International Journal of Innovative Computing, Information and Control
Volume **19**, Number **5**, October **2023**
 ISSN 1349-4198
pp. 1543–1560

RELIABILITY PREDICTION METHOD FOR SERVICE-ORIENTED SYSTEMS BASED ON CRITICAL COMPONENTS IDENTIFICATION

Xiuguo Zhang[1], Yun Zhao[1,*], Zhiying Cao[1,*] and Shuo Jiang[2]

[1]School of Information Science and Technology
Dalian Maritime University
No. 1, Linghai Road, Dalian 116026, P. R. China
Zhangxg@dlmu.edu.cn; *Corresponding authors: { zhao_yun; czysophy }@dlmu.edu.cn

[2]Netshen Information Technology (Beijing) Co., Ltd.
No. 7, Kaikai Road, Haidian District, Beijing 100085, P. R. China
17824853961@163.com

Received October 2022; revised February 2023

Abstract. *Reliability prediction for Service-Oriented Systems (SOSs) can reduce the occurrence of emergencies and ensure the stable operation of systems. In this paper, a reliability prediction method for SOSs is proposed. Firstly, service dependency graph of SOSs is constructed based on service dependency relationship described by CA-CCML (Context-Aware Cooperative Composition Modeling Language) service composition model. Then, an IW-LeaderRank (Improved Weighted LeaderRank) algorithm is adopted to measure the importance of nodes in the service dependency graph and further to identify the critical components of SOSs. After that, a component reliability prediction model for SOSs based on Attention-LSTM network is proposed. Meanwhile, the critical services composition model graph is constructed using Depth-First-Search algorithm to describe composition structures among the identified critical components. Finally, the reliability prediction value of SOSs is calculated by using those of critical components and their composition structures. Experiments show that the proposed reliability prediction method for SOSs has obvious advantages in accuracy and efficiency.*
Keywords: Service-Oriented Systems, Reliability prediction, Improved Weighted LeaderRank algorithm, LSTM neural network, Attention mechanism

1. Introduction.

In the ever-changing information age, a single service can no longer meet the complex needs of users and Service-Oriented Systems (SOSs) based on Service-Oriented Architecture (SOA) have become the current trend of development [1]. For SOSs, component services are called through the network to achieve integration, and the collaboration of multiple component services is required to achieve a common goal. Therefore, compared with the traditional software system, SOSs need better communication between component services, and pay more attention to the quality and performance guarantee of the whole system. This requires that each component service constituting the system has good performance. Otherwise, the failure of a single component service is likely to cause cascade effect and make the whole service system unable to work normally. However, the highly dynamic change of network environment and the fluctuation of component service quality have brought great challenges to the quality assurance of component service. If we can effectively predict the failure of component services before the failure occurs, then some measures can be taken to adjust the service before the failure, so as to ensure the stable operation of SOSs. Therefore, early prediction of component service reliability plays an extremely important role in the stable operation of whole service system. However,

DOI: 10.24507/ijicic.19.05.1543

the failure of non critical components in the system is tolerable. The whole system will fail only when failed components exceed a certain threshold [2]. In addition, users prefer to use critical and reliable services [3]. Therefore, the reliability of critical components can better represent the reliability of the whole system.

Proactive Fault Management (PFM) [4] is a relatively effective way to improve the reliability of computer systems and ensure the continuous and stable operation of system. PFM can avoid faults by predicting the defects of the system and adjusting the system in advance. For the proactive fault management of service system, the key is to accurately predict the reliability of component system. Then on this basis, the whole service system is self optimized. Since the SOSs are deployed in a network environment, the changes in the system's working status, network throughput and other factors are dynamic and uncertain. These factors will have an impact on the operating state of SOSs. The traditional reliability prediction method only analyzes the variation law of a single factor of component reliability, which will lead to inaccurate prediction results. In addition, the traditional method first needs to predict the reliability of each component in SOSs. Then the reliability prediction value of SOSs is calculated through the composite structure between components. In this way, the reliability prediction of SOSs is faced with a large amount of computation, which easily leads to the slow response of prediction results. Due to the strong timeliness of reliability prediction, how to improve the accuracy of reliability prediction results of SOSs while ensuring the high efficiency of operation is a challenge in the current research.

Based on our previous research on critical components identification [5], this paper proposes a reliability prediction method based on critical components identification. The main research contents of this paper are as follows.

1) The critical components identification method of SOSs is proposed. Firstly, the service dependencies are obtained through the service composition model graph of SOSs described by CA-CCML. Thus, the service dependency graph of SOSs is constructed. Then, the IW-LeaderRank algorithm combined with service dependence intensity, service heat and service propagation ability of SOSs is used to rank the importance of components and identify critical components accurately.

2) An Attention-based Long Short-Term Memory (Attention-LSTM) network is proposed for reliability prediction of SOSs components. The historical data of reliability, availability, correctness and throughput time series of critical components are normalized and input into the Attention-LSTM neural network. Thus, the current reliability prediction value of critical components can be obtained. As one of the classical models of deep learning technology, LSTM neural network has great advantages in mining long-term dependence of reliability time series data. By introducing the attention mechanism, the model can better learn the reliability variation law of critical components and improve the accuracy of prediction.

3) The calculation method of service system reliability is proposed. Firstly, the Depth-First-Search algorithm is used to construct the critical services composition model graph of SOSs, which can describe the composition structure of critical components. Then, the reliability calculation method for SOSs is given. The reliability prediction value of SOSs can be obtained through the reliability prediction value of critical components. Finally, the validity of this algorithm is verified by designing and realizing travel SOSs.

The rest of this paper is organized as follows. Section 2 presents the related work on the reliability prediction of SOSs. Section 3 provides the reliability prediction framework for SOSs. Section 4 introduces the identification method for critical components of SOSs. Section 5 explains the reliability prediction method for SOSs in detail. Section 6 describes experimental results and compares the method in this paper with other methods. Finally, Section 7 concludes this paper.

2. **Related Work.** According to the granularity of prediction, the reliability prediction of SOSs can be divided into system level reliability prediction and component level reliability prediction. The reliability prediction results based on system level can be obtained by the combination of prediction results based on component level. However, most reliability prediction methods based on system level assume that the reliability of component services is known. How to obtain the specific component level reliability is often ignored, so the component level reliability prediction is very important. In this section, we introduce an overview of reliability prediction methods.

Zheng and Lyu [6,7] used collaborative filtering algorithm to predict the reliability of Web services. The algorithm defines reliability by collecting the proportion of errors when the client service is invoked in a certain time span. Then the idea of collaborative filtering is used to construct the "user-service" matrix, so as to predict the unknown "user-service" reliability value according to the similarity between users and the known "user-service" reliability value. This method can provide personalized reliability prediction value for users, and the reliability of unknown services can be predicted according to a limited amount of data. Silic et al. [8,9] proposed a CLUS model for predicting the reliability of atomic web services, which combines user, service and environment parameters, and uses k-means clustering algorithm to cluster historical data. Then, the similarity between services is used to find similar services for unknown services, and the reliability of this similar service is regarded as the reliability of unknown services to be predicted. This method considers factors affecting service reliability from different perspectives, and improves the accuracy of the prediction model. However, the above methods are all prediction of the average performance of Web services, rather than real-time and online prediction.

In order to predict the reliability in real time and online, Amin et al. [10] used the statistical time series ARIMA model to predict the reliability of SOSs. Ding et al. [11] also used the ARIMA model to predict the reliability of SOSs. The difference is that it uses an improved Spectrum Fault Location (SFL) technology to locate faulty components that lead to a decrease in system reliability. However, ARIMA requires time series data to be stable. In addition, ARIMA can only capture linear relationships, not nonlinear relationships. In recent years, some neural networks and deep learning models are outstanding in the reliability prediction of service system and have been widely used. Sobhana et al. [12] used artificial neural network model for reliability prediction. The results show that the artificial neural network model has a certain prediction accuracy for reliability prediction. Yadav and Balkishan [13] proposed a novel neural network-based deep learning reliability prediction model. The choice of the deep learning model has been determined because of its ability to automatically capture and learn the discriminative features from data, which results in an improved reliability prediction model. Wang et al. [14] proposed an online prediction method of reliability time series based on LSTM. In this method, the prediction period is divided into multiple intervals, and the reliability of these time segments is predicted. As users make more calls to services, the amount of data available to the model grows rapidly. In order to meet the real-time requirements, Wang et al. [15] integrated CNN on the basis of LSTM for preliminary data processing to improve the overall performance. Due to the dynamic characteristics of QoS parameters, it is difficult to guarantee high prediction accuracy at multiple time points in the near future based on the updated system operating state. Wang et al. [16] used a Probabilistic Graphical Model (PGM) to analyze historical and latest system parameters (including response time, throughput, and reliability). Then motifs-based Dynamic Bayesian Networks (m_DBNs) are used to describe the mode of historical parameter time series, so as to predict future time series. In order to further improve the prediction accuracy, Wang et al. [17] proposed a multi-step trajectory DBNs (multi_DBNs) model based on the cumulative effect of m_DBNs prediction time series. Based on the prediction results and other QoS constraints of important SoSs components, a proactive adaptation strategy is proposed. However, the above-mentioned reliability prediction method requires a huge cost and time. Although there have been some studies [18-20] on critical components, online reliability prediction is rarely considered. In addition, the reliability of system should be calculated considering different architectures, which helps to better understand the requirements of reliability [21].

Based on the above analysis, the current reliability prediction methods for SOSs have following shortcomings.

1) The reliability prediction results of SOSs are not accurate enough. The components of SOSs are affected by many factors such as network state and throughput. However, in many methods for reliability prediction of SOSs, the input of prediction model is usually only the single factor of component reliability. As a result, the reliability prediction results of SOSs are not ideal.

2) The reliability prediction process of SOSs takes a long time. Most of literature aims to improve the accuracy of reliability prediction of SOSs, ignoring the real-time characteristics of reliability prediction. If the time of reliability prediction of SOSs is shortened, it will be helpful to take effective measures in advance to ensure the stable operation of SOSs.

In order to improve the accuracy and efficiency of reliability prediction, a reliability prediction method of SOSs based on critical components identification is proposed in this paper. The reliability prediction value of SOSs is obtained by using critical components, which can greatly reduce the prediction time. Meanwhile, the reliability change law of critical components is obtained by integrating the time series data of service reliability, correctness, availability and throughput, which improves the accuracy of reliability prediction.

3. **Framework of Reliability Prediction for Service-Oriented Systems.** In order to make the reliability prediction result of SOSs more accurate and shorten the prediction time, this paper proposes a reliability prediction method for SOSs based on critical components identification. Critical components are identified by measuring the importance of components of SOSs and then the reliability prediction value of SOSs is obtained through the reliability of critical components. Figure 1 shows the framework of the method in this paper.

The reliability prediction method for SOSs based on critical components identification is proposed in this paper. The service dependency can be obtained from the service composition model graph of SOSs and then the service dependency graph of SOSs can be constructed. The IW-LeaderRank algorithm is used to measure the importance of nodes in the service dependency graph, so that the ranking sequence of component importance values can be obtained. Through the method for determining critical components of SOSs, the sequence of critical components of SOSs is obtained. Then, the time-series data of reliability, availability, correctness and throughput of critical components are used as the input of the Attention-LSTM neural network to obtain the reliability prediction value of critical components. After obtaining the sequence of critical components, the Depth-First-Search algorithm is used to construct the critical services composition model graph of SOSs to describe the composition structure among critical components. In the process of calculating the reliability of SOSs, the calculation formulas for different combination

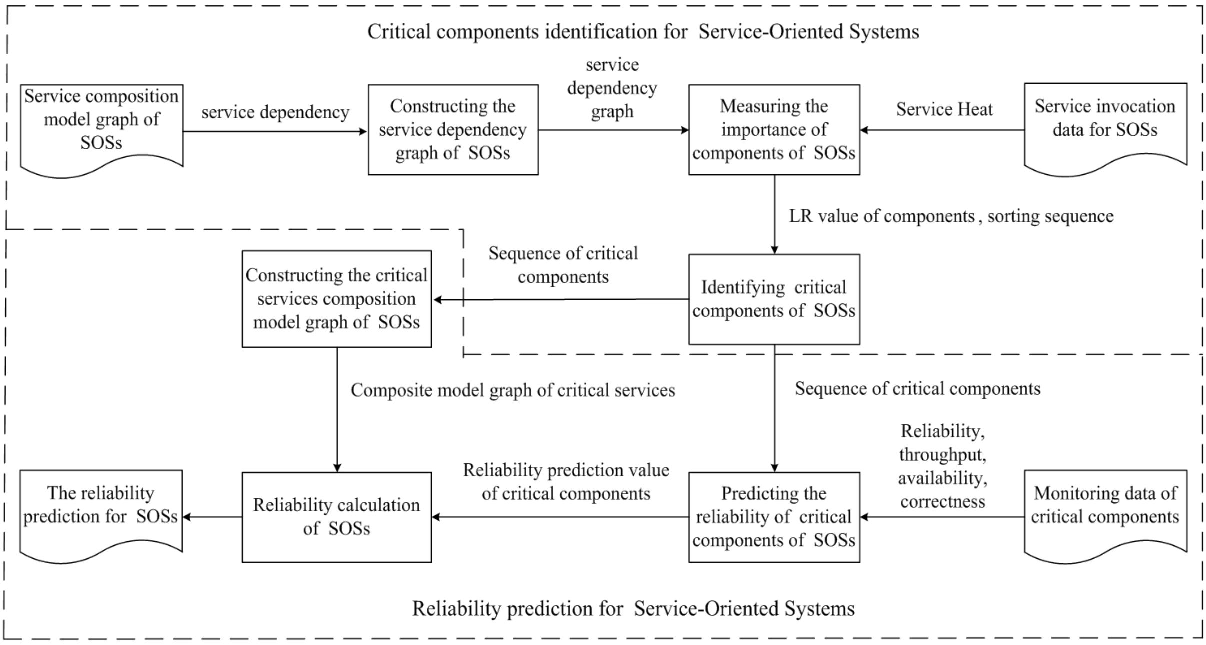


Figure 1. Framework of the reliability prediction for Service-Oriented Systems

Table 1. Main symbols table

Main symbols	Symbolic interpretation
LR_i	The importance value of components
H_i	The service heat value of components
γ	Service heat coefficient
BC_i	The betweenness centrality of components
μ	Betweenness centrality coefficient
R_i	The reliability value of components
C_i	The correctness value of components
A_i	The availability value of components
T_i	The throughput value of components
X_i	The input sequence of Attention-LSTM
h_i	The feature of LSTM hidden layer
a_i	The weight of attention mechanism of hidden layer feature
Y_i	The output sequence of Attention-LSTM, that is, the reliability prediction value of components
$R_{Sequence}$	The reliability value of sequence structure in SOSs
R_{Loop}	The reliability value of loop structure in SOSs
R_{Branch}	The reliability value of branch structure in SOSs
$R_{Parallel}$	The reliability value of parallel structure in SOSs

structures of critical components are given. The reliability prediction value of SOSs is obtained by using the reliability prediction value of critical components and the combination structure of critical services.

In order to make the method in this paper easier to understand, the main symbols in this paper are summarized, as shown in Table 1.

4. **Critical Components Identification for Service-Oriented Systems.** Critical components play a crucial role in the structure and function of the whole system [22], and the reliability of critical components will significantly affect the reliability of the whole system. Based on the service composition model graph of SOSs described by CA-CCML, this section uses the IW-LeaderRank algorithm to measure the importance of components, the sorting sequence of importance values of components in SOSs is obtained and then critical components are identified.

4.1. **Construction of service composition model graph based on CA-CCML.** CA-CCML is a Web service composition modeling language proposed by the research group of our laboratory [23]. It is based on CCML service composition language and adds the characteristics of context-aware personalization. Therefore, the CA-CCML proposed by our laboratory can be used to describe SOSs, which can not only express the static information of service, but also express the dynamic interaction information among services. At the same time, it adds the support of user class, service class and environment class, which is more in line with the characteristics of service. In addition, the visual Web services composition editor has been improved to increase context awareness while maintaining its original state. The comparison table of its visual graphic symbols and graphic models is as follows.

Table 2. Comparison table of visual graphic symbols and graphic models

Structure name	Graphical symbols	Graphical models	Structure name	Graphical symbols	Graphical models
Web Service	Service	ServiceModel	Unidirectional		ArrowConnectionModel
Input	in	InterfaceModel	Bidirectional		LineConnectionModel
Output	out	OutInterfaceModel	Branch		IfModel
Start		StartModel	Parallel		IfParalModel
End		EndModel	Variable	V	VarModel
Context sensor		Sensor	Chain		Reglink

4.2. **Service Dependency Graph of SOSs (SDGS).** Not only should we consider the dependency among services, but also the intensity of dependency. In our previous research [5], SDGS is used to describe the strength of dependencies among components. This method defines the service dependency intensity as the ratio of service interaction attributes. This paper uses this method to construct the SDGS of the flight reservation system as shown in Figure 2. In Figure 2, each ellipse represents a service. Among them, the login service s_{Login} and user information service $s_{UserInfo}$ are completely dependent. The flight ranking service $s_{FlightRanking}$, flight reservation service $s_{FlightReservation}$ and payment service s_{Pay} are also completely dependent. Therefore, the weight of these edges is 1. In addition, the flight query service $s_{FlightEnquiry}$ depends partly on the calendar service $s_{Calendar}$ and login service s_{Login}. The flight ranking service $s_{FlightRanking}$ depends partly on flight query service $s_{FlightEnquiry}$, user information service $s_{UserInfo}$, location service $s_{Location}$ and weather service $s_{Weather}$. The weight of the edge is determined according to the proportion of interaction attributes between services.

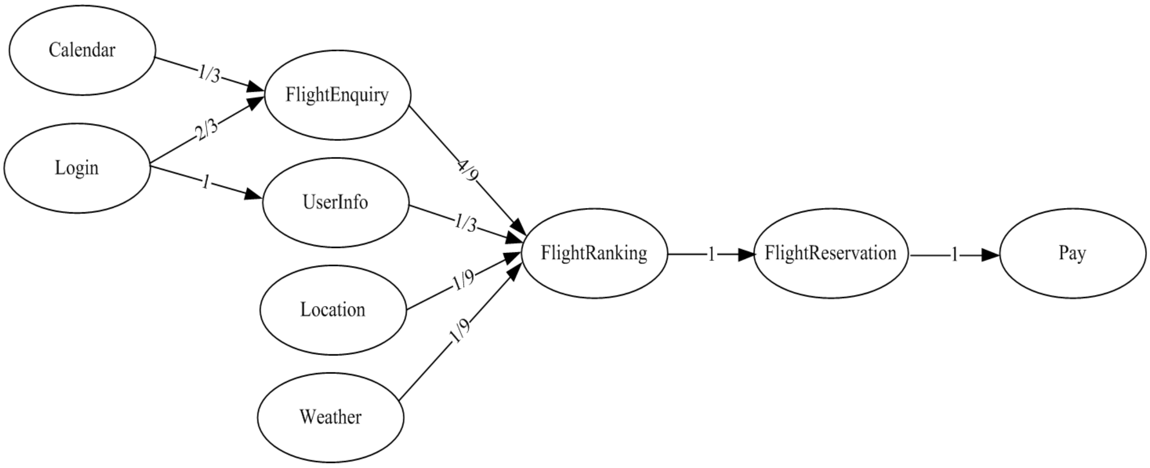


Figure 2. The SDGS of flight reservation Service-Oriented Systems

4.3. **Critical components identification based on IW-LeaderRank.** Although the LeaderRank algorithm removes uncertain parameters of the PageRank algorithm by adding public nodes, it accelerates the convergence speed of algorithm and improves the robustness of algorithm [24]. However, the dependency graph of SOSs is a weighted directed graph and edges in the graph have the weight of service dependency intensity. The original LeaderRank algorithm does not contain the weight coefficient [25]. Because components have different service heat, components with high heat need more attention, and adding service heat factor to the original algorithm can improve the accuracy of algorithm. In addition, components have different propagation capabilities due to their different locations, and combining the propagation capabilities of nodes can make the algorithm converge faster [26]. Therefore, we proposed the IW-LeaderRank algorithm in our previous work [5], which combines service dependency, service popularity and service propagation ability to measure the importance of components in SOSs. This paper adds the balance factor to three factors based on IW-LeaderRank, which can better measure the importance of components of SOSs. In this paper, the importance value of components is represented by LR. The calculation formula of final LR value is as follows:

$$LR_i = \alpha\left(LR_i(t_c) + \frac{LR_g(t_c)}{N}\right) + \beta H_i + \pi BC_i \tag{1}$$

Among them, $LR_i(t_c)$ is calculated by node i after adding the service dependency strength. t_c represents the number of convergence times. $LR_g(t_c)$ is the LR value of public node in the steady state. At each iteration, the LR value of the public node is divided equally to each node. H_i represents the service heat value of node i, that is, the frequency of service components being invoked in SOSs. BC_i represents the betweenness centrality of node i, which is used to control the propagation ability of services. α, β, π are the coefficients of three factors, and their sum is 1. The IW-LeaderRank algorithm can calculate the importance value of each component more intuitively and fairly and give a more reasonable ranking sequence of component importance.

IW-LeaderRank can obtain LR value and ranking of components in SOSs. If the importance value is higher and the ranking is also higher, it means the service is more important. In order to obtain critical components, this paper selects top-k components as critical components according to the LR ranking list of components. Due to the difference in service composition structure and service reliability value, the proportion of core components of each system is different. Therefore, we should select top-k based on experiences or actual conditions.

5. **Reliability Prediction for Service-Oriented Systems.** In this chapter, the LSTM neural network based on attention mechanism is used to analyze the historical reliability data of critical service components and obtain the reliability prediction value of critical components. Then, the Depth-First-Search algorithm is used to construct the critical service components model graph and the composition relationship of critical service components is obtained. Finally, the reliability prediction value of SOSs is obtained by using the reliability calculation method of SOSs.

5.1. **Reliability prediction model for components based on Attention-LSTM.** This paper combines the LSTM neural network [27] with attention mechanism [28] and proposes a reliability prediction model for components of SOSs based on attention-LSTM neural network. The model uses the attention mechanism to assign different weights to the features of LSTM hidden layer, highlighting important information in the input time series data. Therefore, the improved neural network model can better extract data features and improve the predictive ability of model. The Attention-LSTM neural network model is shown in Figure 3. X_t is the input sequence of model, h_t is the feature of LSTM hidden layer learning the input sequence X_t. $A_t = h_t a_t$, a_t is the weight of attention mechanism of hidden layer feature h_t. Y_t is the predicted result of component reliability.

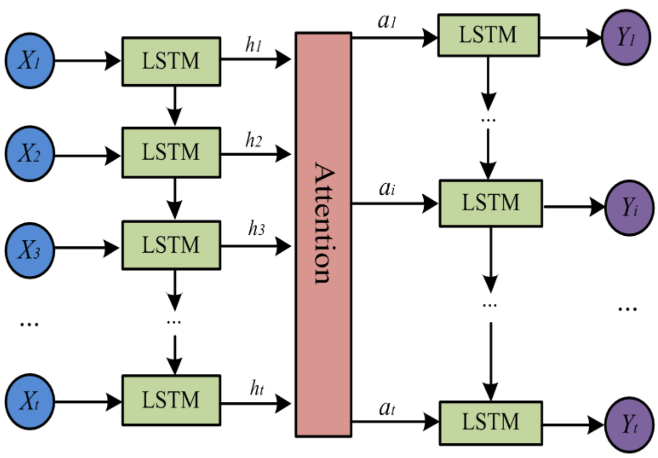


FIGURE 3. Attention-LSTM model

The Attention-LSTM neural network model proposed in this paper is composed of input layer, LSTM hiding layer, attention mechanism layer, LSTM hiding layer and output layer. The input layer inputs time-series data that affect the reliability of critical components. Due to the change of QoS natures, service load and other factors, the reliability of services will be affected. In order to accurately predict the reliability of services, the Attention-LSTM neural network in this paper obtained the variation rules of service reliability by learning the timing characteristics of four factors of service reliability R, correctness C, availability A and throughput T. In order to eliminate the impact of different data dimensions, the data of different factors are normalized to map the data to the range of 0∼1. These four factors are used as the input vector of Attention-LSTM neural network, expressed as $X_i = \{R_i, C_i, A_i, T_i\}$.

The LSTM hidden layer uses forget gates, input gates, output gates, and memory cells to forget unimportant information and save important information, which is conducive to better obtaining time series data laws. As the interface of two LSTM hidden layers, the attention mechanism layer assigns different weights to the features of first LSTM neural network hidden layer, which makes the network pay more attention to important information, thus highlighting the key sequence data. In order to retain more data information, the attention mechanism layer does not perform a weighted summation operation. It directly multiplies the features of hidden layer of first LSTM neural network with the attention weight and inputs them into the hidden layer of latter LSTM, which is conducive to the training of second LSTM hidden layer and obtain time series data characteristics. The reliability prediction value of critical components is output after learning through second LSTM hidden layer.

5.2. **Construction of critical services composition model graph.** Due to the fact that the composite structure among critical services is not unique, it is not possible to simply use a composite structure formula to calculate the reliability value of SOSs [29]. This paper uses the Depth-First-Search algorithm of graphs to remove unimportant components in the service composition model graph. Thus, the composition structure among critical components is preserved. Then, the critical services composition model graph is constructed to describe the composition structure of critical components in SOSs.

The following takes the flight reservation system as an example to illustrate the construction algorithm. In Figure 4, on the left is services composition model graph of the original system and on the right is critical services composition model graph of system. Finally, $s_{FlightRanking}$, $s_{FlightReservation}$, $s_{FlightEnquiry}$ are determined to be critical components of the flight reservation system.

It can be seen from the figure above that the structure of critical services is retained in the process of constructing the critical services composition model graph of SOSs. Therefore, the integrity of overall process of system is maintained. When critical components are used to calculate the reliability prediction value of SOSs, it can not only reduce the calculation amount and improve the prediction speed, but also improve the accuracy of reliability prediction results.

5.3. **Reliability calculation method.** As shown in Figure 5, the critical components composition model graph of SOSs has four structures: sequence, branch, loop and parallel.

The reliability prediction value of SOSs can be calculated according to the composite structure among critical components. The model of each structure and its reliability calculation method are given below.

It can be seen from Figure 5(a) that services are executed sequentially in the sequence structure. Therefore, when any of services in the structure fails, the sequence structure will fail. The reliability calculation process of sequence structure is expressed as Formula (2).

$$R_{Sequence} = \prod_{i=1}^{n} R_i \tag{2}$$

Among them, n is total number of tasks included in the sequence structure and R_i is the reliability value of the ith task.

As shown in Figure 5(b), the service is executed in a loop. If the number of loops is n, the reliability of the loop structure is expressed as Formula (3).

$$R_{Loop} = R_i^n \tag{3}$$

where n is an integer, which represents the number of executions of task S_i, and R_i is the reliability of task S_i.

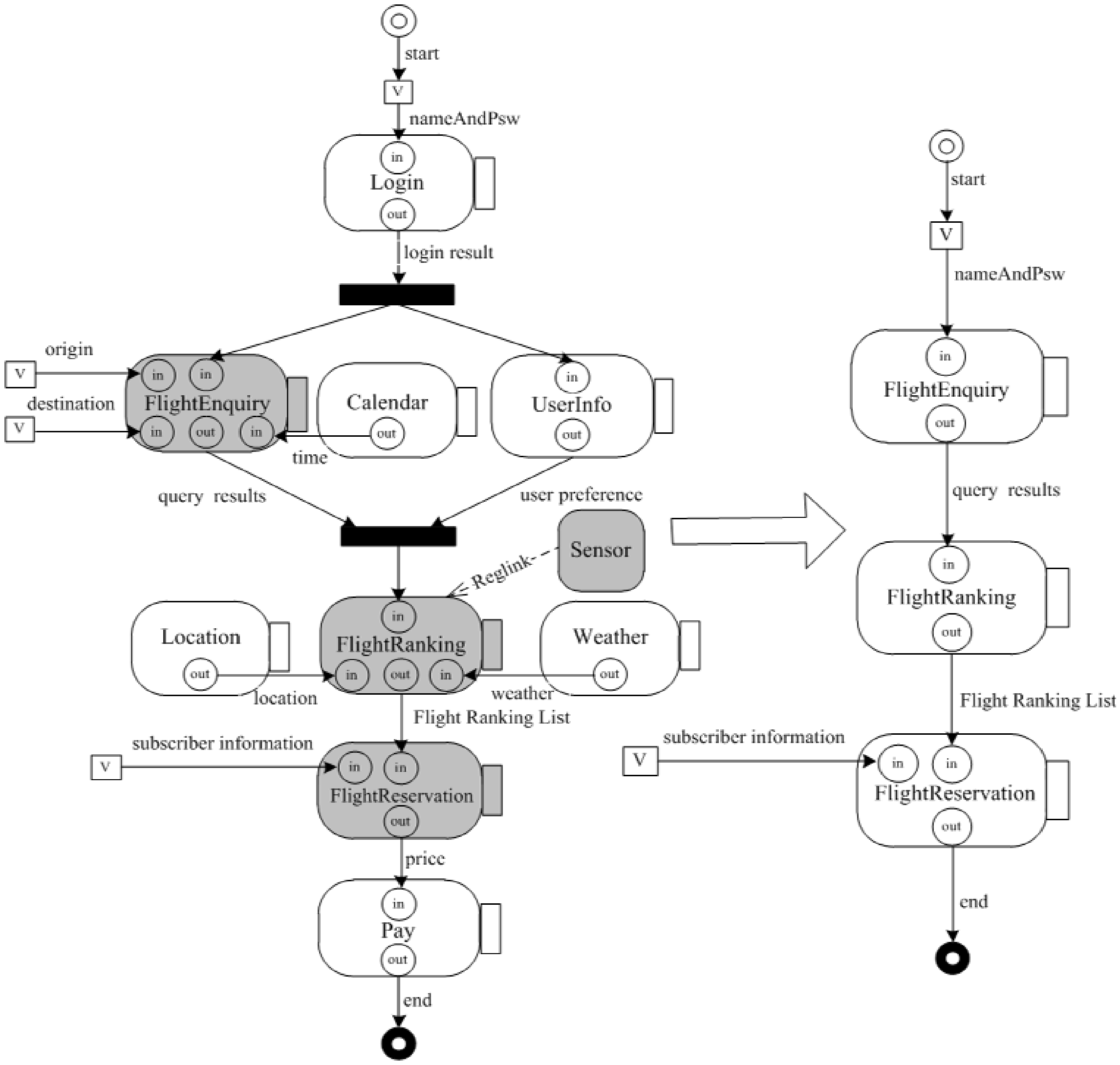
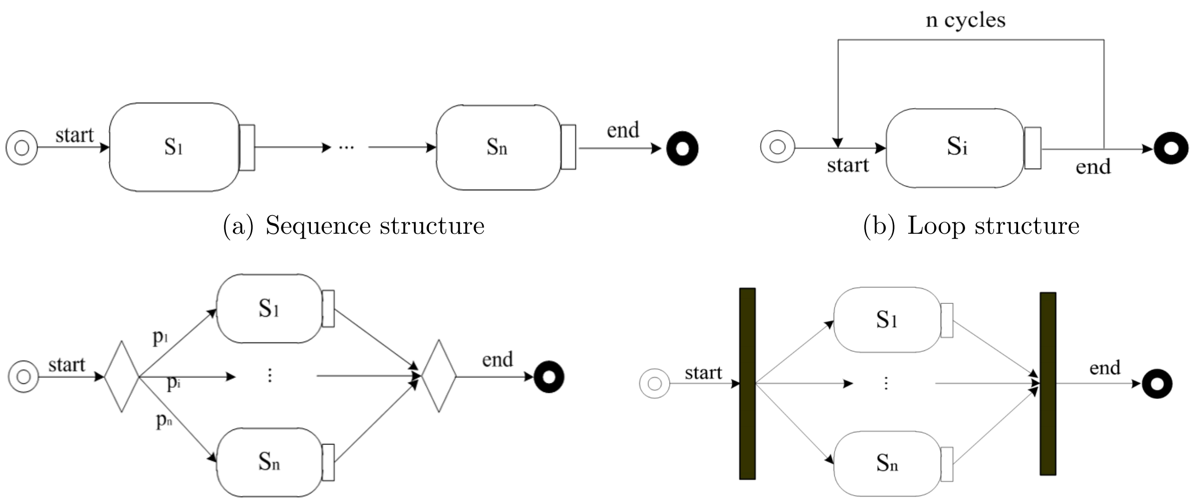


FIGURE 4. Construction of critical services composition model of flight reservation system



(a) Sequence structure (b) Loop structure

(c) Branch structure (d) Parallel structure

FIGURE 5. Four structures of Service-Oriented Systems

As shown in Figure 5(c), the branch structure executes only one branch at a time. The reliability calculation formula of branch structure is expressed as (4).

$$R_{Branch} = \sum_{i=1}^{n} p_i \times R_i \tag{4}$$

Among them, n is the total number of branch tasks included in the branch structure, p_i is the execution probability of the ith branch and R_i is the reliability value of the ith branch. Among them, $\sum_1^n p_i = 1$.

As shown in Figure 5(d), all branches are executed simultaneously in a parallel structure. The execution probability of each branch is 1. The reliability of parallel structure is expressed as Formula (5).

$$R_{Parallel} = 1 - \prod_{i=1}^{n}(1 - R_i) \tag{5}$$

Among them, n is the total number of tasks included in the parallel structure. R_i is the reliability of task S_i.

According to the calculation formulas of above four structures, the reliability prediction value of SOSs can be calculated. For example, in Figure 4, components in the graph after conversion only have a sequential structure. Therefore, the reliability prediction value of flight reservation SOSs can be obtained by Formula (6), and the calculation formula is shown below.

$$R_{SOS} = R_{FlightEnquiry} \times R_{FlightRanking} \times R_{FlightReservation} \tag{6}$$

6. **Experiment.** The experimental environment of this paper was Windows 10 (64-bit) operating system. The hardware configuration was Intel (R) Core (TM) i7-8700CPU@3.20 GHz 16 G RBM and 237 G solid state drive. The development language was python3.6, the development framework was Keras, and the back-end engine was TensorFlow.

6.1. **Experimental setup.** This section takes travel SOSs as experimental object, which is designed and implemented by our laboratory. Travel SOSs are constructed by service APIs (such as Programmable Web, and Amap) exposed by multiple websites and services developed by our laboratory. It includes positioning, location search, pedestrian navigation, bus navigation, driving route planning, ride service, payment and other functions, which is used for daily travel system. The system mainly includes 28 services, of which composite services are represented as atomic services. For convenience, node numbers are used below to represent service components in the system, as shown in Table 3. The experiments in this paper are based on travel SOSs for comparison and analysis.

According to the construction method of SDGS, the service dependency graph of travel SOSs can be obtained, as shown in Figure 6.

[30] showed that the most important service components for all systems are at the top of sort sequence. Therefore, we measure the importance of components of SOSs by the IW-LeaderRank algorithm. Finally, according to the identification method of critical components in Section 4.3, the values of components in the system are normalized and arranged in order. The results are shown in Table 4.

In this experiment, top-12 is selected as the critical components. Therefore, critical components of travel SOSs are V_{12}, V_{17}, V_{18}, V_7, V_{19}, V_{11}, V_{20}, V_{22}, V_{24}, V_{14}, V_{16}, V_{13}. Critical components are built into the critical services composition model graph of travel SOSs by Depth-First-Search algorithm. In order to facilitate the display, we removed input and output terminals and parameters of components. The critical services composite model graph of travel SOSs is shown in Figure 7.

Table 3. Travel Service-Oriented Systems component node number

Node number	Component name	Node number	Component name
V_1	Login	V_{15}	ETA
V_2	IPLocation	V_{16}	Price
V_3	Map	V_{17}	TaxiOrders
V_4	PlaceSerach	V_{18}	OrderReply
V_5	AutoComplete	V_{19}	ShowAroundDrivers
V_6	Regeo	V_{20}	OrderDetails
V_7	Products	V_{21}	OrderCancel
V_8	OtherTraffic	V_{22}	OrderForm
V_9	RoutePlanning	V_{23}	DriversComment
V_{10}	RouteDraw	V_{24}	AlipayTradeCreate
V_{11}	Traffic	V_{25}	AlipayTradeClose
V_{12}	ChooseTaxi	V_{26}	TradePay
V_{13}	Drivers	V_{27}	CloseRefund
V_{14}	ProductsStandard	V_{28}	SuccessRefund

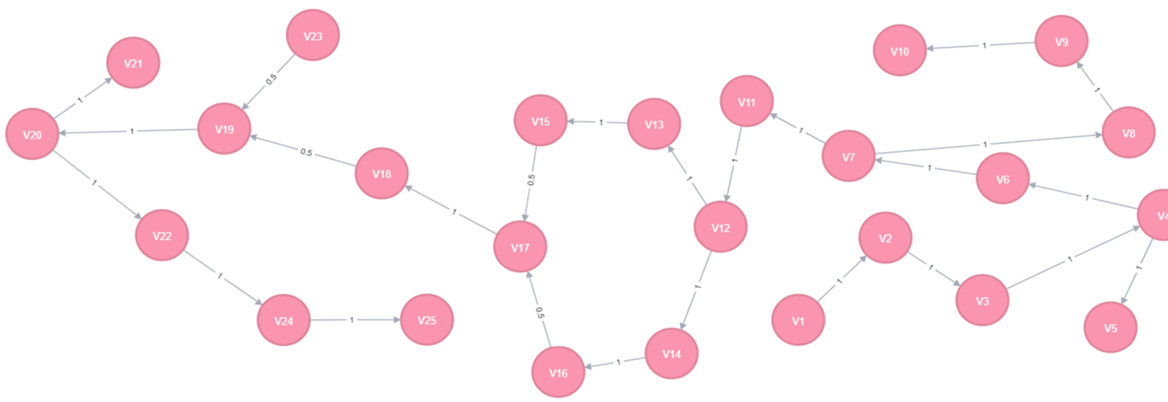


Figure 6. SDGS of travel Service-Oriented Systems

Table 4. Normalized LR value of travel Service-Oriented Systems

Node number	LR value	Node number	LR value	Node number	LR value
V_{12}	0.060891	V_{16}	0.037565	V_1	0.022500
V_{17}	0.058538	V_{13}	0.037394	V_{23}	0.021337
V_{18}	0.057824	V_{15}	0.037294	V_{10}	0.019809
V_7	0.056550	V_6	0.037117	V_{28}	0.019051
V_{19}	0.055537	V_4	0.035013	V_5	0.018187
V_{11}	0.055138	V_3	0.032892	V_{25}	0.017711
V_{20}	0.050488	V_8	0.028579	V_{21}	0.015208
V_{22}	0.047160	V_9	0.026669	V_{27}	0.014935
V_{24}	0.046424	V_2	0.026365		
V_{14}	0.038783	V_{26}	0.025041		

According to the reliability calculation method for SOSs, the reliability prediction value calculation formula of travel SOSs can be obtained as in Formula (7).

$$R_{SOS} = R_{V_7} \times R_{V_{11}} \times R_{V_{12}} \times (1 - (1 - R_{V_{13}}) \times (1 - R_{V_{16}} \times R_{V_{14}})) \times R_{V_{17}} \times R_{V_{18}} \times R_{V_{19}} \times R_{V_{20}} \times R_{V_{22}} \times R_{V_{24}} \quad (7)$$

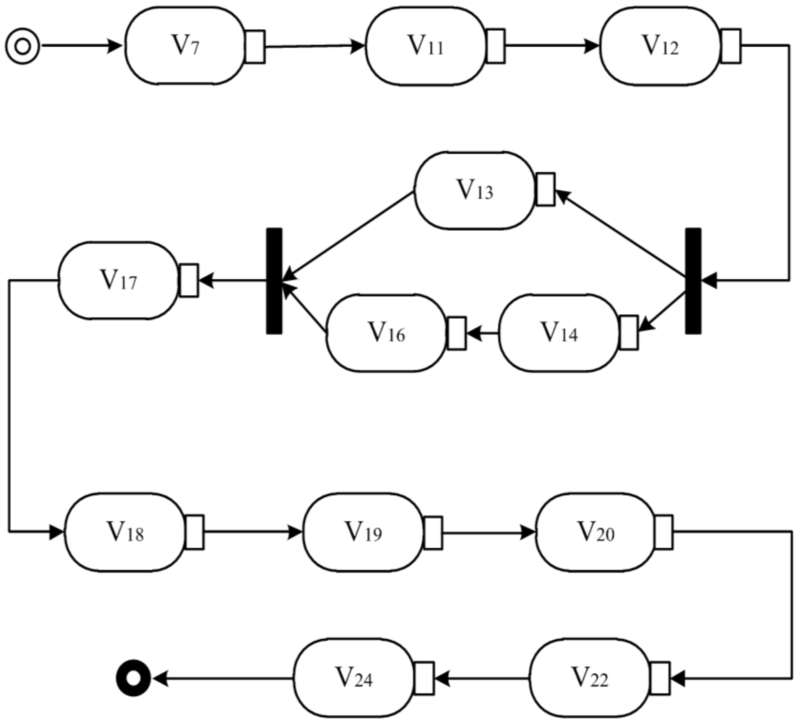


FIGURE 7. Critical services composition model graph of travel Service-Oriented Systems

6.2. **Experimental dataset and parameter settings.** The experimental data of this paper are collected from the travel Service-Oriented Systems. The reliability, correctness, availability and throughput time series data are collected for 90 consecutive days at 10 minute intervals. In this paper, the training set and test set are divided by 8 : 2. This paper uses Attention-LSTM neural network as the prediction model, and its main parameter settings are shown in Table 5.

TABLE 5. Attention-LSTM network parameter value

Parameter	Value
Batch size	128
Number of iterations	100
Optimizer	Adam
Learning rate	0.001
LSTM unit size	64
Sliding window length	10
Input dimension	4
Output dimension	1

6.3. **Experimental evaluation metrics and baseline methods.** In this section, MAE (Mean Absolute Error) and RMSE (Root Mean Square Error) are used as the algorithm evaluation metrics, which can represent the gap between real value and predicted value of component reliability. If the value is smaller, gap is smaller and algorithm is more effective. The calculation formulae are shown below.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|P_i - R_i| \tag{8}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i - R_i)^2} \tag{9}$$

In the formulae, N is the sample number of experimental predictions, R_i is actual reliability value and P_i is reliability value predicted by experimental method.

In order to verify the effectiveness of this method, we selected three methods as the baseline to test the reliability prediction effect of each method.

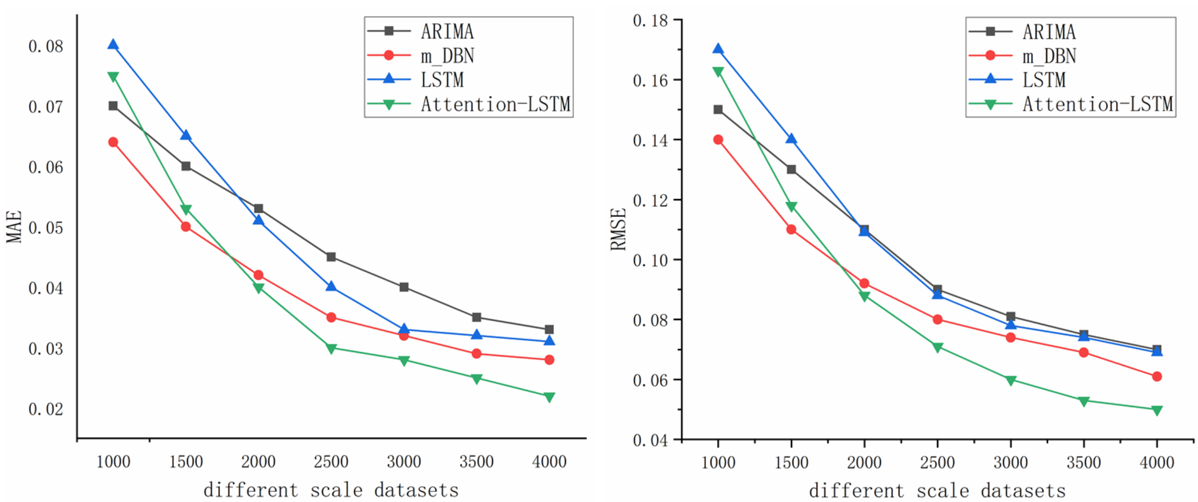
1) RARIMA [11]: RARIMA is a reliability prediction method based on the traditional statistical model Autoregressive Integrated Moving Average (ARIMA). While ARIMA is simple and easy to use, it requires that the timing data be stable. In addition, only linear relations can be captured in nature, but not nonlinear relations.

2) RLSTM [14]: RLSTM is an online reliability time series prediction method based on Long Short-Term Memory (LSTM). Although LSTM has certain advantages in sequence modeling, it cannot focus on the important parts of the sequence, and the prediction effect can still be improved.

3) ROP [16]: ROP is a reliability prediction method based on motifs-based Dynamic Bayesian Networks (m_DBNs). Dynamic Bayesian Network is a transient state model, which can learn the probabilistic dependence relationship between variables and the law of change over time. However, DBN has many assumptions, which require appropriate model design.

6.4. **Experimental results and comparative analysis.** This paper proposes a reliability prediction method for SOSs based on critical components identification, which is named RCCI. Attention-LSTM is selected as the prediction model. RCCI is compared with the reliability prediction method of SOSs based on LSTM (RLSTM), ROP method based on m_DBN and the reliability prediction method of SOSs based on ARIMA (RARIMA). The MAE results of SOSs reliability prediction by ROP, RLSTM, RARIMA and the method in this paper on different scale datasets are shown in Figure 8.

According to Figure 8, it can be found that the accuracies of component reliability predictions of four algorithms are not high when the amount of data is small. With the increase of training data, all methods can better grasp the characteristics of data. MAE and RMSE show a downward trend, indicating that predicted results are closer to the



(a) MAE (b) RMSE

Figure 8. MAE and RMSE results of four algorithms on different scale datasets

true value. Due to ARIMA algorithm requires the time series of samples to be stationary. However, component reliability data is affected by irregular network fluctuations. In a long period time, the reliability data of components cannot be stable. Therefore, ARIMA algorithm has better results when using a small amount of data. After increasing the scale of datasets, the accuracy of component reliability prediction results is lower than other algorithms. The m_DBN algorithm has strong predictive ability and its predictive accuracy is better than that of ARIMA algorithm. Although the Attention-LSTM method proposed in this paper has low prediction accuracy when training data on a small scale. As the data increases, its effect is significantly better than other algorithms. Moreover, by comparing the accuracy of prediction results on different scale datasets, it is found that improved Attention-LSTM algorithm is superior to LSTM algorithm, which indicates that Attention mechanism plays a positive role, enabling the model to better remember the effective information of time series and improve the accuracy of prediction. Although larger-scale data is more suitable for this algorithm than small-scale data, on the whole, this paper proposes that Attention-LSTM model has strong reliability prediction capabilities.

In addition, the reliability prediction efficiency of SOSs is also an important measurement standard. The method in this paper is based on the reliability prediction value of critical components, so as to obtain the reliability prediction value of the whole system. In order to prove the efficiency of this method, the prediction time of four reliability prediction methods on different scale training sets is calculated, and the results are shown in Figure 9.

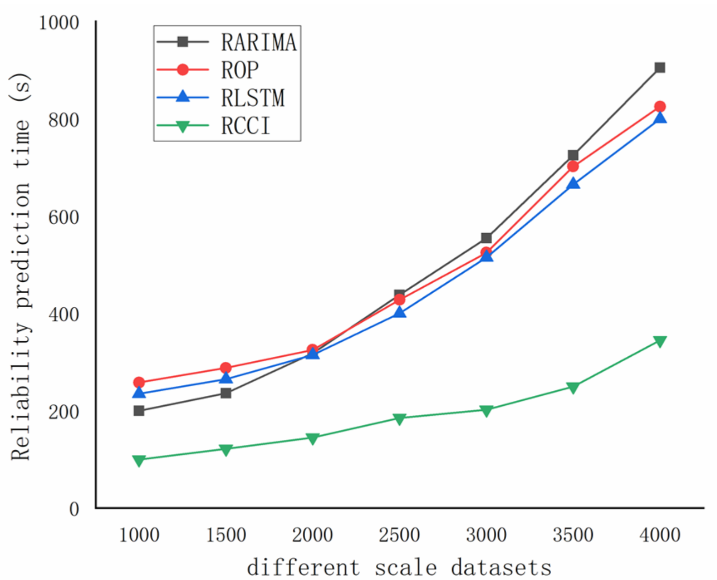


Figure 9. Prediction time of four algorithms on different scale datasets

It can be seen from Figure 9 that each algorithm needs to consume more time to get the reliability prediction value of SOSs with the increase of datasets scale. On the whole, the algorithm in this paper has obvious efficiency advantages. The reliability prediction process of other methods must first predict the reliability of each component. Then, the reliability prediction value of the system is calculated through the structure of components, and the phenomenon of longer prediction time will appear. The method in this paper identifies critical components of SOSs and uses critical components to obtain the reliability prediction value of SOSs, which can shorten a lot of prediction time. The method in this paper can alleviate the problem of slow response of prediction results and improve the efficiency of system reliability prediction.

7. **Conclusions.** Due to the fact taht SOSs run in an unstable network environment, the reliabilities of their components change dynamically. The reliability prediction of SOSs can know the reliability of system in advance and reserve time for taking effective measures. However, the process of calculating reliability of SOSs based on the reliability of each service will consume a lot of time. In order to make prediction results have higher accuracy while reducing the amount of calculation and ensuring that the prediction process is completed in a valid time, this paper proposes a reliability prediction method of SOSs based on critical components identification. This method proposes IW-LeaderRank algorithm to measure the importance of nodes in the service dependency graph of SOSs and obtain the sequence of critical components. The Attention-LSTM prediction model is used to predict the reliability of critical components. At the same time, the reliability calculation method of SOSs is given. The reliability prediction value of SOSs is obtained by the reliability prediction value of critical components and critical services composition model graph of SOSs.

Although this paper has carried out an in-depth study on reliability prediction of service system, there are still some shortcomings. The next research work of this paper is described below.

1) The effect of Attention-LSTM neural network on small scale data sets is not obvious, and there are disadvantages of long training time. Next, we can find a model with faster training and better prediction effect.

2) This paper takes the service system developed by our laboratory as the experimental object. Next, it can be applied to other service systems of different scales to verify the effectiveness of the method in this paper.

Acknowledgments. This work is supported by Liaoning Province Applied Basic Research Program Project (Grant No. 2023JH2/101300195).

REFERENCES

[1] S. Simanta, E. Morris and G. A. Lewis, Engineering lessons for systems of systems learned from service-oriented systems, *Proc. of IEEE International Systems Conference*, San Diego, pp.634-639, 2010.

[2] C. Kang, A device-life model for reliability demonstration test for a product made up of a large array of electronic devices, *International Journal of Innovative Computing, Information and Control*, vol.17, no.1, pp.167-175, 2021.

[3] Y. Zhen, S. Mistry, A. Bouguettaya et al., Long-term QoS-aware cloud service composition using multivariate time series analysis, *IEEE Transactions on Services Computing*, vol.9, no.3, pp.382-393, 2016.

[4] F. Salfner, R. Lenk and R. Malek, A survey of online failure prediction methods, *ACM Computing Surveys (CSUR)*, vol.42, no.3, pp.1-42, 2010.

[5] X. Zhang and S. Jiang, Critical components identification for service-oriented systems, *Symmetry*, vol.11, no.3, pp.427-440, 2019.

[6] Z. Zheng and M. R. Lyu, Collaborative reliability prediction of service-oriented systems, *Proc. of ACM/IEEE 32nd International Conference on Software Engineering (ICSE)*, Cape Town, South Africa, pp.35-44, 2010.

[7] Z. Zheng and M. R. Lyu, Personalized reliability prediction of web services, *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol.22, no.2, pp.12:1-12:25, 2013.

[8] M. Silic, G. Delac and S. Srbljic, Prediction of atomic web services reliability based on k-means clustering, *Proc. of the 9th Joint Meeting on Foundations of Software Engineering*, Saint Petersburg, Russian Federation, pp.70-80, 2013.

[9] M. Silic, G. Delac and S. Srbljic, Prediction of atomic web services reliability for QoS-aware recommendation, *IEEE Transactions on Services Computing*, vol.8, no.3, pp.425-438, 2015.

[10] A. Amin, L. Grunske and A. Colman, An approach to software reliability prediction based on time series modeling, *Journal of Systems and Software*, vol.86, no.7, pp.1923-1932, 2013.

[11] Z. Ding, T. Xu, T. Ye et al., Online prediction and improvement of reliability for service oriented systems, *IEEE Transactions on Reliability*, vol.65, no.3, pp.1-16, 2015.

[12] M. Sobhana, G. Preethi, G. H. Sri and K. B. Sujitha, Improved reliability prediction in engineering systems based on artificial neural network, *International Mobile and Embedded Technology Conference (MECON)*, Noida, India, pp.455-460, 2022.

[13] S. Yadav and Balkishan, Software reliability prediction by using deep learning technique, *International Journal of Advanced Computer Science and Applications*, vol.13, no.6, pp.683-693, 2022.

[14] H. Wang, Z. Yang and Q. Yu, Online reliability prediction via long short term memory for service-oriented systems, *IEEE International Conference on Web Services (ICWS)*, Honolulu, HI, USA, pp.81-88, 2017.

[15] H. Wang, Z. Yang and Q. Yu, Online reliability time series prediction via convolutional neural network and long short term memory for service-oriented systems, *Knowledge-Based Systems*, vol.159, pp.132-147, 2018.

[16] H. Wang, L. Wang, Q. Yu et al., Online reliability prediction via motifs-based dynamic Bayesian networks for service-oriented systems, *IEEE Transactions on Software Engineering*, vol.43, no.6, pp.556-579, 2017.

[17] H. Wang, L. Wang, Q. Yu et al., A proactive approach based on online reliability prediction for adaptation of service-oriented systems, *Journal of Parallel and Distributed Computing*, vol.114, pp.70-84, 2018.

[18] V. Challagulla, F. B. Bastani, R. A. Paul et al., A machine learning-based reliability assessment model for critical software systems, *Proc. of the 31st Annual International Computer Software and Applications Conference (COMPSAC)*, Beijing, China, pp.79-86, 2007.

[19] Y. Wang, Q. He, D. Ye et al., Formulating criticality-based cost-effective monitoring strategies for multi-tenant service-based systems, *Proc. of IEEE International Conference on Web Services (ICWS)*, Honolulu, HI, USA, pp.325-332, 2017.

[20] M. C. Chiang, C. Y. Huang, C. Y. Wu et al., Analysis of a fault-tolerant framework for reliability prediction of service-oriented architecture systems, *IEEE Transactions on Reliability*, vol.70, no.1, pp.13-48, 2021.

[21] A. Ali, M. B. Bashir, A. Hassan et al., Design-time reliability prediction model for component-based software, *Sensors*, vol.22, no.7, 2022.

[22] S. Fattaheian-Dehkordi, M. Fotuhi-Firuzabad and R. Ghorani, Transmission system critical component identification considering full substations configuration and protection systems, *IEEE Transactions on Power Systems*, vol.33, no.5, pp.5365-5373, 2018.

[23] H. Wang, Y. Ren, X. Zhang et al., Research on interpreter of context-aware cooperative composition modeling language, *Proc. of the 4th International Conference on Logistics, Informatics and Service Science (LISS)*, Berlin, Heidelberg, pp.1215-1221, 2014.

[24] L. Lü, Y. C. Zhang, C. H. Yeung and T. Zhou, Leaders in social networks, the delicious case, *PLoS ONE*, vol.6, no.6, e21202, 2011.

[25] L. Lü and L. Qian, Identifying influential spreaders by weighted LeaderRank, *Physica A: Statistical Mechanics and Its Applications*, vol.404, pp.47-55, 2014.

[26] Z. H. Zhang, G. P. Jiang, Y. R. Song, L. L. Xia and Q. Chen, An improved Weighted LeaderRank algorithm for identifying influential spreaders in complex networks, *Proc. of IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, vol.1, pp.748-751, 2017.

[27] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Computation*, vol.9, no.8, pp.1735-1780, 1997.

[28] Y. Zhu, C. Zhao, H. Guo, J. Wang, X. Zhao and H. Liu, Attention CoupleNet: Fully convolutional attention coupling network for object detection, *IEEE Transactions on Image Processing*, vol.28, no.1, pp.113-126, 2019.

[29] L. Xie and F. Wang, Reliability prediction model for web services based on control-structure, *Computer Science*, vol.38, no.B10, pp.92-95, 2011.

[30] S. Ioana, A PageRank based recommender system for identifying key classes in software systems, *Proc. of IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics*, pp.495-500, 2015.

Author Biography

Xiuguo Zhang received the Ph.D. degree in Computer Science from Dalian Maritime University, China. She is a professor at the School of Information Science and Technology, Dalian Maritime University. Her research interests include web services and software engineering. She has authored more than 70 refereed journal/conference papers.

Yun Zhao is a postgraduate student in Computer Science and Technology at Dalian Maritime University, China. He received a bachelor's degree from Liaoning University of Technology in June 2020. At the same time, he also won an outstanding graduate of Liaoning Province. His research focuses on the reliability of service-oriented systems. He has published two papers as the first author.

Zhiying Cao received the master degree in Management Science and Engineering from Dalian Maritime University, China. She is an associate professor at the School of Information Science and Technology, Dalian Maritime University. Her research interests include web services and software engineering. She has authored more than 40 refereed journal/conference papers.

Shuo Jiang received her master degree in Computer Science and Technology from Dalian Maritime University in 2020. Her research focuses on the reliability of service-oriented systems. She has authored three journal/conference papers. She is currently a software development engineer at Netshen Information Technology (Beijing) Co., Ltd.