet-5, and we choose two convolutions layers and use SAME as the convolution method for keeping maps from shrinking in the convolutional part, considering that the input size 6*6 is too small. It is widely accepted that multiple small convolution kernel superimposed, which not only has the same connectivity, but also reduces the number of parameters and computational complexity compared to a single large convolution kernel.

The layer C1 is a convolutional layer with 32 feature maps. The size of the feature maps is 6*6 and each unit in each feature map is connected to a 2*2 neighborhood in the input. layer C2 is a convolutional layer with 64 feature maps. Similar to layer C1, the convolution kernels is of size are 2*2 and the size of the feature maps is 6*6 too. Layer S3 is a subsampling layer with 64 feature maps of size 3*3 and each unit in feature map is connected to a 2*2 neighborhood in the corresponding feature map in C2. The latter part is similar to an ordinary BP fully connected neural network, and the neurons number of the first layer is 1024. Different full connection depths and widths affect the final output. The output of S3 links with the weather and month features, considering that the influence of outside temperature and the grain custodians are more sensitive to warmer months. The result of the combination will be the input of the first fully connected layer. the output layer is defined as $Wx + bk + b$, where W is the parameter of input x, k corresponds to different months, and bk is the offset of the kth month.

the excitation function after each convolution layer and fully connected layer is ReLU (Rectified Linear Unit), and the Dropout operation is additionally added in the ReLU layer inspired by Krizhevsky et al.(2012). This structure can reduce the computational complexity of the model and avoid overfitting of the fully connected layers. Many experiments and studies have verified its effectiveness.

After the model structure being determined, there are still many adjustment tasks that need to be carried out for a specific problem. A series of experiments are needed to find the CNN parameters with better prediction results including the number of feature maps, the depth, and width of the fully connected layers. Finally, this paper compares and analyzes the effect of the conventional algorithm on the prediction results of point temperature.

IV. Experimental Results

In this section, we conduct a series of experiments in order to find the appropriate CNN parameters, and also compared the point temperature prediction results of different algorithms for the grain storage problem.

A. Data Preparation

1) Experimental data set

The experimental data set includes the grain temperature data set and the weather data set. The temperature data set comes from the grain temperature database of bungalow grain warehouse in the Hongze National Grain Reserve of Jiangsu Province, which contains a total of 1,319,568 temperature information from 2,552 points in 20 warehouses. These data were collected in the local database through the grain monitoring system from September 1, 2015 to September 1, 2017. While the weather data set contains 731 records in total. It has the same date as the temperature data set and comes from the Jiangsu Meteorological Bureau.

2) Data preprocessing

Due to the low level of information technology, the error rate of data is relatively high in the grain storage industry, which thus needs to be cleaned. After removing extreme data and filtering null values or abnormal data, the raw temperature data will be converted into numerical data so that the model could identify. Specific operations are as follows.

Data elimination: According to the experience of the grain storage industry, the temperature of grain higher than 45°C or lower than -20°C will be removed because it exceeds the normal range.

Data filtering: The null data, data with value 0, and anomaly data will be replaced with the average temperature of the layer. Abnormal points generally refer to points where the temperature exceeds the average temperature by 15°C in a week.

Time-based data processing: The time data recorded by the system will be accurate to the second, however only the data of year, month and day are useful in the prediction. The hour and minute parts will be discarded.

B. Experimental Setup

1) Input preprocessing

When predicting the point temperature of the grain, the input of each algorithm is a one-dimensional vector, including temperature values of all sensors in the entire warehouse, outside temperature, and month information. Since the collection cycle of point temperature information is irregular in actual business, it is impossible to use a continuous multi-day temperatures as an input. Therefore, we need to make a hypothesis of formula when predicting the temperature r of the next kth day. Under this assumption, we only use the temperature of that day as the grain temperature feature of the model, which is as shown in formula (4).

$$P(r|t_1, t_2 \cdots t_{k-1}, t_k) = P(r|t_k) \quad (4)$$

Temperature of air is an important factor influencing the change of grain temperature. The greater the difference between the air temperature and the grain temperature, the faster the grain

temperature changes. In addition, the average temperature of grain piles of each month is different. Ventilation often occurs during the month when the temperature of grain piles is high. The custodian is more sensitive to the high temperature months. Therefore, we take the average temperature of air in the next fifteen days as the input feature, make the month one-hot and map it to 1*12 vectors as an additional feature of the input.

2) Algorithm parameters

The conventional algorithms used for comparison include linear regression, SVR, and BP neural network.

Linear regression is a straightforward and easy-to-understand algorithm. Calling LinearRegression function in scikit-learn requires no adjustment of parameters. Three parameters are used in calling sklearn.svm.SVR function in scikit-learn: C, kernel and gamma. Here C is the penalty coefficient, means the tolerance of the error. An extremely large or small C will reduce the generalization ability. kernel function will be generally selected from the linear and RBF(Radial Basis Function). If the RBF function is chosen as the kernel, the function will come with the parameter gamma, which can affect the speed of training and prediction. Hence, the parameters used in the experiment were: C = 1.0, kernel = 'rbf', gamma = 'auto'. The BP neural network is constructed by TensorFlow and the hidden layer is a single layer. In the network, the neurons number is 1024 and the excitation function selects ReLU, as in the conventional CNN. The loss function is the mean square error (MSE). ReLU is usually used as the excitation function when MSE is used as the lost function.

All the experiments use K-fold cross-validation in this paper. The input data is divided into K piles by the number of granaries. One of the samples is used as the test set, and the remaining K-1 samples are used as the training set. Cross-validation is repeated K times. The K-time result is the final single estimate.

3) Evaluation method

There are two kinds of regression indicators: Mean Squared Error Root (RMSE) and R-Squared (R-Squared). The value of R^2 is from negative infinity to 1. The value 0 represents the average value, and 1 represents 100% correct rate. The definition is as shown in formula (7).

$$R^2 = 1 - \frac{MSE(\hat{y}_i, y_i)}{Var(y)}, \qquad MSE = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2 \qquad (7)$$

The custodians are more sensitive to the high average temperature of the grain pile, because abnormal warming of the grain pile may damage the safety of grain storage in the actual ventilation. Considering this factor, we have modified the weight coefficient of month on the basis of R^2 and changed the weight coefficient to be proportional to the average temperature of grain piles in the month. The experimental results show that the performance has become better after adding month weights, where the evaluating indicator in the experiments Score is defined as.

$$Score = 1 - \frac{u}{v} \qquad (8)$$

$$u = \sum_{m=1}^{12}\sum_{k=1}^{N_m}\frac{\omega_m\left(y_{true_{km}} - y_{pre_{km}}\right)^2}{N_m} \qquad (9)$$

$$v = \sum_{m=1}^{12}\sum_{k=1}^{N_m}\frac{\omega_m(y_{true_{km}} - y_{true_{mean}})^2}{N_m} \qquad (10)$$

C. Experimental Results

1) Exploring the effect of different CNN parameters on predicting results

The CNN network structure is shown in Fig 3. A series of comparative experiments have been conducted to obtain the appropriate parameters through changing the number of feature maps in each convolution layer, the depth and breadth of the fully connected layers. The experimental results are as follows.

The depth of the fully connected layer is 1, and the number of neurons is 512. The number of feature maps in two convolution layers will be changed during comparison experiments.

TABLE I. THE SCORES OF DIFFERENT FEATURE MAPS NUMBERS

Maps numbers	8→16	16→32	32→64
Scores	0.8476	0.8684	0.8796

The depth of the fully connected layer is 1, and the maps number of the convolution layers is 32→64. The breadth of fully connected layer will be changed during comparison experiments:

TABLE II. THE SCORES OF DIFFERENT NEURONS NUMBERS

Neurons numbers	1024	512	256
Scores	0.8806	0.8796	0.8735

The numbers of feature maps in convolution layers are 16→32, and the depth and breadth of fully connected layers are 512, 512 → 256, 512 → 256 → 128, respectively. The additional information includes the temperature of air and the month information:

TABLE III. THE SCORES OF DIFFERENT FULLY CONNECTED LAYERS

Fully connected layers	1	2	3
Include additional information	0.8684	0.8578	0.7498
No extra information	0.8433	0.8313	0.7334

From the above results, some conclusions can be drawn. For example, as the number of fully connected layer increases, the CNN prediction results of the point temperature gradually deteriorates. With the increase of the numbers of feature maps and the width of fully connected layer, the prediction results are improved, but the running speed will be slower at the same time. The additional information such as temperature of air and month information of grain can improve the experimental results. Because the grain temperature data has a smaller dimension and the entire three-dimensional space is not complicated, the data does not adapt to multiple non-linear transformations. The multilayer neural network will reduce the convergence speed of the network, and even cause over-fitting. A layer of fully connected layers is sufficient.

2) Comparing the effects of different algorithms on point temperature prediction

Comparison experiments based on real data sets were conducted to analyze the effects of various algorithms including linear regression (LR), SVR, BP neural network (BP-ANN) and CNN in the point temperature prediction. The scores in Table 4 are the sum of the average of the experimental granaries. The specific experimental results of each granary are shown in Fig 4.

TABLE IV. THE SCORES OF DIFFERENT ALGORITHMS IN THE POINT TEMPERATURE PREDICTION

Algorithm	CNN	BP-ANN	SVR	LR
Scores	0.8806	0.8719	0.8406	0.7971

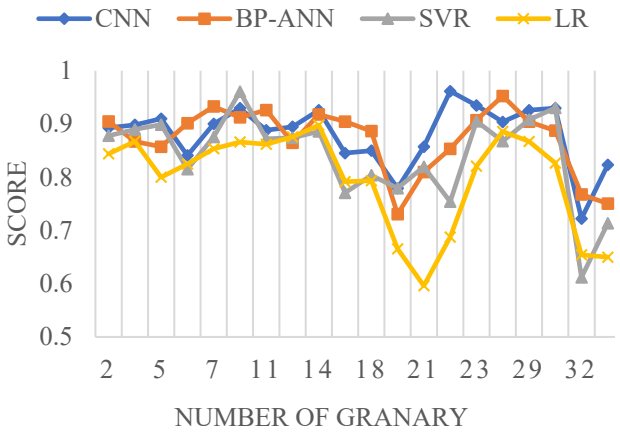


Fig. 4. CNN network structure for point temperature prediction

It can be seen from Table 4 and Fig 4 that, CNN works best on point temperature prediction. The linear regression cannot cope with complicated situations. It is natural the experiment results are quite different in each granary under the influence of different conditions, and the comprehensive result is poor because of the simple model. SVR is relatively better than linear regression, but the information of surrounding points cannot be used in point temperature prediction. That is why there is no dazzling performance on the SVR. In the fully connected neural network, the multiple neurons can capture more linear combinations and the nonlinear transformation of ReLU makes the result better. It is not surprising that the effect of BP-ANN is better than the previous algorithms. However, it does not have the ability of rotation invariance and that is why it is beaten by CNN.

CNN can use all the information of every point in the grain piles. At the same time, it can deal with more complicated situations because of the strong generalization ability under rotation invariance. That is why the performance is relatively stable and the variance is small. Compared with other algorithms, CNN has achieved the best experimental result. However, it should be noted that CNN does not have obvious advantages as previously conjectured. If the parameters of CNN are not particularly suitable, the prediction results may not be as effective as other algorithms. It is worth noting that the convolution and pooling operations do indeed help the result, but the effect of improvement is not significant.

On the one hand, CNN has too many parameters and it is difficult to find the best parameters. Apart from the parameters involved in the experiment, it also includes the selection of weight initialization method, regularization coefficient, and the momentum, etc. On the other hand, there is only 6*6 points in a layer and the difference in grain temperature data is not as obvious as the image data. The latitude of the image data is larger, and the difference of gray scale value between pixels is relatively large. However, the latitude of grain temperature data is too small and the temperature difference is not obvious, so that the feature is insignificant. Therefore, the role of convolution and pooling is not as outstanding as it was previously thought.

V. CONCLUSIONS

This paper focuses on the problem of point temperature prediction of grain piles, and uses the CNN with highly invariable spatial transformation such as translation and rotation to learn the temperature characteristics of the other temperature points. Through a series of adjustment experiments, we obtained a set of good parameters. The experimental results show that compared with other conventional algorithms, CNN is more suitable for the point temperature prediction of grain piles, and performs better under real data.

REFERENCES

[1] Abdel-Hamid, O., Mohamed, A. R., Jiang, H., Deng, L., Penn, G., & Yu, D. (2014). Convolutional neural networks for speech recognition. IEEE/ACM Transactions on audio, speech, and language processing, 22(10), 1533-1545.

[2] Bengio, Y. (2009). Learning deep architectures for AI. Foundations and trends in Machine Learning, 2(1), 1-127.

[3] Chen, C., Shu, X., Mo, T., Zang, C., Chen, Z., & Wang, Y. (2016, October). A Context Model for Mechanical Ventilation in Grain Storage. In Service Science (ICSS), 2016 9th International Conference on (pp. 88-93). IEEE.

[4] Gao Junyu, Yang Xiaoqi, Zhang Tianzhu, & Xu Changsheng. (2016). Robust visual tracking based on deep learning. Journal of Computers, 39(7), 1419-1434.

[5] Gao Song, & Song Hui. (2015). Prediction of temperature field in large warehouses based on BP neural network method. Cereals, Oils, Food Science & Technology, (2015 01), 94-97.

[6] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[7] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[8] Lei Menglong, & Tang Shaoxian. (2014). Multivariate linear eegression function and wireless sensors for prediction of granary temperature. hunan agricultural sciences: First Half Month, (2), 98-100.

[9] Li Hong, Liu Fang, Yang Shuyuan, & Zhang Kai. (2016). Remote sensing image fusion based on deep support value learning network. Chinese Journal of Computers, 39(8), 1583-1596.

[10] Ni Fan. (2017). Exploration of temperature field prediction methods in lateral ventilation process based on intelligent algorithm optimization SVM. Grain Storage, 46(1), 28-36.

[11] Shi Ruihua. (2015). The Application of BP Neural Network in the Forecast of Mean Warehouse Temperature. Software Guide, 14(8), 42-44.

[12] "Technology of grain storage", FAO annual report, 2013.

[13] Xu, J., Wang, P., Tian, G., Xu, B., Zhao, J., Wang, F., & Hao, H. (2015, June). Short Text Clustering via Convolutional Neural Networks. In VS@ HLT-NAACL (pp. 62-69).

[14] Zhou Feiyan, Jin Linpeng, & Dong Jun. (2017). Convolutional Neural Network Research. Journal of Computers, 40(6), 1229-1251.

Predicting Service Collaboration for Developers based on Data Variation Patterns

Jiaqiu Wang, Zhongjie Wang

Department of Computer Science and Technology,
Harbin Institute of Technology, Harbin, China
E-mail: wangjiaqiu@hit.edu.cn, rainy@hit.edu.cn

Abstract: Service collaboration allows the realization of more complicated business logic by using existing services. Nowadays, developers use a large number of services (e.g., Stack Overflow, Github, Blogger, etc.) to develop programs. Services are used continuously. Since most of the developer's data is distributed in these different service providers, these data are separated from each other although they are correlated. If we coordinate different services based on these semantic correlation data, we can provide developers with seamless and effective support. This is very significant because it greatly increases developers' productivity. However, due to the segregation of data, it is difficult to coordinate different services based on data correlation. To deal with this challenge, we propose a novel deep recurrent neural network (runs in a centralized service) to predict future services collaboration and their generated data. The network captures the semantic correlation between different data and discovers patterns of data variation by using multiple hidden layers, which are beneficial to services collaboration prediction. Extensive experiments are conducted on the real world data set. Experimental results show that our model significantly outperforms a few competitive baseline methods.

Keywords: Service Collaboration Prediction; Semantic Correlation Data; Data Variation; Deep Recurrent Neural Network.

1 Introduction

Service Oriented Computing (SOC) (Papazoglou, 2007) provides a new generation of computing paradigm to implement application integrations across organizational boundaries. Web services have been widely accepted as an important implementation of service-oriented architecture (Papazoglou, 2003). In SOC, loosely coupled and reusable services are used as basic building blocks (Alonso, 2004), whereby different users will be able to integrate their services to achieve the work goal or offer value-added services.

For instance, a real developer wants to build a deep learning model. Firstly, he learns how to program deep learning model by using the Youtube service (his preference). He watches and collects a video with title: "How to Make a Prediction - Intro to Deep Learning". Next, he uses the Github service to create a repository named "Prediction Model form Deep Learning" and commits his code to this repository. He runs into a program error (title: Deep Learning: Out of Memory error for data vector that's too wide) while training the model. If the developer can't solve this program error by himself, he may look for the solution

("decrease the dimension of vector") to this error by using Google Search (search service) and Stack Overflow (ask/answer question service). After that, he commits the updated code to same repository in Github. Last, the developer wants to record how to solve this program error, he uses WordPress (blog service) to write a blog with the title: "reduce the size of the data vector for out of memory in deep learning".

In this example, the program error data (stored in Github) is isolated from the answered data (stored in Stack Overflow). So Stack overflow couldn't automatically know what is the program error from Github service, and automatically give the right answer to the developer (or recommend related/relevant data), although the program error and the answer are semantically related. On the contrary, the developer manually performs Github and Stack Overflow services by himself to correct this program error. Similarly, the blog data is separated from updated program data (the developer corrects the error data). The WordPress service couldn't automatically write a related blog about how to modify this program error (data in Github).

In this paper, we propose a new deep recurrent neural network for predicting future service flow (services reuse) and their correlated data. Moreover, our model is running in a centralized service which collaborates various services. So it has the ability to build a service work-flow by using the API (Application Programming Interface) of each service. It helps developers efficiently complete their tasks in a smarter manner, and increase productivity by effectively completing more tasks (or more complicated tasks) during same period of time. Data semantic correlation and Data Variation are proposed in our model to help predict services reuse. They are two kinds of data features (Nazar and Radha, 2017).

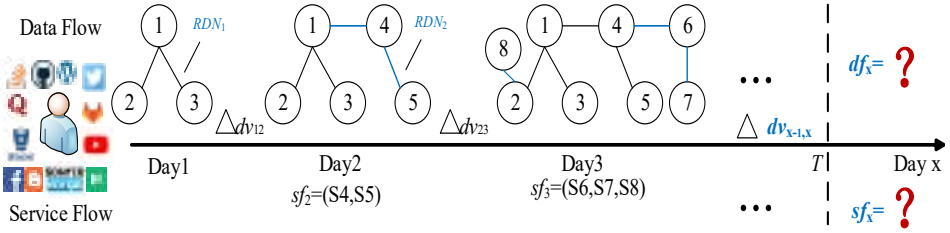
Our key contributions are summarized as follows:

- We propose a novel deep recurrent neural network for predicting future service flow and their correlated content/data jointly.
- we incorporate the semantic correlation between different data into typical deep recurrent neural networks model, which is more effective to find correlated data and recommend it to developers (it's beneficial to data prediction).
- we also incorporate the (temporal) Data Variation into our model, which discovers inherent patterns: what kind of Data Variation drives services to be performed (it's beneficial to services prediction).
- We show the efficacy of our model by comparing against state-of-the-art competitors for both service flow prediction and correlated data/content flow prediction using real data set.

2 Related Work

The Methodology of Prediction: Most related works focus on methodology for predicting service flow or recommending some relevant data/items for users. For some representative works: a deep recurrent neural network is used for real-time recommendation service in E-commerce (Wu, 2016), its network tracks how users browse websites by using multiple hidden layers. Improved collaborative filtering recommendation algorithm is employed (Hu and Peng, 2015) to solve the personal service recommendation problem, which mixes the temporal dynamics factors and personalized behavior. Mining association rules (Mohammad, 2016) is used for prediction, which takes advantage of a set of rules and

Figure 1 Modeling Data Variation and Service Flow.



a predictive class for these rules. A Dynamic Poisson Factorization model is presented (Charlin, 2001), where a state-space process is superposed on the user latent factor and item latent factor to capture the evolving preferences/attributes for the user and the item respectively.

However, these works focus on dynamics of services-usage behaviors itself, where they try to learn the gradually changing users' preference, instead of considering the semantic correlation between data and Data Variation's effect on the services-usage behaviors, which is the focus of this work.

3 Problem Definition and Formulation

Definition Services and Data Flow Prediction Problem (SDP): Given a developer's historical service flow SF_t^u and corresponding data DF_t^u, this problem can be formulated as a classification problem: when new data variation ($\Delta dv_{x-1,x}$) happens, the goal is to learn a function for **service flow prediction** (sf_x) and its corresponding **data flow prediction** (df_x) (Since these data are semantic correlation and form the RDN_x, it is an intermediate process, we can know df_x from RDN_x): $f : (\Delta dv_{x-1,x}) \mapsto (sf_x, df_x)$. For a graphical illustration of the SDP problem, see Fig. 1.

In Figure 1, each data network aggregates semantic related data in each day. Nodes represent different data, one edge represents the semantic relationship between data. $\Delta dv_{1,2} = \{(d_1, d_4), (d_4, d_5)\}$ denotes the Data Variation between RDN_1 (Day 1) and RDN_2 (Day 2). For each $\Delta dv_{t,t+1}$, we observe the sf_{t+1} and its df_{t+1}. If new $\Delta dv_{x-1,x}$ is generated, the goal is to predict service flow sf_x and its corresponding data flow df_x.

4 Proposed Method

4.1 Correlating Data-based Recurrent Neural Network

A particularly useful variant is the Long-Short Term Memory (LSTM) with specially designed units to avoid the problem of vanishing gradients. It proves to work well across many problems when the data is sequential, e.g., speech recognition, machine translation, or image captioning**?** . It also fits to solve our prediction problem (SDP). There are several variants of LSTM. In this paper, we use the following equations for our hidden units,

Input and Output Tensor. We incorporate the $dv_{i,i+1}$ embedding representation into the input tensor:

$$\mathbf{x_i} = [\mathbf{time\_steps}, (\mathbf{dv_{i,i+1}}, \mathbf{df_i}), \mathbf{dim\_input}] \quad (1)$$

time_step is the number of 2-tuple $(\mathbf{d_m}, \mathbf{d_n})$ in $\mathbf{dv_{i,i+1}}$. The output tensor is our prediction target (service flow embedding and data flow embedding):

$$\mathbf{o_i} = [\mathbf{time\_steps}, (\mathbf{sf_{i+1}}, \mathbf{df_{i+1}}), \mathbf{dim\_output}] \tag{2}$$

In one time step, the input of the neural network is still a vector. All vectors are input to the neural network one by one. You can regard it as a sequence of vectors. Similarly, you can also regard the output tensor as a sequence of vectors. We predict $\mathbf{sf_{i+1}}$ and $\mathbf{RDN_{i+1}}$ using this representation (ob_t),

$$ob_t = softmax(\mathbf{Vh_t}) \tag{3}$$

where V is the weight matrix, the output of the $softmax$ function can be used to represent a categorical distribution, that is a probability distribution **?**. The parameters in our model CDRNN and the embedding matrix $\mathbf{dv_{i,i+1}}$ are trained by minimizing the objective,

$$min. - \sum_x t_x log ob_x + (1 - t_x) log(1 - ob_x) \tag{4}$$

where t_x is the true output sequence of vectors. This objective is differentiable and the parameters can be learned using stochastic gradient descent.

5 Experiments

5.1 Evaluation

We split the data set into training (80%) and testing (remaining 20%) sets by selecting a time point T which is assumed to be the time point where we have the data set. This mimics the real-world situation where we observed developers' behaviors from the past up to the current time, and our model is asked to predict the future behaviors given all the available data at hand. For our data set, we set T to be 20170101, which results in 7,428,381 training samples and 1,357,621 testing samples respectively (e.g., one sample is $\mathbf{dv_{i,i+1}}$ with its prediction $\mathbf{sf_{i+1}}$ and $\mathbf{RDN_{i+1}}$).

We adopt $Precision_k$, $Recall_k$, $mean\ reciprocal\ rank$ (MRR) and $mean\ average\ rank$ (MAR) to assess the model performance for service flow prediction task. These metrics are also used to evaluate the quality of the top-K data ($\mathbf{df_{i+1}}$) recommendation. We compute each metric for each sample and take the average over the number of test samples.

Assume that there are $|N_{test}|$ samples in the test set. For a user u in a test sample s, $rank(u, v, s)$ be the rank that the model produces for data v and df_s^u denotes the set of user u's true data flow in that sample. Function $count(x)$ is to calculate the number of x. The four metrics of *df* prediction task are calculated as follows (metrics of sf prediction task is same to *df*),

5.2 Accuracy

To implement our deep networks, we use $TensorFlow$, a scalable deep learning library **?**. We train our model with 100 iterations (model can be convergence) over the training set with learning rate η 0.005 (this parameter analysis is described in Section 5.5). We use a

Table 1 Service flow sf prediction results for different models

	$Precision_{12}$	$Recall_{12}$	MRR	MAR
TRNN	0.103	0.301	0.129	398.23
TCF	0.095	0.284	0.084	520.68
DPF	0.087	0.297	0.108	401.21
PAR	0.055	0.263	0.114	451.82
Our Model	**0.161**	**0.374**	**0.152**	**380.79**

Table 2 Data flow df prediction results for different models

	$Precision_{12}$	$Recall_{12}$	MRR	MAR
TRNN	0.097	0.319	0.056	803.24
TCF	0.075	0.271	0.071	1000.79
DPF	0.062	0.184	0.088	943.08
PAR	0.083	0.292	0.096	1123.35
Our Model	**0.104**	**0.353**	**0.145**	**871.28**

deep network with three hidden layers and set the hidden unit number to 400. The size of the embedding and the dimension of the factorization-based competing methods are also set to 400. For all the compared models, we tune the parameters to give the best performance.

The results on service flow sf prediction tasks are presented in table 2. The results on df data flow prediction are presented in table 3, where our model outperforms competitors by sizable margins. The reason can be that, in our scenario, compared models predict slight short-time service flow (at one time) by using users' preferences/personalized behaviors. It is difficult for compared models to predict much longer time (at least one day) by using the Data Variation (at one time). Moreover, compared models predict some data flows, in which different d_i is not always semantic related. Instead, our model can do better than them by using the hidden state to represent the semantic relationship between different data.

6 Conclusion and Future Work

We present a novel deep recurrent neural network (runs in a centralized service) to predict future services collaboration (flow) and their generated data flow when Data Variation happens. It helps developers efficiently complete their tasks, and increase productivity by completing more (complicated) tasks, in a smarter manner. We incorporate the semantic correlation between different data and Data Variation into our model because the two data features are beneficial to our prediction problem SDP. Via the hidden states of our model, it can capture the data semantic correlation (is contributed to data flow prediction), and discover patterns: what kind of Data Variation drives what services to be performed (is contributed to service flow prediction). The overall experimental results show that our model outperforms the competitive baseline methods effectively.

References

Papazoglou, M. P.and Heuvel, W.-J. (2007) 'Service oriented architectures: approaches, technologies and research issues', *The VLDB Journal*, Vol. 16, No. 3, pp.389–415.

Papazoglou, M. P and Georgakopoulos, D. (2003) 'Service oriented computing', *Communications of the ACM*, Vol. 46, No. 10, pp.24–28.

BAlonso, G. and Casati, F. (2004) 'Web Services: Concepts, Architecture and Applications', *Springer Verlag*, Vol. 5, No. 4, pp.12–20.

Nazar, B. and Radha, N. (2017) 'An online approach for feature selection for classification in big data.' *Turkish Journal of Electrical Engineering and Computer Sciences*, Vol. 25, No. 1, pp.163–171.

Wu, S., Ren, W., Yu, C. (2016) 'Personal recommendation using deep recurrent neural networks in NetEase.' *International Conference on Data Engineering*, pp. 1218–1229.

Hu, Y., Peng, Q., Hu, X. (2015) 'Time aware and data sparsity tolerant web service recommendation based on improved collaborative filtering.', *Transactions on services computing.*, Vol. 8, No. 5, pp.782–794.

Mohammad, H., Johne, H. (2016) 'A Next Application Prediction Service Using the BaranC Framework.', *international conference on Computer Software and Applications Conference.*, pp.519–523.

Charlin, L., Ranganath, R., McInerney, J. (2001) 'Dynamic poisson factorization', *Proceedings of the 9th ACM Conference on Recommender Systems.*, pp.155–162.

Du, N., Dai, H., Trivedi, R., Upadhyay, U. (2016) 'Recurrent marked temporal point processes: Embedding event history to vector', *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*, pp.1555–1564.

Kazi, A., Kazi, R. (2012) 'Supporting the personal cloud', *Proceedings of the Asia Pacific Cloud Computing Congress.*, pp.25–30.

Montjoye, Y., Wang, S., Pentland, (2012) 'On the trusted use of large-scale personal data.', *IEEE Data Engineering Bulletin.*, Vol. 35, No. 4, pp.5–8.

Belhajjame, K., Paton, N., Embury, S. (2012) 'Incrementally improving dataspaces based on user feedback,' *Information Systems.*, Vol. 38, No. 5, pp.656–687.

Roberto, P., Franco, D. (2017) 'An SDN/NFV Platform for Personal Cloud Services.' *IEEE Transactions on Network and Service Management.*, Vol. 10, No. 1, pp.150–163.

Crossing Scientific Workflows Fragments Detection and Recommendation

Jinfeng Wen, Zhangbing Zhou*

School of Information Engineering,
China University of Geosciences (Beijing), Beijing 100083, China
E-mail: zhangbing.zhou@gmail.com
*Corresponding author

Abstract: Functionally-similar activity and fragment discover crossing workflows are considered challenging and have not been widely studied. This article proposes to detect and recommend fragments crossing scientific workflows by means of activity abstraction and an abstract network model upon abstract workflows. A dynamic approach for workflow fragments rank and recommendation is presented considering abstraction and instantiation according to their semantic and structural similarities.

Keywords: Scientific Workflow; Abstract Activity; Community Discovery Clustering; Abstract Network Model; Fragment; Ranking and Recommendation.

1 Introduction

Considering the fact that developing a scientific workflow from scratch is definitely a knowledge-intensive and error-prone mission, when a scientist would like to conduct a new scientific experiment, Bolte t al. (2016) think that developing this requirement-oriented scientific workflow through reusing or repurposing *best-practices* evidenced by current scientific workflows should be a cost-effective and risk-avoiding strategy. When this requirement can be completely, or at least partially, satisfied by a single scientific workflow archived in a repository, Zhou et al. (2018) present the mechanism that can discover the most appropriate candidates, which is appropriate by adopting workflow similarity computation techniques. Consequently, discovering appropriate fragments from multiple scientific workflows, and facilitating the reuse and repurposing of these assembled fragments, is a promising research challenge. Intuitively, developers may discover functionally-similar activities that have been developed by other scientific workflows from the repository, and try to reuse or repurpose these activities, when constructing new workflows, although some developers may prefer to create new, but functionally-similar, activities from scratch.

Besides, typical fragments can be retrieved from workflows to improve their sharing and reuse when possible, and Sarno et al. (2015) present an efficient method to promote the search and reuse of service process fragments with various granularities. Generally, current techniques aim to study the reuse and repurposing of complete workflows or their fragments. As far as we know, composing fragments, which are contained in various workflows, has not been explored extensively. These techniques seldomly consider the functional relevance of

activities in different workflows. In this setting, reusing and repurposing fragments crossing workflows, while considering the activity relevance in various workflows, is a promising topic to be further explored.

To address this challenge, this article proposes a crossing workflows fragments discovery mechanism, where the activity relevance in workflows is considered. Our contributions are summarized as follows: (i) Activities are clustered by adopting modularity-based community discovery clustering algorithm, and activities contained in a certain cluster are assumed to be functionality-relevant and thus can be represented by an *abstract* activity. (ii) A abstract network model is constructed upon abstract workflows. Candidate fragments are instantiated through replacing abstract activities by appropriate activities in a certain cluster, and these instantiated fragments are ranked according to their semantic and structural similarities with respect to the requirement specification.

2 Preliminaries

A scientific workflow swf is a tuple (tl, dsc, SWF_{sub}, ACT, LNK), where tl and dsc are the title and text description of swf, respectively; SWF_{sub} is a set of sub-workflows contained in swf; ACT is a set of activities in swf; LNK is a set of data links that connects sub-workflows in SWF_{sub} and activities in ACT. Note that a scientific workflow swf may have a set of sub-workflows SWF_{sub} and activities ACT, which are connected by links specifying the execution sequence between them. In this setting, a scientific workflow can be transformed into a virtual graph. A virtual graph vgr_{swf} transited from scientific workflow swf is a tuple (ACT, LNK, LNK_{vrg}). Particularly, LNK_{vrg} is a set of links in between ACT and inherit from LNK regarding to (sub-) workflow where those activities are contained, while ACT, and LNK are the same as those of swf. Note that semantic similarity computation for two activities leverages the activity similarity computation method is presented, which is briefly introduced as follows:

- The minimum cost and maximum flow algorithm and WordNet are adopted for computing the similarity between names of activities $sim_{aN}(act_1.nm_1, act_2.nm_2)$ and information of activities $sim_{aI}(act_1.info_1, act_2.info_2)$.

- After obtaining the similarity for the name and text information of two activities, the semantic similarity for these activities is calculated as follows through Formula 1 :

$$sim_{act}(act_1, act_2) = \alpha \times sim_{aN}(act_1.nm_1, act_2.nm_2) + \beta \times sim_{aI}(act_1.info_1, act_2.info_2) \quad (1)$$

  where factor $\alpha \in [0, 1]$ and $\beta \in [0, 1]$ reflect the importance of sim_{aN} and sim_{aI}, respectively. And the sum of α and β is 1. Generally, $sim_{act}(act_1, act_2)$ returns a value between 0 and 1.

It is worth noting that the additional text information $info$ is used to enhance the semantic similarity between activities. In this setting, it is thought that the text information is more meaningful and contributing than their names. Therefore, β is set to 0.7 accordingly in our experiments.

3 Functionally-similar Abstract Activity Formation

Leveraging the semantic similarity computation between activities, this section proposes to construct an activity network model for facilitating the formation of functionally-equivalent activities. An activity network model $ActN$ is a tuple (ACT, LNK, WGT), where ACT is a set of activities, LNK is a set of links that connect activities contained in ACT , and WGT is a set of weights defined upon LNK, which specifies distances between activities. An activity network model is construct, where the similarity reflects indirectly the weight upon links, after computing the semantic similarity between activities. As the similarity between activities is greater, the distance between activities is closer. Therefore, WGT, which is the distance between activities in the $ActN$, is calculated by (1 - similarity). The links between the distant activities are to be pruned for obtaining superior clustering. By considering the average path length and clustering coefficient of the network, the pruning threshold about similarity is determined accordingly.

We aim to group activities, which correspond to the activities in an activity network model (denoted $ActN$), into functionality-relevant clusters. Note that $ActN$ can be seen as a directed weighted graph. Importantly, a community discovery clustering algorithm based on modularity presented by Blondel et al. (2008), called $Louvain$, is applied to this $ActN$. Accordingly, high values of the modularity correspond to good divisions of a network into clusters, then one should be able to find such good divisions by searching through the possible candidates for ones with high modularity. After the $Louvain$ algorithm is applied, clusters are determined. However, activities that are far away from them don't belong to any clusters. Therefore, the procedure of these $outliers$ clustering is presented. In our technique, thresholds for each outlier is determined, such as the average similarity, which is obtain through calculating the similarity sum and count between this outlier and activities of the $ActN$. The average similarity of each cluster and maximum value involved in this threshold are calculated. Consequently, the cluster with the largest average similarity in is selected as the cluster which the this outlier belongs to.

4 Abstract Network Model Construction

An abstract network model $AbsN$ is a tuple (ACT_{abs}, LNK), where ACT_{abs} is a set of abstract activities and LNK is a set of links that connect abstract activities contained in ACT_{abs}. Workflows are rewritten into their abstract format through replacing activities by their abstract counterparts. An abstract network model is constructed, where the vertices correspond to abstract activities while the directed link reflect the invocation relation specified upon contiguous abstract activities. To summarize, any structures of abstract fragments, which are flexible to adapt to the user's various requests, exist in the $AbsN$. It is note worthy that the self-connection situation of abstract activities is considered in the $AbsN$. In real life, a user request is accomplished by one or more types of functional services. Correspondingly, at the abstract level, similar functional activities are in the same category, so the request will partial generate self-connected structures.

Leveraging the abstract network model constructed, we propose to generate keyword information for each abstract activity. This work will be prepared to further rank and recommend workflow fragments for facilitating their reuse and repurposing. A abstract activity semantics aas is a tuple of ($word_0$, . . . , $word_i$), where $word_i$ is the word at ith ($i \in \{1, n\}$) place chosen from the names of activities. A Rapid Automatic Keyword Extraction (*RAKE*) algorithm is adopted, which is an unsupervised, domain-independent, and language-independent

method for extracting keywords from individual documents. The processing of extraction keywords is (i) the candidate keywords, (ii) the keyword scores and (iii) the extracted keywords. After the RAKE algorithm is applied, the semantic information for each abstract activity in the $AbsN$ is generated. Furthermore, semantic discovery lay the foundation for further fragment mining.

5 Workflow Recommendation Based Abstraction and Instantiation

After an abstract network model with semantic information is constructed, appropriate fragments will be mined according to the user's requirement. We adopt a comprehensive strategy, which is a combination of the abstraction and instantiation in terms of structure and semantic matching, to rank and recommend workflow fragments for facilitating reuse and repurposing.

Assuming the user is an expert in this domain, whose requirement $User$ is considered to be professional and accurate. In order to search optimal or suboptimal fragments in the $AbsN$, we firstly transform the requirement into his abstract structures $User_{abs}$. Note that $User_{abs}$ is obtained through replacing activities by functionality-similar abstract activities. Then candidate abstract fragments composed by abstract activities can be found dynamically with respect to the requirement specification. Particularly, the above question, which refers to find optimal or suboptimal structures of fragments from $AbsN$, becomes graph or sub-graph matching in case of our fragment analysis and recognition. In this setting, the most representative $VF2$ algorithm, can solve the graph isomorphism problem, is applied in our experiment. Leveraging solutions discovered in the $AbsN$, we propose to select the semantically similar candidate fragments from them. With respect to the $AbsN$, whose abstract activities have aas abstracted from Section 4. The similarity value of each solution is calculated in the manner of the average semantic similarity between $User_{abs}$ and solutions. Note that the above work has ensured that selected candidate abstract fragments are structurally and semantically similar or superior to the abstract structure of user.

In order to identify candidate abstract fragments satisfied the scientists, we intend to discover the closest matching workflow fragments from the set of candidate abstract fragments. Therefore, instantiations are obtained, and in this phase they are diagnosed, ranked and recommended to scientists afterwards. Specifically, the main approach is as follows: (i) Keep structures of candidate abstract fragments unchanged. (ii) Abstract activities are replaced to the corresponding appropriate activities in this functionality-relevant cluster. (iii) Candidate instantiated fragment sets are ranked and recommended. For comprehensive consideration of requirement-related fragments, two evaluation parameters are adopted: the path length ratio and the semantic similarity.

When all similarity values have been calculated, suitable instantiations (or workflow fragments) should be ranked for recommendation. Consequently, the top $tp\%$ of instantiations are selected and the scientist will examine and determine which is the most appropriate with respect to his requirement.

6 Experimental Evaluation

We adopt four commonly used metrics, namely Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), to measure the recommendation performance of our technique. For the both metrics, smaller values indicate better performance.

$$MAE = \frac{1}{N}\sum_{i} |Simvgr_{true} - Simvgr_{rec_i}| \tag{2}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Simvgr_{true} - Simvgr_{rec_i})^2}{N}} \tag{3}$$

where the symbol "$Simvgr_{true}$" represents the similarity value (denoted 1) of the exact fragment with respect to a certain requirement. And the symbol "$Simvgr_{rec_i}$" indicates the similarity values about the recommended ith fragments according to our technique, where i is no more than the number of candidate fragments N.

To evaluate the quality of the proposed method in recommending appropriate fragments for a given requirement, we also use two other metrics, $precision$ and $recall$, to evaluate our proposed method on the testing experiments. The precision and recall are computed as follows:

$$precision = \frac{(|VGR_{ept} \cap VGR_{rec}|)}{|VGR_{rec}|} \tag{4}$$

$$recall = \frac{(|VGR_{ept} \cap VGR_{rec}|)}{|VGR_{ept}|} \tag{5}$$

where the symbol "$|VGR_{rec}|$" refers to the number of fragments in the set VGR_{rec} while the symbol "$|VGR_{ept}|$" specifies the number of elements in the set VGR_{ept}. And the symbol "$|VGR_{ept} \cap VGR_{rec}|$" refers to the number of fragments in the VGR_{ept} contained in the VGR_{rec}.

Meanwhile, the experimental results on the performance of our technique are presented in terms of (i) whether is the community discovery clustering algorithm can improve the recommendation performance and (ii) what is the impact of key parameters of the proposed model.

7 Conclusion

This paper proposes to discover appropriate workflow fragments crossing scientific workflows in the repository, and to identify and recommend them for reference when a existing or novel request of the scientist is to arrive. Specifically, an activity network model ($ActN$) is constructed considering the semantic similarity of activities. And activities in the $ActN$ are grouped into clusters by community discovery clustering based on modularity. Meanwhile, activities contained in a certain cluster are assumed to be functionality-relevant and thus can be represented by an *abstract* activity. An abstract network model upon abstract workflows is constructed. When a virtual graph is proposed to represent the requirement of a scientist, structurally and semantically similar workflow fragments are ranked and recommended considering abstraction and instantiation for reuse and repurposing.

References

Blondel, V. D., Guillaume, J. L., Lambiotte, R., and Lefebvre, E. (2008) 'Fast unfolding of communities in large networks', *Journal of statistical mechanics: theory and experiment*, Vol. 2008, No. 10, pp.P10008.

Bolt, A., de Leoni, M., van der Aalst, W. M. (2016). Scientific workflows for process mining: building blocks, scenarios, and implementation. International Journal on Software Tools for Technology Transfer, 18(6), 607-628.

Sarno, R., Pamungkas, E. W., Sunaryono, D. (2015, May). Workflow common fragments extraction based on WSDL similarity and graph dependency. In Intelligent Technology and Its Applications (ISITIA), 2015 International Seminar on (pp. 309-314). IEEE.

Zhou, Z., Cheng, Z., Zhang, L. J., Gaaloul, W., and Ning, K. (2018) 'Scientific workflow clustering and recommendation leveraging layer hierarchical analysis', *IEEE Transactions on Services Computing*, Vol. 11, No. 1, pp.169–183

A Multidimensional Service Template for Data Analysis in Highway Domain

Weilong Ding*

Data Engineering Institute, North China University of Technology, Beijing, China
Beijing Key Laboratory on Integration and Analysis of Large-scale Stream Data, Beijing, China
E-mail: dingweilong@ncut.edu.cn
*Corresponding author

Jie Zou

Research Institute of Highway Ministry of Transport, Beijing, China
TransChina(Beijing) Technology Co.,Ltd. Beijing, China
E-mail: 297632999@qq.com

Zhuofeng Zhao

Data Engineering Institute, North China University of Technology, Beijing, China
Beijing Key Laboratory on Integration and Analysis of Large-scale Stream Data, Beijing, China
E-mail: edzhao@ncut.edu.cn

Abstract: In highway domain, business analyses are multidimensional on massive data for traffic monitor and control. It is tedious to develop jobs from mutable requirements and is hardly to consider enough factors conveniently. In this paper, we propose a domain specific service template on massive toll data. Instantiated from the service template, abundant analysis jobs as services can be built in multiple dimensions flexibly. In a practical project, our method proves the feasibility and advantages by exhaustive experiments and case study.

Keywords: highway, service template, data analysis, spatio-temporal data

Biographical notes: Weilong Ding is an assistant professor at North China University of Technology, China. He earned the Ph.D. degree from Institute of Computing Technology, Chinese Academy of Sciences. He focuses on real-time data processing, and has published more than 40 articles in related fields.
Jie Zou is a senior engineer at Research Institute of Highway Ministry of Transport, China. He earned the Ph.D. degree from Shandong University of Science and Technology, China. He interests in industry informatization of smart highway, and has designed lots of major projects in highway domain.
Zhuofeng Zhao is a professor at North China University of Technology, Beijing, China. He earned the Ph.D. degree from Institute of Computing Technology, Chinese Academy of Sciences. He mainly concentrates on Service Computing and Internet of Things, and he been authorized ten more patents in practices.

Author

1 Introduction

In the inter-city traffics, the congestion of highway has become one of most serious problems worldwide, and the highway transportation systems are significant to govern urban situations (Holguín-Veras and Preziosi, 2011). For example, the toll data in Henan, one of the provinces in China, is about one million records a day and has been more than 0.2 billion in the year 2017. Such data is typical spatio-temporal, because the timestamp and location are kept in a record when a vehicle enter or exit a station. Like such toll data, various domain specific data is generated as stream from the sensors (de Assuncao et al., 2017) to capture the states in real-time, and would be accumulated as historical data to achieve the situations in macroscopes for the routine data analyses. The officials are eager to find effective ways for those data analyses. After necessary pre-processing (Ding and Cao, 2016), the domain specific data can be analyzed through statistic models on limited small samples (Ghesmoune et al., 2016) or through individual analysis jobs on massive or real-time data (Ding et al., 2017). However, it faces inherent limitations to the development when the business requirements are mutable in practice.

In this paper, on massive historical toll data, a service template is proposed in highway domain to achieve the multidimensional analyses. Instantiated from the service template, abundant analysis jobs as services can be built flexibly. It shows the advantages in a practical project through extensive experiments and case study.

2 Service template for multidimensional data analyses

Our work was initiated by *Highway Big Data Analysis System in Henan Province*, which has been applied by Henan Transport Department since October 2017. A record of toll data is typical spatio-temporal and contains 12 attributes as the *Table 1*.

Table 1. The structure of toll data

Attribute	Notation	Type
collector_id	toll collector identity	
vehicle_license	vehicle identity	
vehicle_type	vehicle type	Entity
card_id	vehicle passing card identity	
etc_id	vehicle ETC card identity	
etc_cpu_id	ETC card chip identity	
entry_time	vehicle entry timestamp	Time
exit_time	vehicle exit timestamp	
entry_station	identity of entry station	
entry_lane	lane number of entry station	Space
exit_station	identity of exit station	
exit_lane	lane number of exit station	

On such data, the analyses are jobs executed in a certain period (e.g., once a month) and on given input data (e.g., the last monthly data). After the data importation and cleaning, those jobs would output their results into traditional relational database or No-SQL distributed storage. We found that the jobs in highway domain can be described as the combination of six dimensions: time granularity, space granularity, vehicle type, vehicle direction, statistic object and ordering style as *Table 2*.

Table 2. The dimensions of the data analysis service in highway domain

Time	Space	Vehicle	Direction	Object	Ordering
5 minutes					
15 minutes					
1 hour	Station	Local		Traffic flow	Ascend
1 day	Section	Ecdemic	Entry	Traffic trend	Descend
1 week	Line	ETC	Exit	Proportion	No order
1 month	Region	MTC		Mileage	
1 year	Network	ALL			
Until now					

(1) The time dimension depicts the time granularity concerned in the job. For the values not larger than "1 hour", the jobs achieve the short-term statistics, otherwise are the long-term ones. The value "until now" implies all the historical time. (2) The space dimension shows the space granularity concerned in the job. For example, the value "station" means the jobs would achieve the results on each toll station; while "network" would be the results on the whole road network. (3) The vehicle dimension shows the vehicle type involved in the job. For example, some jobs only focus on the specific "ETC" vehicles. (4) The direction dimension shows the driving direction at a certain toll station. In practice, the analysis jobs only concern the exit direction, because the fee is charged when a vehicle exit at toll stations. (5) The object dimension explains the statistic objective of the job. Generally, "traffic flow" job counts the vehicle amount passing by; "proportion" job calculates the percentage of different element types (e.g., ETC versus MTC). (6) The order style dimension signifies whether the results are ordered or not.

With the dimensions of business analysis, the service template can be modeled as an abstract procedure of Hadoop MapReduce as *Figure 1*, in which all the dimensions have been considered conditionally. The dimensions time, space, vehicle, direction object, and order (respectively abbreviated to *t*, *s*, *v*, *d*, *ob*, *od* in the figure) are the parameters in the template and own the range as *Table 2*.

(a) In the map phase of *Table 2*, all the parameters would be assigned the actual values of concrete service during the initiation. When a record is read, the direction parameter *d* is used to extract exit time or entry time in the record. Then the key of intermediate result is built according to the intervals determined by the time parameter *t*. The space parameter *s* is considered for spatial granularity of statistics. The vehicle parameter *v* is required for the vehicle type to calculate. Except the mileage, other objectives are related with the traffic flow counting the vehicles amount at certain spatial range in a given temporal interval. Hence, according to these two object types, the intermediate results are outputted as two formats. (b) In the reduce phase, the intermediate results are gathered by groups according to the key. When a group is read, the value would be summarized as the output. For a job of TRAFFIC TREND, the predictive value should be supplemented by self-defined prediction algorithm, which should be implemented by the service instance. For a job of PROPORTION, the percentage of compared types should be supplemented then by self-defined algorithm, which should also be implemented by the service instance. At last, if the order parameter *od* is designated, the results achieved above should be rearranged by given order. For example, if *od*= DESCEND, the result would be in descending order.

Author

Accordingly, a concrete analysis job as a service instance can be built rapidly just by inheritance and extension from template.

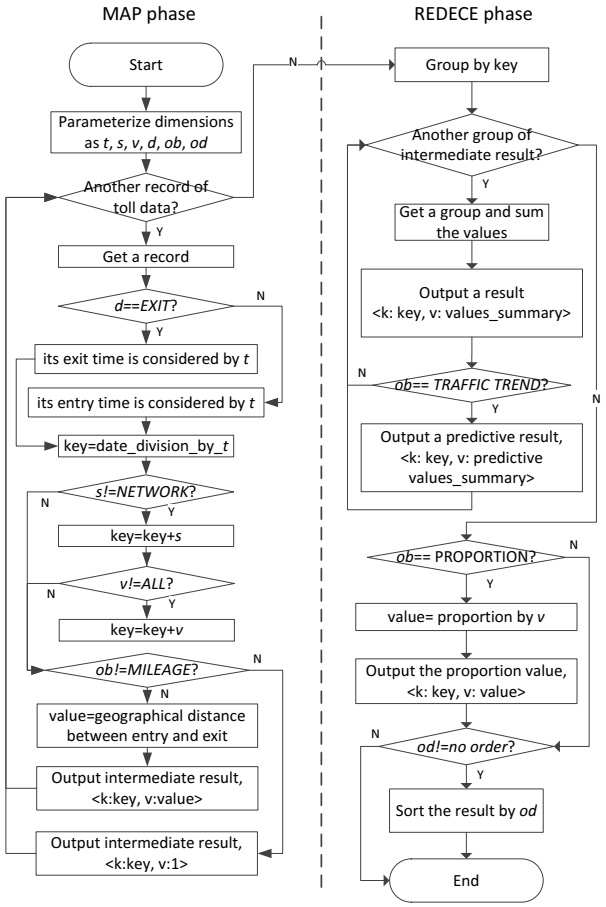


Figure 1. Processing procedure in service template

3 Evaluation

In our private Cloud, three virtual machines are used to deploy the analysis system, each of which owns 4 cores CPU, 22 GB RAM and 2 TB storage installing CentOS 6.6 x86_64 operating system. Here, Hadoop 2.6.0 cluster, HBase 1.6.0 and MySQL 5.1.0 are installed. We focus on the differences between our template (i.e., *with template*) and traditional ways (i.e., *no template*) during the analysis jobs' development. Both methods are compared in the following experiments.

Experiment. All the current 13 jobs are used, which include *hourly traffic flow of network*, *daily traffic flow of ETC vehicle*, *monthly vehicle type proportion*, and so on. All the jobs can be built by traditional and template way. We compare the effects from the development time and code quantity. Five students of similar programming experience as volunteers build those jobs, and each of them employs both methods. Their spent time and written code are noted in average. The result is showed in *Figure 2*.

The same trends are found in either the accumulative development time or the code quantity. When 10 more jobs are developed through tradition method, the required time grows to 10 folds and the codes are near to 15 folds. Through the service template, it only needs no more than 20 minutes with 100 lines code. The superior is from the fact that a job is the parameterized instantiation from service template. Only few additional codes have to be written for the tailored function, which can reduce the built time dramatically.

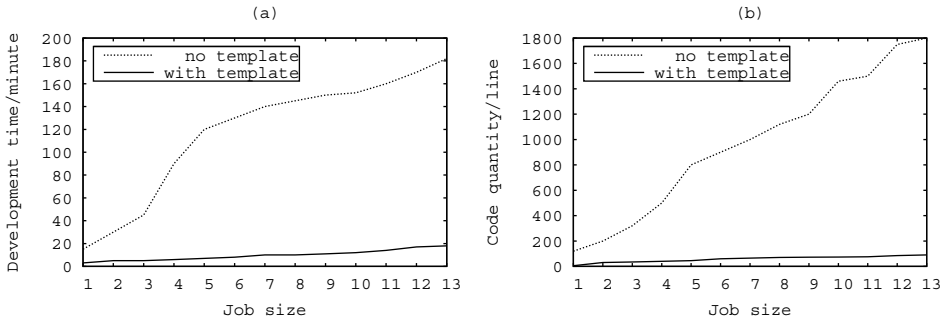


Figure 2. Build time and code quantity during job development

4 Conclusion and future work

A service template is proposed in highway domain on massive toll data to achieve the multidimensional analyses. Instantiated from the template, various analysis jobs as services can be developed flexibly. It shows the advantages of rapid development and high efficiency in a practical project through extensive evaluations.

Acknowledgments

This work was supported by the Youth Program of National Natural Science Foundation of China (No. 61702014), and the Ministry of Transportation, Institute of Highway Science Key Projects (No. 2015-9024).

References

de Assuncao, M.D., Veith, A.D.S. and Buyya, R. (2017) 'Distributed Data Stream Processing and Edge Computing: A Survey on Resource Elasticity and Future Directions', *Journal of Network and Computer Applications*.

Ding, W. and Cao, Y. (2016) 'A Data Cleaning Method on Massive Spatio-Temporal Data', in Wang, G., Han, Y. and Martínez Pérez, G. (Eds.): *Advances in Services Computing: 10th Asia-Pacific Services Computing Conference, APSCC 2016, Zhangjiajie, China, November 16-18, 2016, Proceedings*, Springer International Publishing, Cham, pp. 173-182.

Ding, W., Zhang, S. and Zhao, Z. (2017) 'A collaborative calculation on real-time stream in smart cities', *Simulation Modelling Practice and Theory*, Vol. 73, No. 4, pp. 72-82.

Ghesmoune, M., Lebbah, M. and Azzag, H. (2016) 'State-of-the-art on clustering data streams', *Big Data Analytics*, Vol. 1, No. 13.

Holguín-Veras, J. and Preziosi, M. (2011) 'Behavioral investigation on the factors that determine adoption of an electronic toll collection system: Passenger car users', *Transportation Research Part C: Emerging Technologies*, Vol. 19, No. 3, pp. 498-509.

Detect and Analyse the concurrent flaws of the BPEL process in a VPN-based approach

Puwen Cui, Ru Yang, Zhijun Ding

Tongji University, Shanghai, China.

Abstract: We focus on the modelling and verification the BPEL process using the Variable Petri net (VPN). We introduce an example as the motivation and basic mapping rules. Based on rules, we propose the analysis methods to detect the concurrent flaws including the deadlock and sequence violation of the BPEL process.

Keywords: BPEL, Variable Petri Net, concurrent flaws

1 Introduction

Nowadays, plenty of companies expose their services via web service. But a single web service only provides limited functionality. Therefore, service composition is a common requirement. BPEL is proved to be the most mature technology and widely used.

Though many efforts have been devoted to BPEL verification [4] [5], it is still a challenge. Many researches have been made based on Petri net to verify the BPEL process. Tan et al. [1] uses the ordinary petri net and Stahl et al. [2] uses the coloured Petri net. some other researches [3] use other high-level Petri nets.

Our previous work has introduced a new Petri Net named Variable Petri net abbreviated as VPN [6]. Based on the dynamic features of the VPN, we propose a VPN-based model and some analysis methods for the BPEL process to detect concurrent flaws.

The rest of paper is organized as follows: Section 2 introduces a BPEL process example. Section 3 describes the transformation rules from BPEL process to VPN. Section 4 presents the analysing methods and evaluation. Section 5 concludes our study and discusses our future work.

2 Overview and Scenario

Fig. 1. The travel process

3 BPEL is an XML-based formal specification programming language used for automated business processes. We introduce a BPEL process (travel process) which is composed of the hotel reservation service and flight booking service. The process receives the user information firstly and execute two services concurrently. The BPEL process graph is shown in the Fig. 1.

A business process should satisfy the deadlock-free before execution. Meanwhile, the sequence of service invocation shouldn't violate the designer's intention. We propose a class of new modeling and analysis methods using the VPN. VPN is the further abstraction of ordinary Petri Net. It realizes the dynamicity and uncertainty of the interactions by making full use of variable names. And thus it is a very appropriate model for systems with dynamic interactions.

4 Transforming BPEL to VPN

This section introduces the methods to transform a BPEL process to the VPN-based model. In the following parts, the actual parameter begins with the lower-case letter and the formal parameter begins with the upper-case letter unless noted.

4.1 Basic activity model

In this part, we pay attention to the receive activity's model while other activities can be deduced in the similar way.

A <**Receive**> activity specifies the partnerLink used to receive messages, the portType and operation that it expects the partner to invoke. We take the <**receiveUserInfo**> activity for example which is shown in Fig. 2.

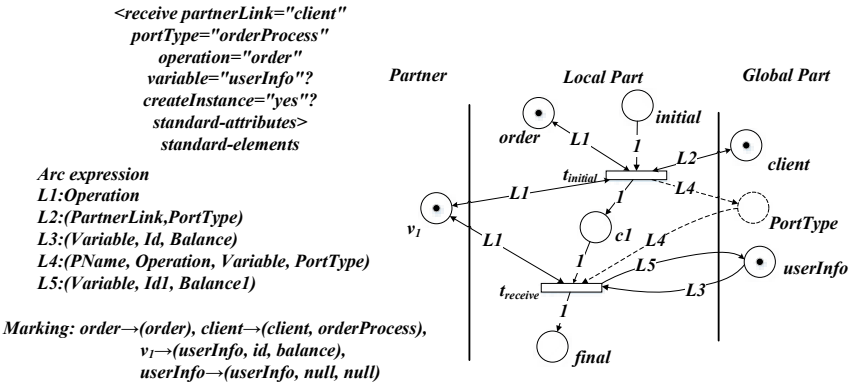


Fig. 2. The <**ReceiveUserInfo**> activity code&model

4.2 Structure activity model

In this part, we present one structure activity which is used in the travel process. For simplicity, we ignore the details of the composited activities and focus on the logical part between composited activities within one structure activity.

The <**flow**> activity provides concurrency and synchronization. To define the synchronization relationship in the <**flow**> activity, BPEL provides a mechanism called link. The <**flow**> activity with link structure is shown in Fig. 3.

We construct the model for the travel process which is shown in Fig. 4. The connection between activities is realized by merging the place *final* of the former activity and *initial* of the latter activity.

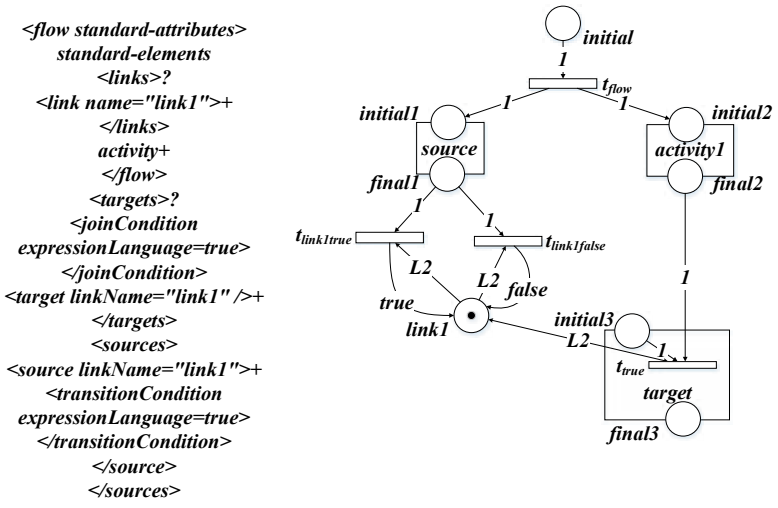


Fig. 3. the <**flow**> activity code&model

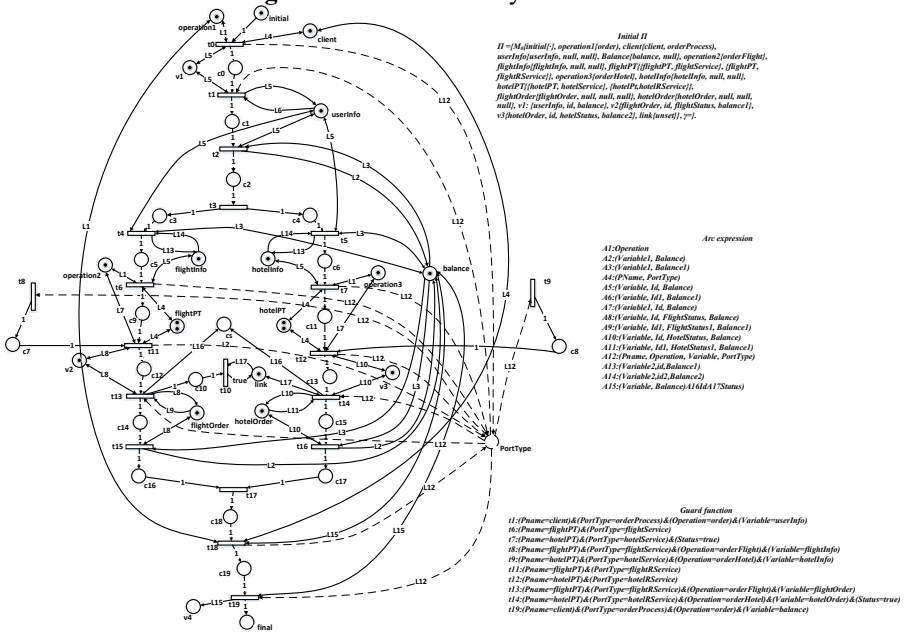


Fig. 4. the VPN-based model of the travel process

5 Analysis Method and evaluation

In this section, based on the VPN-based model for BPEL process, we propose the analysis methods to detect concurrent flaws including deadlock and sequence violation.

5.1 Deadlock detection

In the BEPL process, a deadlock may occur if two or more activities belonging to one flow structure are waiting for the completion of another. Based on the configuration tree of the VPN, the deadlock in the VPN-based model is defined as follows:

Definition 1: Let N be a BPEL model based on VPN, the M_0 be the initial marking of the net, $R(M)$ be the reachability marking set. If there exists a marking M satisfying

$$\forall t \in T,\ \neg M[t > \cap M(final) = \phi,$$

a deadlock occurs at the marking M of the process. Alg. 1 provides the algorithm of the deadlock detection.

Algorithm1: Deadlock Detection

Input: the configuration tree *CT* of a VPN-based model *N*
Output: a set of deadlocks: *SDL*
1*: SDL= { };*
2: **for** each node *m*∈ *CT* **do**
3: **If** (*m.child()*==null) **then**
4: **If** (*m.M(final)== null*) **then**
5: *SDL.add(m);*
6: **end if**
7: **end if**
8: **end for**

By traversing all the nodes in the configuration tree, if there exists a node whose place *final* has no token and the node is dead, then it indicates that a deadlock occurs under the marking of the node.

Then we analyse the travel process based on the Algorithm 1. There exists one deadlock in the process. The link structure defines that the <**ReceiveFlight**> activity happens before the <**invokeHotel**> activity. Meanwhile, the <**ReceiveFlight**> activity must execute after the correlation set is initialized by the <**invokeHotel**> activity. The conflict of the synchronization relationship leads to the occurrence of a deadlock.

TABLE Ⅳ deadlock detection based on three methods

Deadlock	**The number of deadlocks**		
	Our method	***Verbeek* [4]**	***Tan*[1]**
number	1	0	0

5.2 Sequence Violation Detection

The function of the travel process is to arrange the flight and hotel service for travellers. The designer should give all the possible sequence invocations in advance. And then the algorithm calculates all the permitted γ functions.

Alg. 2 provides the algorithm of the sequence violation detection.

Algorithm 2: Sequential Violation Detection

Input: the configuration tree *CT*, the VPN Model *G* including place set *P*, transition set *T*
Output: a set of the sequence violation nodes *SVN*
Define a γ function set FS, Define a mapping function *MF*
1: **for** (i=0; i<*PT*.length; i++)
2: **for** (j=0; j< *PT*[i].length(); j++)

```
3:          Define a mapping function MF
4:          for (k=0, k<j, k++)
5:          end for
6:          if (! FS.isContain(MF) then
7:              FS.add(MF);
8:          end if
9:      end for
10: end for
11: for each node n of the CT
12:     If ( n.γ(portType) ∉ FS ) then
13:         SVN.add(n);
14:     end if
15: end for
```

There exists a node whose γ function is PortType→(orderProcess, hotelService). The γ function of the PortType in this node which reflects that the process has invoked the hotel reservation service without invoking the flight service before. Therefore, it is a sequence violation in the process.

The above works aim at detecting the concurrent flaws of the BPEL process., we propose algorithms to detect it for each concurrent flaw and the travel process proves that our work deserves.

6 Conclusion

In this paper, one BPEL process can be transformed into a VPN-based model following the mapping rules. By describing the dynamicity and the data flow, sequence violation and deadlock can be detected more comprehensively. For large-scale process verification, we are developing an automated tool to accelerate the analysis speed.

Acknowledge

This work is partially supported by the Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji university, Shanghai, China, through the National Natural Science Foundation of China under Grant No.61672381. Corresponding author: Zhijun Ding.

References

[1] ***Tan W, Fan Y, Zhou M C, et al. A Petri Net-Based Method for Compatibility Analysis and Composition of Web Services in Business Process Execution Language[J]. IEEE Transactions on Automation Science & Engineering, 2009, 6(1):94-106.***

[2] ***S. Hinz, K. Schmidt, and C. Stahl. Transforming BPEL to Petri Nets. In W.M.P. van der Aalst, B. Benatallah, F. Casati, and F. Curbera, editors, Proceedings of the International Conference on Business Process Management (BPM2005), volume 3649 of Lecture Notes in Computer Science, pages 220–235, Nancy, France, September 2005. Springer-Verlag.***

[3] ***Y Xia, Y Liu, J Liu, et al. Modeling and Performance Evaluation of BPEL Processes: A Stochastic-Petri-Net-Based Approach[J]. IEEE Transactions on Systems Man & Cybernetics Part A Systems & Humans, 2012, 42(2):503-510.***

[4] ***Verbeek H M W, Aalst W M P V D. Analyzing BPEL processes using Petri nets[J]. Florida International University, 2012:59--78.***

[5] ***Melo P, Cunha P R D, Silva C F D, et al. Automatic run-time versioning for BPEL processes[J]. Service Oriented Computing & Applications, 2017:1-13.***

[6] ***Zhijun Ding, Ru Yang et al. Variable Petri Net. (submitted)***

State prediction and servitization of manufacturing processing equipment resources in smart cloud manufacturing

Shenghui Liu[1], Xin Hao[2], Shuli Zhang[1], Chao Ma*,[1]

[1]School of Software, Harbin University of Science and Technology, 150080, China
[2]School of Computer Science and Technology, Harbin University of Science and Technology, 150080, China
E-mail: hrbust.lsh@126.com
E-mail: charilyhao@foxmail.com
E-mail: zhangshuli0523@163.com
E-mail: machao8396@163.com
*Corresponding author

Abstract: For enabling the manufacturing processing equipment resources to intelligently perceive its own operating state in the machining workshop of manufacturing enterprise, the paper put forward an integrated prediction method based on combined BP neural network. In this method by combining the clustering ability of SOM neural network and the classification ability of BP neural network together, an integrated intelligent prediction model with the ability of both qualitative and quantitative analysis is defined and used to realize the accurate prediction of the operating state of manufacturing processing equipment resources. Next, the service encapsulation specification for the various algorithms and model in the integrated prediction method are given. These algorithms and models are encapsulated as a set of cloud services and then published to the smart cloud manufacturing service platform, so as to enable the virtualized manufacturing processing equipment resources in the smart manufacturing cloud pool combine their own processing ability and the intelligent perception ability of these cloud services together by carrying out service composition. Finally, experimental results demonstrate the effectiveness of the proposed method.

Keywords: Smart Cloud Manufacturing; Integrated Prediction Model, Combined BP Neural Network; Servitization; Intellectualization.

1 Introduction

Smart cloud manufacturing not only has the characteristics of networking and servitization, but also focuses on how to autonomously and intelligently carry out perception, interconnection, collaboration, learning, analysis, cognition, decision-making and operation for the people, machine, material, environment and information during the whole life cycle of the whole system (Li et al.,2016). While manufacturing resources are being encapsulated and published to the smart manufacturing cloud pool, they also need to have the intelligent ability such as the autonomous perception, learning, and analysis.

This paper mainly focuses on the manufacturing processing equipment resources in the machining workshop of manufacturing enterprise. Specifically, it is need to comprehensively use the internet and artificial intelligence method to monitor and predict the operating state of the manufacturing processing equipment resources, and then, it is also need to encapsulate the prediction model as a cloud service and published it, so as to support the networked, servitization and intellectualization of the resources.

Nowadays, the research on the state prediction of manufacturing processing equipment resources is in the development stage. Literature (Liu et al., 2010) extracted the feature vector by analyzing milling force signals, and took it as the input of BP neural

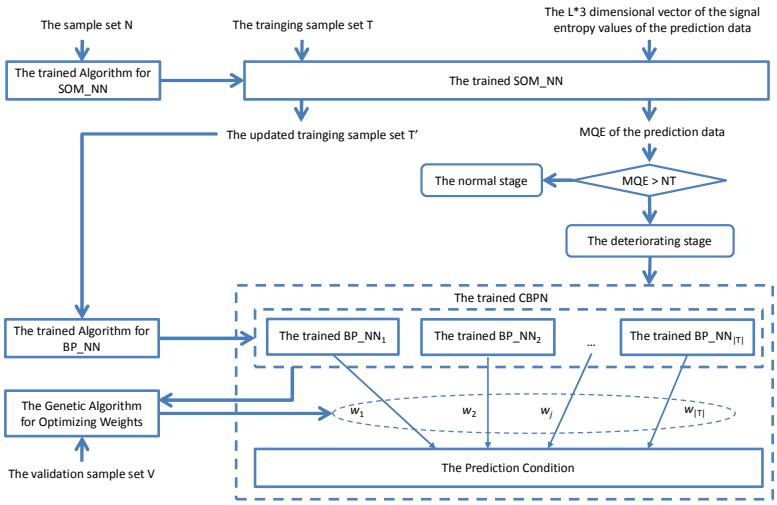
network, which is used for the detection of tool wear and the prediction of residual life. Literature (Hu et al., 2012) proposed a real-time residual life prediction method based on wavelet support vector machine and fuzzy C-means clustering from the perspective of studying similarity of deteriorating trajectories. It can be seen that although the current research has achieved preliminary results, in order to obtain a more applicable prediction model, it is necessary to further study.

Therefore, this paper comprehensively uses SOM neural network, BP neural network and genetic algorithm to propose an integrated prediction model based on combined BP neural network, which not only ensures the availability of the prediction model, but also makes the prediction model have a good adaptability. In addition, some researchers have also made valuable achievements in the servitization of manufacturing resources (Zhang et al.,2014). Our research group also obtained preliminary research results in this aspect. We encapsulated the soft manufacturing resources "the job shop scheduling algorithm" as a set of cloud services, and published them to the virtualized algorithm resource cloud pool, which improved the utilization of the algorithm resources, and also brought the algorithm resources the additional benefit (Liu et al.). Based on this, this paper further provides a servitization method for the soft manufacturing resource "the integrated prediction model", and then encapsulates and publishes it to the smart manufacturing cloud pool, so as to eventually realize the networking, servitization, and intellectualization of manufacturing processing equipment resources.

2 The integrated prediction model based on combined BP neural network

The framework of the CBPN-based integrated prediction model is shown in Figure 1. The input of the prediction model in the training process includes the three data sets N, T and V. The sample set N consists of the signal entropy values of the monitoring data at the normal stage. Both the training sample set T and the validation sample set V all consist of the signal entropy values of the monitoring data at the deteriorating stage. This paper calculates the wavelet time entropy, wavelet energy entropy and wavelet singular entropy for each indirect measurement indicator value to extract the characteristics of the monitoring data in the time domain and frequency domain. The L×3 (L is the number of selected measurement indicators) dimension feature vector is the input of the comprehensive prediction model. The more details are given in the subsequent sections.

Figure 1 The framework of the CBPN-based integrated prediction model



2.1 Calculation Method of MQE Based on SOM Neural Network

The entire life cycle of manufacturing processing equipment resources includes the normal state, the deteriorating state, and the failure state. Therefore, specifically, the goal of manufacturing processing equipment resources prediction is to firstly determine at

what stage the state of the manufacturing process equipment resources is at a certain time. For this reason, this paper draws on the concept of MQE proposed by Qiu H et al. (Qiu et al.,2003). MQE is a quantitative indicator to evaluate the degree of deviation from the normal state of prediction data.

In the specific implementation, the SOM neural network is first trained using the manufacturing processing equipment resources signal entropy data at the normal state. Each training, randomly selected part of the training data to form the input sample set S. Calculate the distance between this sample and the weight vectors of all SOM neural networks to obtain the Best Matching Unit for the current sample (Best Matching Unit (BMU)-The closest unit of weight vector to sample S). Finally, after the BMU is selected, its corresponding weight vector and topology neighbors will also be updated in time.

The minimum quantization error MQE is defined as:

$$m_{MQE} = \left\| \vec{x} - m_{BMU} \right\| \tag{1}$$

In formula (1), $\vec{x}$ is an input L*3-dimensional feature vector, m_{BMU} represents the weight vector corresponding to the BMU, m_{MQE} is the MQE value corresponding to the input vector.

2.2 Training combination BP neural network

The pseudo-code of the combined BP Neural Network CBPN training algorithm is shown below:

Input: Sample sets of signal entropy values in decline state of manufacturing processing resource-M= $T \llcorner V$, Where T is the training sample set and V is the validation sample set; Signal entropy value at the normal state of the manufacturing processing resource sample set-N

Output: Training completed combined BP neural network-CBPN

1. Training using sample set N to get SOM neural network, marked as F_{som}

2. For(j=1,j≤|T|, j++)

3. Perform time-based interpolation sampling operation on T_j ,

4. T_j={ $< \overrightarrow{x_j^1}, \overrightarrow{x_j^2}, \ldots, \overrightarrow{x_j^{NUM}} > | < ot_j^1, ot_j^2, \ldots, ot_j^{NUM} >$ **}**

5. For (i=1,i≤NUM, i++)

6. m_j^i **=F_{som}(** $\overrightarrow{x_j^i}$ **), Enter** $\overrightarrow{x_j^i}$ **to F_{som}, Get** $\overrightarrow{x_j^i}$ **corresponding MQE value** m_j^i

7. $\overrightarrow{x_j^i} \leftarrow m_j^i$ **,Replace** $\overrightarrow{x_j^i}$ **values in T_j with** m_j^i **,Get updated T'_j value**

8. T'_j={ $< m_j^1, m_j^2, \ldots, m_j^{NUM} > | < ot_j^1, ot_j^2, \ldots, ot_j^{NUM} >$ **}**

9. End For

10. At the sampling point i, the BP neural network BP_NN_j is trained using the updated training set T'_j

11. End For

12. Get |T| Training-completed Neural Network Sets BP_NN={BP_NN1, BP_NN2,…, BP_NN|T|}

13. For (i=1,i≤NUM,i++)

14. < W_1^i **,** W_2^i **,…,** $W_{|T|}^i$ **>=ForOptimizingWeights_GA(BP_NN,V);**

15. End For

16. Output training-completed combined neural network

17. $CBPN$={<BP_NN1,BP_NN2,…,BP_NN|T|>⊗< W_1^i **,** W_2^i **,…,** $W_{|T|}^i$ **>|i=1,2,…,NUM}**

In step 3, ∀i∈NUM, $\overrightarrow{x_j^i}$ is an L*3 dimensional feature vector. It is the signal entropy value vector obtained by preprocessing the monitoring data during the decline phase of

the manufacturing process resource by the wavelet entropy analysis method. ot_j^i is the actual running time of manufacturing processing resources at sampling point i.

2.3 Weight Optimization Method Based on Genetic Algorithm

In order to ensure that the CBPN output is generic and it is as close as possible to the actual results of different classes of validation samples. Set the actual remaining life of each verification sample and the variance statistics of the predicted output as the objective function of the genetic algorithm. After much iteration, the algorithm converges to a universal weight vector that fits all the validation samples. Therefore, the weighted sum of the final remaining life predicted by manufacturing resource at time point *i* is:

$$T_k^{NUM} = \sum_{j=1}^{|T|} W_j^i \times T_j^{NUM} \tag{2}$$

In formula (2), T_k^{NUM} is the failure time of the resources *k* at sampling time *i*; T_j^{NUM} is the operating time of resources *j* at sampling time *NUM*, i.e., the failure time.

3 Servitization of intelligent perception ability

The service encapsulation specification of algorithm and model resource is defined as a triple: Service=<BasicDescription, ParameterDescription, FunctionDescription>.

- BasicDescription=<ServiceID, ServiceName, ServiceDescription>

BasicDescription is the basic description of algorithm resource service and describes the basic information of the algorithm resource service. ServiceID is an ID of an algorithm resource service, ServiceName is a name of an algorithm resource service, and ServiceDescription is a detailed description of an algorithm resource.

- ParameterDescription=<ServiceInput, ServiceOutput>

ParameterDescription is the parameter description of the algorithm resource service, ServiceInput is the input parameter of the algorithm resource service, and ServiceOutput is the output parameter of the algorithm resource service.

- FunctionDescription=<FunctionName, FunctionProcess, CiteService>

FunctionDescription is the function description of the algorithm resource service, FunctionName is the function name description of the algorithm resource service, FunctionProcess describes the operation process of the algorithm resource service, and CiteService describes which algorithm resource service is cited.

In the integrated prediction model based on CBPN, the algorithms and models that need to be encapsulated include: 1) the input data preprocessing model; 2) the training algorithm of SOM neural network; 3) The calculation model of MQE based on the trained SOM_NN; 4) the genetic algorithm for optimizing weights; 5) the training algorithm of BP neural network; 6) The trained combination BP neural network Model.

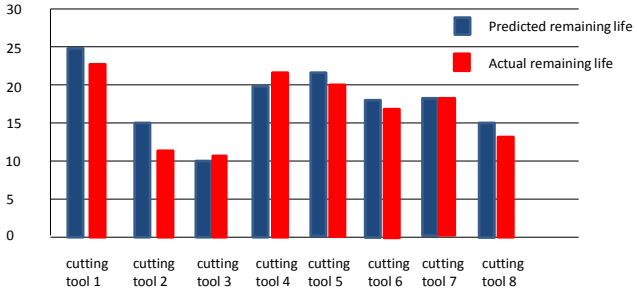
Under the support of WSDL, SOAP, UDDI and so on, the above algorithm and model resources can be encapsulated as set of Web service, and then can be registered and published to the manufacturing resource cloud pool, at last can be requested, accessed and shared.

4 Experiments and Analysis

This paper selects the manufacturing processing equipment resources CNC machine tools as a case to verify the above method. By observing the appearance of the cutting tools of CNC machine tools, we first selected the monitoring data of 20 similar tools in the normal state stage and preprocessed the data. Input the preprocessed sample data to SOM neural network and complete the training of SOM neural network. Then select 50 identical cutting tools for accelerated test to get 50 samples. Data preprocessing, time-based interpolation sampling operations and SOM neural network-based MQE calculations can be used to obtain sample datasets in the decline phase. From the

declining sample set, 30 groups were selected as training samples, 12 groups as validation samples, and 8 other groups as test samples. 30 training samples and 12 validation samples were selected as input of CBPN to complete the training and verification of CBPN. Finally, select 8 test samples for predictive testing.

Figure 2 Predict model validation results



5 Conclusions

This paper presents an integrated predicting method for manufacturing processing equipment resource based on combined BP neural network. The method uses the clustering results of SOM to achieve a qualitative prediction of the resource operating state. Based on this, the combined BP neural network model was trained according to the monitoring data of deteriorating state phase and used to achieve quantitative prediction. Further, a series of service encapsulation specifications for the integrated predicting model are given, so as to enable the virtualized manufacturing processing equipment resources combine their own processing ability and the intelligent perception ability of these cloud services together by carrying out service composition, and then realize the sharing of smart manufacturing processing equipment resources.

Acknowledgements

The research presented in this paper is supported by the National Natural Science Foundation of China (No.51375128), University Nursing Program for Young Scholars with Creative Talents in Heilongjiang Province (No.UNPYSCT-2016032).

References

Li Bohu, Chai Xudong, Zhang Lin. (2016) ‘Smart Cloud Manufacturing—A New Kind of Manufacturing Paradigm, Approachand Ecosystem of Deep Integration of the Internet and the Manufacturing Industry’, ZTE Technology Journal, Vol.22, No.5, pp.2-6.

Liu Rui, Wang Mei, Chen Yong. (2010) ‘A methodology for on_line toolwearmonotoring and predicting the remaining useful life of the cutting tool in facemilling’, Modern Manufacturing Engineering, Vol.06, pp. 102-105.

Hu Youtao, Hu Changhua, Kong Xiangyu.(2012) ‘Real-time Lifetime Prediction Method Based on Wavelet Support Vector Regression and Fuzzy c-means Clustering’. Acta Automatica Sinica, Vol. 38, No. 03, pp. 331-340.

Zhang Yingfeng, Zhang Geng, Yang Teng, et al .(2014) ‘Service encapsulation and virtualization access method for cloud manufacturing machine’. Computer Integrated Manufacturing Systems , Vol. 20, No.08, pp.2029-2037.

Liu Shenghui, Zhangxing, Zhangshuli, et al. Servitizationof job shop scheduling algorithm. Journal of Harbin University of Science and Technology, in press.

Qiu H, Lee J, Lin J, et al. (2003) ‘Robust performance degradation assessment methods for enhanced rolling element bearings prognostics’. Advanced Engineering Informatics, Vol.17, No.4, pp.127-140.

A Transition and Solution System for Uncertain Web Service Composition

Sen Niu[1], Yang Xiang[1], Shengye Pang[1], Hao Wu[1], and Ming Jiang[1]

Shanghai University, Shanghai, 200444, China
{sniu,yxiang,sypang,hwu,mjiang}@shu.edu.cn

Abstract. Uncertain Web service composition has become an important research issue in service computing. Although some research has been done on U-WSC, they have not integrated into a whole system, including the transition and solution process. In this paper, a Transition and Solution System for Uncertain Web service composition is proposed. The system can translate a U-WSC problem to a U-WSC planning problem, and then a planning algorithm is applied to solve the planning problem. We have conducted a case study based on the system. The results demonstrate the system can translate and solve the U-WSC problem effectively and efficiently.

Keywords: Uncertain · Web service composition · Transition · Solution.

1 Introduction

Web services are self-contained, self-describing and modular Web components that can be published, located and invoked on the Internet environment. They have well interoperability and reusability in real-world applications. As the development of Service-Oriented Architecture (SOA) paradigm in enterprise application integration, Web services have become more and more important in dynamic distributed applications. Its applications increase rapidly in many fields, such as electronic commerce, enterprise application integration and geographic information systems[10]. They are becoming an important part in modern services industry. However, in many cases there is no single service satisfying a given complex request.

Web service composition (WSC) is the task of combining a set of single Web services together to create a complex, value-added and cross-organizational business process[11]. It is applicable to those scenarios where individual Web service cannot satisfy the functionality requirement of a composition request. Many works have been done on WSC [1,8,3,4,2], where a WSC problem is modeled as a workflow business model or a classic AI planning problem that can be solved by an off-the-shelf automated planner to find a plan solution. However, most of these approaches suppose that Web services are stateless with certain execution effects. Thus, they seldom took into account the feature of inherent uncertainty of Web services.

Since Web services are published, deployed and invoked in dynamic Web environment, they have internal uncertain feature with non-deterministic effects on functionality properties. Therefore, uncertain Web service composition (U-WSC), composing existing Web services with the consideration of their uncertain features, has received many attentions and become a challenging research issue to be solved in service-oriented business applications. Some effects have been made in recent years. However, there is not existing a real transition and solution system for uncertain Web service composition.

To address the problem, a transition and solution system based on our preview works is proposed to solve the uncertain Web service composition problem. The system is consist of two sections. On one hand, the U-WSC problem is translated into a non-deterministic planning problem. On the other hand, the planning problem is solved by different algorithms. Finally, some experiment is conducted based on a case study and the results demonstrate that the system can solve the uncertain Web service composition effectively.

The rest of this paper is organized as follows. In Section 2, we describe a motivating example. Section 3 presents the architecture of the system. Experimental results on the running example are shown in Section 4. Section 5 reviews related work on Web service composition. Finally, Section 6 concludes the paper.

2 A Running Example

A running example from e-commerce application will be used throughout the paper. It consists of six Web services, including Retailer, Manufacturers M_1, M_2, and M_3, Assemble and Ship. Each service is responsible for a specific task with a collection of functionalities by its operations. Specifically, the **Retailer** sends a product request. The **Manufacturers** M_1,M_2 and M_3 have the same functionality, receiving a purchase request and checking its availability for the given request of the product purchase. The **Assemble** makes the assembling service based on the available status of the product and the **Ship** provides the shipping service of the product order.

The goal is to construct an integrated business process (i.e., a composed service) for product purchasing, assembling and delivering by combining a set of uncertain Web services. These services collaborate with each other to achieve a situation where the **Ship** can successfully provide the service with a requested product delivery for the **Retailer**. The abstract process of product ordering among six services is shown in Fig.1.

In Fig.1, the **Retailer** sends a given product request (*product name, numbers*) to these manufacturers. The **Manufacturers** M_1, M_2 and M_3 can provide the given product requested by the **Retailer**. Assume that M_1 and M_2 may not be satisfied for the request, while M_3 must be available. After receiving the given product request, the manufacturers check their availability (*CheckAvail*) for the request. If the checking status is available (*Available_yes*), the **Assemble** provides the assembling order service, and finally the **Ship** service delivers the given product to the **Retailer** (*product name, numbers, price, date*).

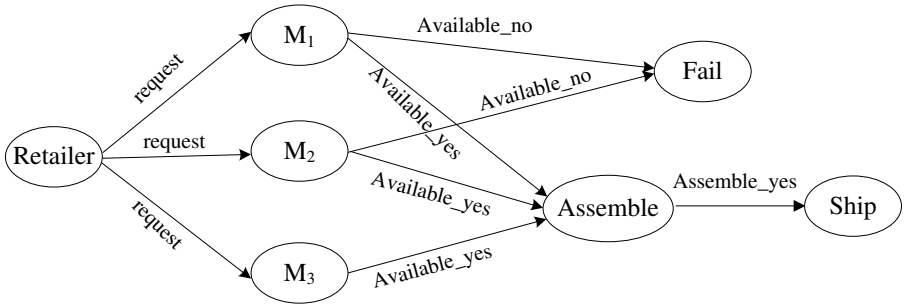


Fig. 1. Abstract process in Retailer and Manufacturer.

There are two uncertainties when invoking M_1 and M_2. When M_1 checks its availability, it may return two situations, including availability (*Available_yes*) or unavailability (*Available_no*). If and only if the returning status is *Available_yes*, the **Assemble** can be invoked and it provides the assembling order. The uncertainty of M_2 is similar with M_1.

With the consideration of uncertainty in Web services, we focus on how to effectively and efficiently compose these uncertain services and find all the possible solution paths as a whole satisfying the given request. In this running example, we aim at finding an uncertain solution to the request, such that we can always go through an execution path when those uncertain services (M_1 and M_2) output different execution effects. Thus, designing an effective and efficient U-WSC approach is desirable for service requesters.

3 Transition and Solution System

We develop a system for uncertain composition of Web services via non-deterministic planning. After the planning transition from a given U-WSC problem to a non-deterministic planning problem that is expressively represented as U-WSC planning problem, we take advantage of two algorithms for solving the problem to find all possible execution paths as a composition solution. The whole process is integrated into the framework as shown in Fig.2.

In Fig.2, the input of the framework is a U-WSC problem which involves an uncertain Web servicer repository and an uncertain composition request, while its output is a composition solution with all the possible execute path. Internally, the framework goes through two crucial steps: (1) Transition from a U-WSC problem to a U-WSC planning problem; (2) Solving the U-WSC planning problem by our proposed two uncertain planning algorithms.

More specifically, we first convert an uncertain Web service repository W in BPEL (Business Process Execution Language) and an uncertain composition request $\{R_{in}, R_{out}\}$ into a U-WSC planning domain D and a planning problem P in PPDDL(Probabilistic Planning Domain Description Language), respectively.

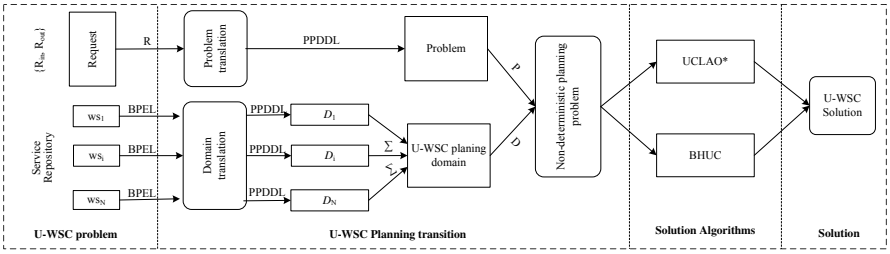


Fig. 2. The approach for uncertain composition of Web services via non-deterministic planning.

They are combined together as a U-WSC planning problem. Second, the U-WSC planning problem is fed into two uncertain planning algorithms (UCLAO* and BHUC) which find a composition solution with all the possible execution paths.

4 Experimental Evaluation

4.1 Experimental setup

In order to validate the effectiveness of our proposed U-WSC approaches and compare the efficiency with state-of-the-art non-deterministic planning algorithms, we developed a prototype system in Java. The empirical experiments were conducted on a PC with Intel Dual Core 2.8 GHZ processor and 3G RAM in Windows 7. A set of empirical experiments on the running example have been conducted on the prototype system platform.

4.2 Uncertain composition solution

We solve the U-WSC problem on the running example using the four algorithms, including AO*, LAO*, UCLAO* and BHUC, respectively. The experimental results are shown in the following.The composition solution result with cyclic actions is generated by LAO* algorithm and shown in Fig.3, while the composition solution generated by UCLAO* and BHUC is illustrated in Fig.4.

From the composition solution in Fig.3, it contains a loop when the uncertain actions $check\_m_1$ and $check\_m_2$ are unavailable. Consequently, the planner based on LAO* algorithm invokes the uncertain actions until they return the status of availability. However, in most cases, this cannot satisfy a U-WSC composition request, because we most possibly find another action with the same functionality and replace it the complete the task instead of waiting for its repetitive execution.

In the Fig.4, all the possible execution paths are generated by uncertain planner that integrates UCLAO* and BHUC algorithms into the process of finding

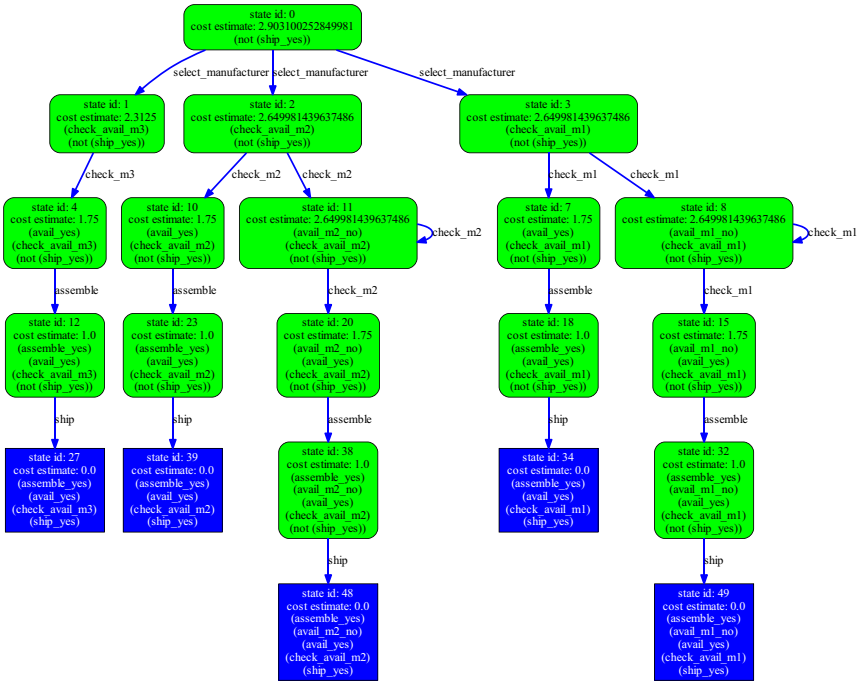


Fig. 3. The composition solution without cyclic actions generated by LAO* algorithm.

an uncertain composition solution. We observe that all the possible execution paths from the initial state to an output state can reach a goal state. There are nine execution paths involved in the U-WSC composition solution. They can handle all the situations where the execution output effects may not be available. Partial of them are as follows.

(1) If the uncertain action $check\_m_2$ outputs the status of availability, then the execution path of composite solution will be $< select\_manufacturer, check\_m_2, assemble, ship >$;

(2) Otherwise, when the uncertain action $check\_m_2$ outputs the status with unavailability, the execution path will be $< select\_manufacturer, check\_m_2, check\_m_3, assemble, ship >$;

(3) If uncertain action $check\_m_2$ still outputs the status with unavailability, while the uncertain action $check\_m_1$ has the output effect on availability, then the execution path becomes $< select\_manufacturer, check\_m_2, check\_m_1, assemble, ship >$;

(4) If uncertain actions $check\_m_2$ and $check\_m_1$ both outputs the unavailability status, then the execution path may be$< select_manufacturer, check_m_2, check_m_1, check_m_3, assemble, ship >$.

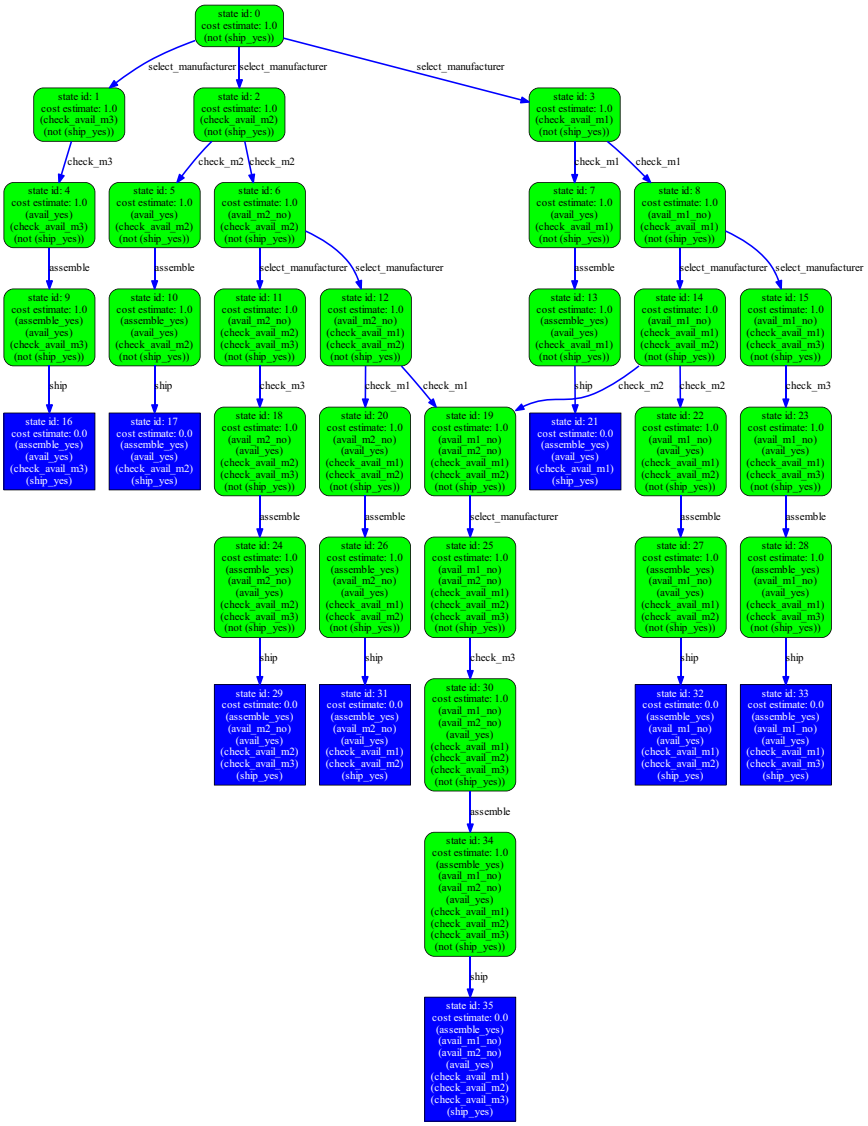


Fig. 4. The composition solution without any cyclic actions generated by UCLAO* and BHUC algorithms.

5 Related Work

We review the latest research results on dynamic composition of Web services using the techniques of automated planning in artificial intelligence.

Automated planning in AI has proved to be one of the most promising techniques for solving a WSC problem. Several works [10][5][6][9][7] have addressed different aspects of WSC, where a WSC problem is modeled as a classic planning problem. Web service planner (WSPR) [5][6] goes through two phases including forward search and regression search to find a feasible composition solution. During the search process, a heuristic function is used to choose a service with the biggest contribution to match a subgoal. A service composition algorithm is proposed by planning graph model in [9]. The process of finding a composition solution is the construction of a planning graph. It just selects a subset of services in order for new planning graph level, which possibly incurs redundant services in a feasible composition solution. Recently, we proposed an efficient approach for automatic composition of Web services using the state-of-art AI planners [10], where a WSC problem is regarded as a WSC planning problem that can be solved by any PPDL supported automated classic planners. A Hierarchical Task Network (HTN) planning system, SHOP2, is exploited for WSC in [7]. All of the available services in Web service repository are converted into a domain, and then the SHOP2 planner divides a composition task into many subtasks, until every atomic task can be invoked by a single service. Although many innovative approaches have been proposed to solve a composition problem, they cannot handle the situation of uncertainty where the execution of Web services possibly outputs multiple different effects.

Based on above investigations, we propose a novel framework for uncertain composition of Web services via non-deterministic planning techniques. After the mapping from a U-WSC problem to a U-WSC planning problem, we propose two algorithms to solve this U-WSC planning problem. The proposed algorithms can effectively and efficiently find a composition solution to an uncertain composition problem with good scalability.

6 Conclusion

This paper presents a system on solving service composition problem that first converts a U-WSC problem into a U-WSC planning problem via a set of transition rules. Then, two novel algorithms with heuristic state space search and optimization strategies, UCLAO* and BHUC, are used to solve the translated U-WSC planning problem. The two algorithms can find a composition solution with all the possible execution paths. Finally, we have conducted empirical experiments on the running example in an E-commerce application. The experimental results demonstrate that the proposed system is very effectively.

References

1. Aggarwal, R., Verma, K., Miller, J., Milnor, W.: Constraint driven web service composition in meteor-s. In: Proceedings of the IEEE International Conference on Services Computing (SCC). pp. 23–30 (2004)
2. Jiang, W., Zhang, C., Huang, Z., Chen, M., Hu, S., Liu, Z.: Qsynth: A tool for qos-aware automatic service composition. In: Proceedings of the IEEE International Conference on Web Services (ICWS). pp. 42–49 (2010)
3. Klusch, M., Gerber, A., Schmidt, M.: Semantic web service composition planning with owls-xplan. In: Proceedings of the AAAI Fall Symposium on Semantic Web and Agents. pp. 55–62 (2005)
4. Li, Y., Lin, C.: Qos-aware service composition for workfow-based data-intensive applications. In: Proceeding of the International Conference on Web Services (ICWS). pp. 452–459 (2011)
5. Oh, S.C., Lee, D., Kumara, S.R.T.: Web Service Planner (WSPR): An effective and scalable Web service composition algorithm. International Journal of Web Services Research (JWSR) **4**(1), 1–22 (2007)
6. Oh, S.C., Lee, D., Kumara, S.R.T.: Effective Web service composition in diverse and large-scale service networks. IEEE Transactions on Services Computing (TSC) **1**(1), 15–32 (2008)
7. Sirin, E., Parsia, B., Wu, D., et al.: HTN planning for Web service composition using SHOP2. Journal of Web Semantics (JWS) **1**(4), 377–396 (2004)
8. Zeng, L., Benatallah, B., Ngu, A.H., Dumas, M., Kalagnanam, J., Chang, H.: Qos-aware middleware for web services composition. IEEE Transactions on Software Engineering (TSE) **30**(5), 311–327 (2004)
9. Zheng, X.R., Yan, Y.H.: An efficient syntactic Web service composition algorithm based on the planning graph model. In: Proceedings of the IEEE International Conference on Web Services (ICWS). pp. 691–699 (2008)
10. Zou, G., Gan, Y., Chen, Y., Zhang, B.: Dynamic composition of web services using efficient planners in large-scale service repository. Knowledge-Based Systems (KBS) **62**, 98–112 (2014)
11. Zou, G., Lu, Q., Chen, Y., Huang, R., Xu, Y., Xiang, Y.: Qos-aware dynamic composition of web services using numerical temporal planning. IEEE Transactions on Services Computing (TSC) **7**(1), 18–31 (2014)

Index of Author

Y

Z