Springer Proceedings in Business and Economics

Hui Yang
Robin Qiu
Weiwei Chen *Editors*

Smart Service Systems, Operations Management, and Analytics

Proceedings of the 2019 INFORMS International Conference on Service Science

Springer

Springer Proceedings in Business and Economics

Springer Proceedings in Business and Economics brings the most current research presented at conferences and workshops to a global readership. The series features volumes (in electronic and print formats) of selected contributions from conferences in all areas of economics, business, management, and finance. In addition to an overall evaluation by the publisher of the topical interest, scientific quality, and timeliness of each volume, each contribution is refereed to standards comparable to those of leading journals, resulting in authoritative contributions to the respective fields. Springer's production and distribution infrastructure ensures rapid publication and wide circulation of the latest developments in the most compelling and promising areas of research today.

The editorial development of volumes may be managed using Springer's innovative Online Conference Service (OCS), a proven online manuscript management and review system. This system is designed to ensure an efficient timeline for your publication, making Springer Proceedings in Business and Economics the premier series to publish your workshop or conference volume.

More information about this series at http://www.springer.com/series/11960

Hui Yang · Robin Qiu · Weiwei Chen
Editors

Smart Service Systems, Operations Management, and Analytics

Proceedings of the 2019 INFORMS International Conference on Service Science

Springer

Editors
Hui Yang
Harold and Inge Marcus Department of Industrial and Manufacturing Engineering
Pennsylvania State University
University Park, PA, USA

Robin Qiu
Division of Engineering and Information Science
Pennsylvania State University
Malvern, PA, USA

Weiwei Chen
Department of Supply Chain Management
Rutgers, The State University of New Jersey
Piscataway, NJ, USA

ISSN 2198-7246 ISSN 2198-7254 (electronic)
Springer Proceedings in Business and Economics
ISBN 978-3-030-30966-4 ISBN 978-3-030-30967-1 (eBook)
https://doi.org/10.1007/978-3-030-30967-1

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

Cleaning and Processing on the Electric Vehicle Telematics Data

Shuai Sun, Jun Bi and Cong Ding

Abstract The development of the Internet of Vehicles (IoV) enables companies to collect an increasing amount of telematics data, which creates plenty of new business opportunities. How to improve the integrity and precision of electric vehicle telematics data to effectively support the operation and management of vehicles is one of the thorniest problems in the electric vehicle industry. With the purpose of accurately collecting and calculating the driving mileage of electric vehicles, a series of data cleaning and processing methodologies were conducted on the real-world electric vehicle telematics data. More specifically, descriptive statistics was conducted on the data, and the statistical results showed the quality of the data in general. Above all, the driving mileage data were segmented according to the rotate speed of the electric motor, and the anomaly threshold of the driving mileage data was obtained by the box-plot method. Then, the typical anomalies in the data were screened out by the threshold and analysed, respectively. Ultimately, the real-time and offline abnormal processing algorithms are designed to process real-time and offline data, respectively. After debugging and improvement, these two sets of abnormal processing algorithms we designed have been able to run on a company's big data cloud platform. According to the feedback of the operation results of real-world massive data, the two sets of algorithms can effectively improve the statistical accuracy of driving mileage data of electric vehicle.

Keywords Internet of Vehicles · Telematics data · Data cleaning and processing · Box-plot method

1 Introduction

The Internet of Vehicles (IoV) is a vast interactive network of vehicles around information such as location, speed and routes [1]. It can realize information sharing through vehicle-to-vehicle, vehicle-to-person, vehicle-to-road interconnection and

S. Sun (✉) · J. Bi · C. Ding
School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China
e-mail: 17114261@bjtu.edu.cn

H. Yang et al. (eds.), *Smart Service Systems, Operations Management, and Analytics*, Springer Proceedings in Business and Economics,
https://doi.org/10.1007/978-3-030-30967-1_1

intercommunication, processing, sharing and releasing information collected from multiple sources on the information network platform. The Internet has brought society into the era of big data, and the Internet of vehicles has also brought vehicles into the era of big data [2]. This paper investigates the value of IoT data with particular attention to telematics data in the electric vehicle industry.

The commercial vehicle industry is one of the first to put the Internet of Vehicles technology into use due to its own industry background and industry demand [3]. The application of the Internet of Vehicles in the field of commercial vehicles, to a certain extent, solves the problem of lacking telematics data. However, in the actual process, incomplete, inaccurate and unreliable telematics data often fail to achieve the expected effect of supporting operation management. Therefore, how to improve the integrity and precision of electric vehicle telematics data to effectively support the operation and management of vehicles is one of the thorniest problems in the electric vehicle industry.

At the business level, the fleet management generally have the corresponding index requirements. Mileage is an important index of transportation cost accounting. Most of the driver's salary is linked to the mileage. Inaccurate mileage data is not conducive to the cost accounting and operation management of transportation enterprises. Accurate mileage and fuel consumption data is conducive to the promotion and application of commercial vehicle network in transportation enterprises. Therefore, our purpose is to find the existing problems in the driving mileage data of electric vehicles and propose solutions to these problems through cleaning, mining and processing.

2 Data Description

First of all, descriptive statistics have been carried out on the data to show the quality of data from the statistical results. There are 62 properties in each valid data file, and the key focus of this data analysis is some basic parameters, e.g. four mileage indicators and one standard mileage indicator, which is shown in Table 1.

Given that there is supplementary in data transmission, the missing distribution of mileage data was calculated. The statistical results of mileage data missing are shown in Table 2.

The statistical results show that some of the mileage data are missing. After preliminary observation, most of the missing cases were found in the supplementary data. Coincidentally, most of the supplementary data are out of driving trip. Therefore, it is very necessary to segment trips, which can effectively improve data quality.

Table 1 Basic information of key data items

Type	Attributes	Use for
Basic parameters	ID	Identify the terminal
	GPS time	Identify the time
	ACC	Segment the trip
	Motor speed	Segment the trip
	Supplementary	Identify the missing
Driving mileages	Mileage 1	Vehicle mileage
	Mileage 2	ECU mileage
	Mileage 3	GPS mileage
	Mileage 4	Integral mileage
	Standard mileage	Tagged mileage

Table 2 Percentage of mileage data missing

Mileage attributes	Missing percentage (%)
Mileage 1	6.18
Mileage 2	18.74
Mileage 3	0
Mileage 4	2.57
Standard mileage	10.51

3 Data Processing

The statistical analysis results of the data enable us to have a preliminary understanding of the data, but in order to find the problems in the data and propose solutions to the problems, the data needs to be processed.

3.1 Trip Segmentation

According to practical experience, ACC cannot be used as the basis of trip segmentation because it has been always ON in some cases. Therefore, the segmentation of trips is mainly based on motor speed. That is, the first point whose motor speed is not 0 counts as the start point of a trip, and the point whose motor speed is reduced to 0 and lasts for 5 min counts as the end point of that. According to the above rules, the vehicle's driving trip is segmented, and a typical example of trip segmentation is shown in Fig. 1.

As shown in the figure below, we can clearly see seven complete trips. Good trip segmentation results will help us to process and clean the data.

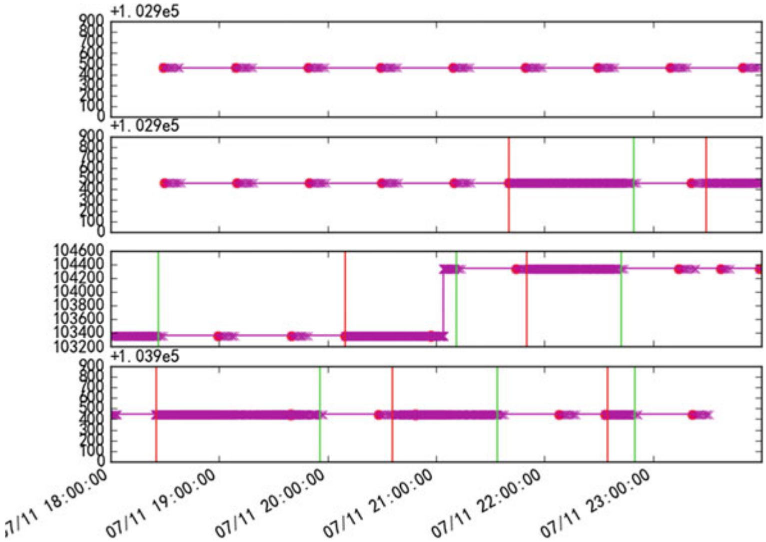

Fig. 1 Result of segmentation of trips

3.2 Threshold Selection

After the trip is segmented, outliers in the data can be filtered. Here, we define a new speed indicator, that is, the ratio of the distance difference between two adjacent points to the time difference between two adjacent points (unit: km/s), as a criterion to screen data outliers. For the speed indicator we defined, we used box-plot method and three-delta method to conduct statistical screening of all the data, and some results are shown in Table 3.

According to the comparison results in Table 3, the result of box-plot method is more stable than that of three-delta method, and the speed threshold obtained by the former method is of more practical reference value. Therefore, box-plot method is used in the threshold selection and subsequent data processing.

Table 3 Comparison of box diagram method and three-delta method

	Box diagram method	Three-delta method
Count	88	78
Mean	0.038069	0.20065
Std	0.031169	0.118598
Min	0.004833	0.049026
25%	0.0275	0.115383
50%	0.0325	0.176352
75%	0.037125	0.238261
Max	0.2	0.685202

4 Data Cleaning

4.1 *Real-Time Algorithm*

The basic idea of real-time algorithm is calculating the average speed of the latest two points of high-priority data, comparing to the predetermined speed threshold. If it is within the threshold range, the mileage difference between these two points is selected for accumulation; if it is beyond the threshold range, the two-point mileage difference with low priority in the simultaneous segment that meets the threshold requirements is selected for accumulation [4].

4.2 *Offline Algorithm*

Taking the data with the trip as the research object, the basic idea of offline algorithm is segmenting the data according to the threshold selected in advance. In the normal segment, the value of the maximum moment minus the value of the minimum moment is used to obtain the mileage difference. In the segment with abnormal conditions, the real-time algorithm is used to traverse the data to obtain the mileage accumulation value. Finally, the processing results of each segment are summed to obtain the total mileage difference value.

The results of the two algorithms are shown in Fig. 2. After debugging and improvement, these two sets of abnormal processing algorithms we designed have been able to run on a company's big data cloud platform. According to the feedback of the operation results of real-world massive data, the two sets of algorithms can effectively improve the statistical accuracy of driving mileage data of electric vehicle.

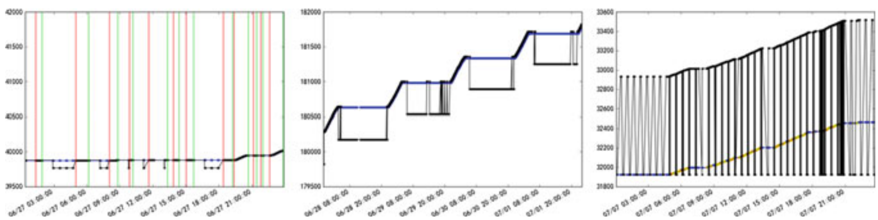

Fig. 2 Algorithm processing results

5 Summary

With the fundamental purpose of accurately counting the mileage of Internet of Vehicles data, a series of data cleaning and processing have been carried out for telematics data. In the following work, we will constantly adjust the existing algorithms according to the actual business rules, so that the real-time algorithm and the offline algorithm can meet the market demand.

The electric vehicle telematics data have not been made available because we signed a confidential agreement with an IoV company, and all telematics data related to commercial secrets is not suitable for disclosure.

Acknowledgements This research is supported by the National Key R&D Program of China under grant No. 2018YFC0706005 and No. 2018YFC0706000.

References

1. S. Duri, J. Elliott, M. Gruteser, X. Liu, P. Moskowitz, R. Perez et al., Data protection and data sharing in telematics. Mob. Netw. Appl. **9**(6), 693–701 (2004)
2. J.F. Ehmke, Data chain management for planning in city logistics. Int. J. Data Min. Model. Manag. **1**(4), 335–356 (2009)
3. I. Reimers, B. Shiller, *Welfare Implications of Proprietary Data Collection: An Application to Telematics in Auto Insurance* (Social Science Electronic Publishing, 2018)
4. J. Lauer, L. Richter, T. Ellersiek, A. Zipf, TeleAgro+: analysis framework for agricultural telematics data, in *ACM SIGSPATIAL International Workshop on Computational Transportation Science* (ACM, 2014)

Performance Analysis of a Security-Check System with Four Types of Inspection Channels for High-Speed Rail Stations in China

Chia-Hung Wang and Xiaojing Wu

Abstract In recent years, the High-Speed Rail (HSR) in China has continued to thrive rapidly with the development of China's booming economy, and it has become the preferred mode of transportation for many travelers. This paper investigates the stochastic process of security inspection for passengers in the high-speed rail station. A queuing model is developed for studying the proposed security-check system via computer simulation. In the numerical experiments, we illustrate the influence of varying model parameters on the average waiting time and safety level of the queuing system. The sensitivity analysis of our simulation model could contribute to improve the service level and efficiency on security check before implementing operation plans for high-speed rail stations.

Keywords Security check · Queuing system · Rail station · Homeland security management · System performance

1 Introduction

High-Speed Rail (HSR) in China, the country's network of passenger-dedicated railways, has been the world's longest high-speed rail network [1]. In 2018, HSR has extended to 30 of China's 33 provincial-level administrative divisions and accounted for almost two-thirds of the world's high-speed rails in commercial service. The total length of HSR network in China is going to reach 30,000 km in 2020 and 38,000 km in 2025 according to The Mid-to-Long-Term Railway Network Plan [2], issued by National Development and Reform Commission of the People's Republic

C.-H. Wang (✉) · X. Wu
College of Information Science and Engineering, Fujian University of Technology, Fuzhou City 350118, Fujian Province, China
e-mail: jhwang728@hotmail.com

C.-H. Wang
Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fuzhou City 350118, Fujian Province, China

H. Yang et al. (eds.), *Smart Service Systems, Operations Management, and Analytics*, Springer Proceedings in Business and Economics,
https://doi.org/10.1007/978-3-030-30967-1_2

of China. The demand for HSR in China steadily increases over time, whereas the public concerns about its efficiency and safety.

The security screening on passengers is a key step for entering the quarantine area of rail stations from the public areas. According to the Anti-Terrorism Act of the People's Republic of China [3], the management units of high-speed rail stations should conduct security checks on the passengers, personal belongings, and vehicles that enter the stations. The growing terrorist threats make the research on security-check systems become a hot issue [4–6]. Interested readers may refer to [7–12] and references therein. From the related works in [13], it shows that the security checkpoint is a bottleneck in the passenger traveling process at the high-speed rail station. With the increase of passenger flow, passengers often need to wait in a long line for security check due to the complexity of the security inspection process, which would significantly reduce passenger satisfaction [14]. Therefore, it's important and necessary to set up a research on improving the security-check efficiency while ensuring safety.

In this paper, we formulate the passenger security-check process as a queuing model with four types of inspection channels. Based on the Arena simulation platform, we observe the effect on the average waiting time of passengers and safety level of the security-check system through varying each interested model parameter. The main contribution of this work is supposed to perform a sensitivity analysis for managing the studied security-check system at the high-speed rail stations.

2 Security-Check Mode of High-Speed Rail Station

The purpose of the security check is to prevent passengers from carrying items that may endanger safety into the high-speed rail stations or trains [15]. The security screening on incoming passengers includes the inspection for personal identity, body check, and carry-on baggage. When passengers are going into the security checkpoints of rail stations, their identity cards, and train tickets will be inspected by the ticket inspectors. After the ticket is inspected, the passenger puts his/her belongings into a tray which is going to be inspected via the X-ray machine, while he/she passes the security door for body check. After the security-check procedure is completed, the passenger picks up and organizes his/her belongings or baggage, then leaves the security checkpoint. There are two possibilities in the inspection process of passengers' baggage: containing or without contraband. Some suspicious passengers who need to open their baggage would go through additional complete inspection.

There are two kinds of signal cases in the output of inspection process: alarm and clear (no alarm). It may result in the following four possibilities: (1) Correct Alarm: Raise the alarm when passenger carries a threat or contraband; (2) Correct Clear: Pass the inspection when there is neither a threat nor contraband; (3) False Alarm: Raise the alarm when there is neither a threat nor contraband; (4) False Clear: Pass the inspection when passenger carries a threat or contraband.

3 A Queuing Model with Four Types of Inspection Channels

As shown in Fig. 1, there are four types of inspection channels in the security-check system of high-speed rail station, including Green channel, Strict inspection channel, Normal inspection channel, and Fast-Pass inspection channel. Based on passengers' identities and attributes, they are differentiated into four different risk classes and then go through the corresponding type of inspection channels. It is assumed that incoming passengers are inspected based on a First-Come-First-Served order. All model parameters of the studied queuing model are summarized in Table 1.

The description of the proposed queuing model is introduced as follows. The overall arrival rate of passengers to the high-speed rail station is denoted as λ, and it holds that $\lambda = \lambda_g + \lambda_s + \lambda_n + \lambda_f$. The Green channel is designed to help special passengers quickly complete security checks, such as VIPs, government officials, and soldiers on duty. We denote μ_g as the average service rate of Green channel, and the symbol λ_g represents the average arrival rate of passengers to the Green channel. While in Normal inspection channel, the security personnel conduct a routine inspection of ordinary passengers. The average arrival rate of passengers to the Normal inspection channel is given as λ_n, and the average service rate is μ_n. Note that the queue length of Normal inspection channel is limited and denoted as the finite integer number M.

In Fast-pass inspection channel, those passengers without baggage are carried out a quick and simple inspection procedure. According to the practical experience, the inspection time for personal baggage is much longer than personal inspection time. The average arrival rate of passengers to the Fast-pass inspection channel is set as λ_f, and the average service rate is given as μ_f. On the other hand, there is a complicated and strict inspection procedure for suspicious passengers with baggage in Strict inspection channel, which results in longest security-check time, whereas the highest recognition rate for "threats". We denote λ_s as the average arrival rate of passengers to the Strict inspection channel, and the average service rate of Strict inspection channel is given as μ_s.

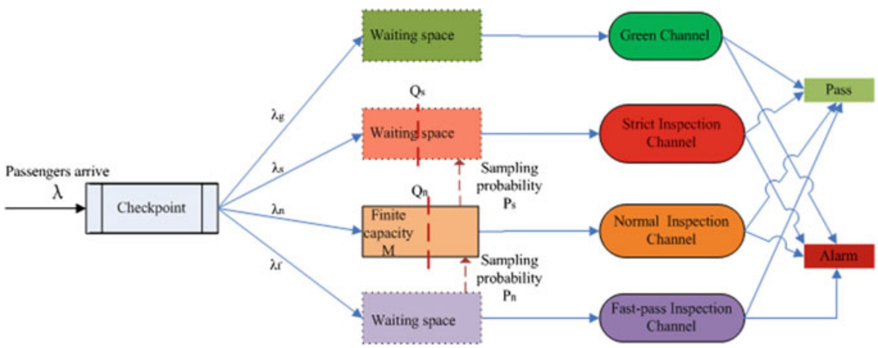

Fig. 1 An illustration of queuing modeling of a security-check system with four types of inspection channels

Table 1 Notations for the model parameters

Notation	Definition
λ_g	Average arrival rate of passengers to the Green channel
λ_s	Average arrival rate of passengers to the Strict inspection channel
λ_f	Average arrival rate of passengers to the Fast-pass inspection channel
λ_n	Average arrival rate of passengers to the Normal inspection channel
λ	The overall arrival rate of passengers to the high-speed rail station
μ_g	Average service rate of Green channel
μ_s	Average service rate of Strict inspection channel
μ_f	Average service rate of Fast-pass inspection channel
μ_n	Average service rate of Normal inspection channel
M	Finite queue length of Normal inspection channel
γ_n	Risk threshold used to differentiate the normal and low-risk passengers
γ_s	Risk threshold used to differentiate the normal and high-risk passengers
Q_n	Queue threshold used to draw out a proportion of passengers from Fast-pass inspection channel
Q_s	Queue threshold used to draw out a proportion of passengers from Normal inspection channel
P_n	Sampling probability of passengers from Fast-pass inspection channel
P_s	Sampling probability of passengers from Normal inspection channel

The security-check system will assign passengers into the corresponding inspection channels in accordance to the risk threshold value. The risk threshold γ_n and γ_s are used to differentiate the risk value of high-risk and low-risk passengers in the security-check system of the high-speed rail station. When a passenger's risk value is smaller than a given risk threshold γ_n, he/she could go to the Fast-pass inspection channel for security check. If his/her risk value is between two risk thresholds γ_n and γ_s, he/she would go to the Normal inspection channel. For those suspicious passengers whose risk values are larger than a given risk threshold γ_s, he/she ought to go through complete inspection in Strict inspection channel. It holds that $0 \leq \gamma_n \leq \gamma_s \leq 1$.

The queue thresholds Q_n and Q_s in the higher level inspection channels are used to draw out a proportion of passengers in the lower level inspection channels. The sampling probability P_n indicates the proportion of passengers entering Normal inspection channel from Fast-pass inspection channel when the current queue length in Normal inspection channel is less than Q_n. The sampling probability P_s represents the proportion of passengers entering Strict inspection channel from Normal inspection channel when the current queue length in Strict inspection channel is less than Q_s.

In this paper, the system performance of the studied security-check system includes the following two performance measures: the average waiting time of passengers and safety level of the system. Here, the safety level of the studied high-speed rail station is defined as 100% minus the percentage of "simulation threats" passed the mistaken inspection channels.

4 Numerical Results

In this section, the relationship between the model parameters and the system performance will be illustrated by performing multiple sets of experiments on the change of each decision variable. As shown in Table 2, our experimental settings are taken from data collected at the Changsha South High-Speed Station. The settings of our model parameters are given as follows: $\gamma_n \in [0, 0.145]$, $\gamma_s \in [0, 0.99]$, $Q_n \in [1, 20]$, $Q_s \in [1, 20]$, $P_n \in [0, 1]$, $P_s \in [0, 1]$, and $M \in [1, 25]$. Here, the average service rate at Green channel is $\mu_g = 16.5$ people per minute. The average service rate of Fast-pass inspection channel is $\mu_f = 13.4$ people per minute. The average service rate of Normal inspection channel is $\mu_n = 11$ people per minute, and the average service rate at Strict inspection channel is $\mu_s = 3.083$ people per minute. Our numerical experiments are run by using Arena simulation software version 14.00 installed on the PC platform with Intel Core i5-2520M (2.5 GHz) and 8 GB RAM. In each simulation experiment, the simulation will be carried out with sufficient running time and repeated times.

In the following experiments, we analyze the effect on the average waiting time and the safety level of the proposed queuing system by varying a certain decision variable. We conduct a sensitivity analysis to study the managerial effect of model parameters on the interested system performance. Our numerical results are illustrated in Figs. 2, 3, 4, 5, 6, 7, and 8.

From Fig. 2, when we increase the value of risk threshold γ_n, it can be observed the average waiting time decreases first and then increases up, and there is a minimum value at the data point $\gamma_n = 0.025$. Nevertheless, there is no obvious changing trend in the safety level of the system with the increase of the risk threshold γ_n.

As shown in Fig. 3, with the increase of risk threshold γ_s, the average waiting time keeps decreasing, and it tends to be gentle after the point $\gamma_s = 0.24$. With the increase of risk threshold γ_s, the safety level of the system is unchanged before the point $\gamma_s = 0.60$. After that data point, it gradually becomes smaller, and finally stabilizes.

In Fig. 4, we find that there is no obvious changing trend in the average waiting time with the increase of the queue threshold Q_n. The minimum waiting time is

Table 2 The number of trains and passenger flow at various time periods in Changsha South High-Speed Station

Time period	Number of HSR trains	Passenger flow
07:00–09:00	9	6720
09:00–11:00	16	9228
11:00–13:00	25	13,836
13:00–15:00	21	12,276
15:00–17:00	24	14,760
17:00–19:00	22	12,996
19:00–21:00	17	9120
21:00–23:00	4	840

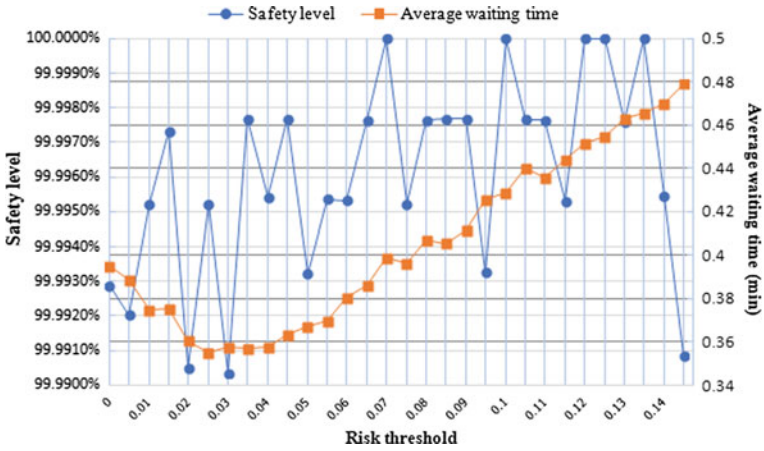

Fig. 2 Risk threshold γ_n versus two performance measures

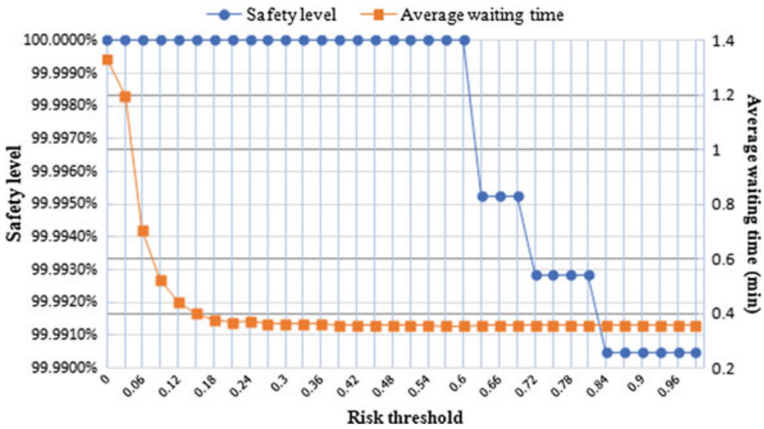

Fig. 3 Risk threshold γ_s versus two performance measures

obtained when we take $Q_n = 3$. The safety level of the studied system is not notably affected by the change of the queue threshold Q_n. The safety level has no significant changes while we take $1 \le Q_n \le 12$, and it tends to be flat and stable around a value after we take $Q_n \ge 12$.

In Fig. 5, we observe that the average waiting time first decreases slightly and then continues to increase when increasing the queue threshold Q_s. Besides, we obtain the minimal value of average waiting time as the queue threshold $Q_s = 3$. However, as the queue threshold Q_s increases, there is no obvious changes in the safety level.

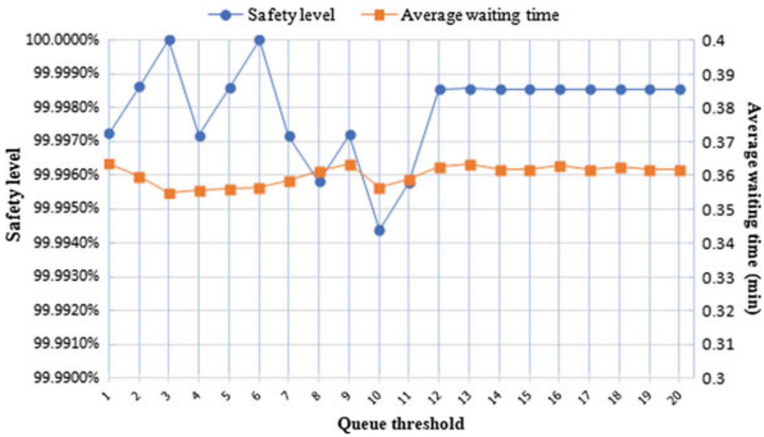

Fig. 4 Queue threshold Q_n versus two performance measures

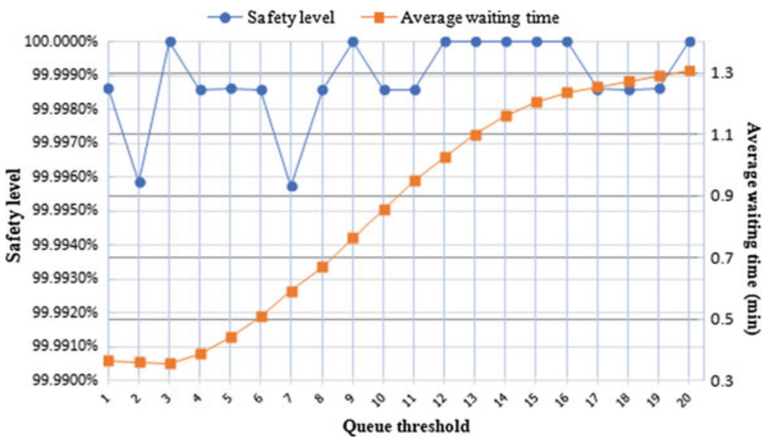

Fig. 5 Queue threshold Q_s versus two performance measures

We can observe from Fig. 6 that as the sampling probability P_n increases, there is no obvious changing rule for the average waiting time of passengers. When we set $P_n = 20\%$, it gives a minimum of the average waiting time. In addition, there is no significant change in the safety level as the sampling probability increases.

As illustrated in Fig. 7, it shows a downward trend in the average waiting time of passengers, and finally tends to be flat as the sampling probability P_s increases. At a sampling probability of 45%, there is a minimum waiting time for passengers. Nevertheless, when we vary the value of sampling probability P_s, there is no significant change in the safety level.

It can be observed in Fig. 8 that with the increases of the finite capacity M of Normal inspection channel, it illustrates a downward trend in the average waiting

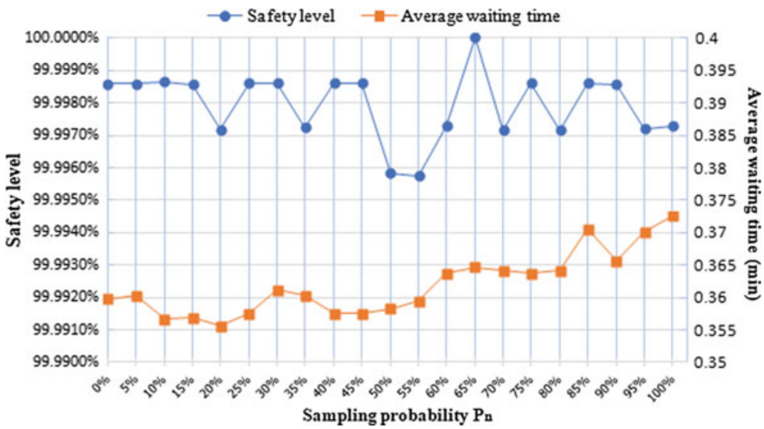

Fig. 6 Sampling probability P_n versus two performance measures

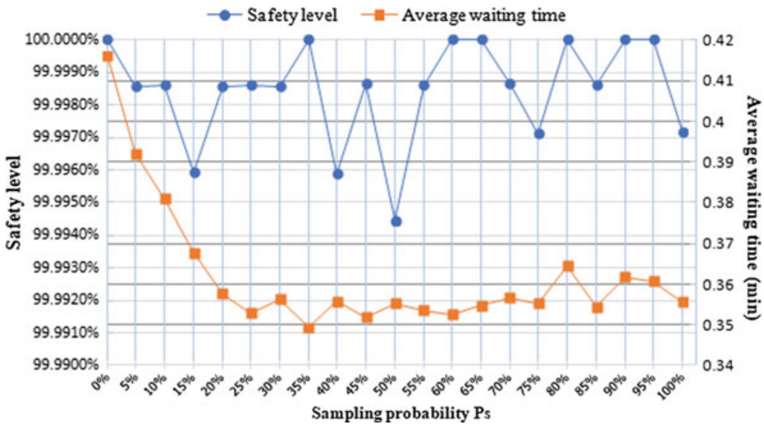

Fig. 7 Sampling probability P_s versus two performance measures

time of passengers and tends to be gentle. Besides, when the finite capacity is set as $M = 9$, we have the smallest value of these average waiting time of passengers. It can also be observed in Fig. 8 that there is no obvious change in the safety level when we vary the finite capacity of Normal inspection channel.

From Fig. 2, 3, 4, 5, 6, 7, and 8, we have studied the service performance of a security-check queuing system with data from a high-speed rail station in China. In the numerical experiments, we have discussed the influences of seven decision variables on the interested performance indicators for the presented security-check system with four types of inspection channels. Note that not all seven decision variables could have a significant impact on the average waiting time. There is a clear trend in the influence of several decision variables, such as risk threshold γ_n, risk threshold γ_s, queue threshold Q_s, sampling probability P_s, and finite capacity M.

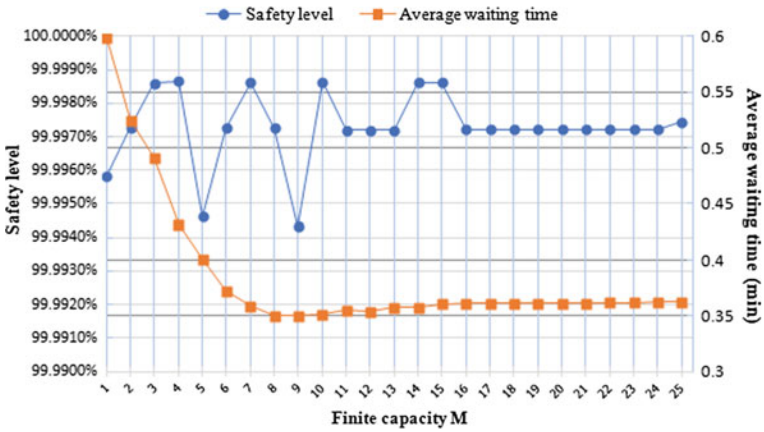

Fig. 8 Finite capacity M versus two performance measures

On the other hand, only one decision variable can have a significant impact on the safety level, i.e., risk threshold γ_S. The other six decision variables have no significant impact on the safety level.

5 Conclusions

In this paper, we studied the service performance of security-check queuing system for high-speed rail stations in China. A simulation model was developed for studying the security-check process via using Arena simulation platform. With data of a train station, we investigated the stochastic behavior of the presented queuing model through a series of computer simulations, where the numerical results are discussed and managerial implications are obtained. A sensitivity analysis was also conducted to understand the relationship between the decision variables and system performance. From the presented data analysis, it illustrates that there are several significant trends in the change of system performance. The data analysis in this work could provide effective and practical guidance for not only service management but also security issues of high-speed rail stations in China.

Acknowledgements The authors are grateful to the anonymous referees for their constructive comments, which led to the improvement of this paper in numerous ways. This study was supported in part by Fujian Provincial Department of Science and Technology under Grant No. 2016J01330, and the Research Fund from the Fujian University of Technology under Grant No. GY-Z18148.

References

1. C.-H. Wang, A queueing analysis of a security-check system with two types of inspection channels, in *Proceedings of 2018 International Conference on Mathematics, Modelling, Simulation and Algorithms (MMSA2018), Advances in Intelligent Systems Research*, vol. 159, Chengdu, China (2018), pp. 102–106
2. The National Development and Reform Commission (NDRC) of the People's Republic of China, *Mid-to-Long-Term Railway Network Plan* (2016 Revision No. 1536), issued on July 13, 2016, http://www.ndrc.gov.cn/zcfb/zcfbtz/201607/t20160720_811696.html
3. The National People's Congress of the People's Republic of China, *The Anti-Terrorism Act of People's Republic of China*, issued on December 27, 2015, http://www.npc.gov.cn/npc/xinwen/2015-12/28/content_1957401.htm
4. Z.G. Zhang, H. Luh, C.-H. Wang, Modeling security-check queue. Manag. Sci. **57**, 1979–1995 (2011)
5. X. Nie, G. Parab, R. Batta, L. Lin, Simulation-based selectee lane queuing design for passenger checkpoint screening. Eur. J. Oper. Res. **219**, 146–155 (2012)
6. C.-H. Wang, J. Lan, Data analysis and optimization strategy for a risk-based three-level inspection channel. Mod. Manag. **9**(1), 9–23 (2019)
7. F. Kaakai, S. Hayat, A.E. Moudni, A hybrid petri nets-based simulation model for evaluating the design of railway transit stations. Simul. Model. Pract. Theory **15**, 935–969 (2007)
8. C.-H. Wang, M.-E. Wu, C.-M. Chen, Inspection risk and delay for screening cargo containers at security checkpoints, in *Proceedings of the 11th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP-2015)*, Adelaide, Australia (2015), pp. 211–214
9. C.-H. Wang, A modelling framework for managing risk-based checkpoint screening systems with two-type inspection queues, in *Proceedings of the 3rd International Conference on Robot, Vision and Signal Processing (RVSP 2015)*, Kaohsiung, Taiwan (2015), pp. 220–223
10. P. Huang, H. Luh, Z.G. Zhang, A queueing model for tiered inspection lines in airports. Int. J. Inf. Manag. Sci. **27**, 147–177 (2016)
11. C.-H. Wang, Arena simulation for aviation passenger security-check systems, in *Proceedings of 10th International Conference on Genetic and Evolutionary Computing (ICGEC2016), Advances in Intelligent Systems and Computing*, vol. 536, Fuzhou, China (2016), pp. 95–102
12. C.-H. Wang, R. Chen, A simulation analysis of airport security-check queues based on passengers' risk classification. Manag. Sci. Eng. **7**(2), 110–124 (2018)
13. C.-H. Wang, A review of operational mechanisms and simulations for security screening systems. Comput. Sci. Appl. **7**(11), 1067–1078 (2017)
14. C.-H. Wang, J. Lan, Performance evaluation of a risk-based three-tier inspection system, in *Proceedings of 2nd International Conference on Computational Modeling, Simulation and Applied Mathematics (CMSAM2017)*, DEStech Transactions on Computer Science and Engineering, Beijing, China (2017), pp. 464–468
15. E. Matsika, C. O'Neill, U. Battista, M. Khosravi, A. de Santiago Laporte, E. Munoz, Development of risk assessment specifications for analysing terrorist attacks vulnerability on metro and light rail systems. Transp. Res. Procedia **14**, 1345–1354 (2016)

LSTM-Based Neural Network Model for Semantic Search

Xiaoyu Guo, Jing Ma and Xiaofeng Li

Abstract To improve web search quality and serve a better search experience for users, it is important to capture semantic information from user query which contains user's intention in web search. Long Short-Term Memory (LSTM), a significant network in deep learning has made tremendous achievements in capturing semantic information and predicting the semantic relatedness of two sentences. In this study, considering the similarity between predicting the relatedness of sentence pair task and semantic search, we provide a novel channel to process semantic search task: see semantic search as an atypical predicting the relatedness of sentence pair task. Furthermore, we propose an LSTM-Based Neural Network Model which is suitable for predicting the semantic relatedness between user query and potential documents. The proposed LSTM-Based Neural Network Model is trained by Home Depot dataset. Results show that our model outperforms than other models.

Keywords LSTM · Deep learning · Semantic search · RNN

1 Introduction

Having a better understanding of user's intention is very important to improve search engine's quality and optimize the user experience. Traditional search engine is mainly based on matching keywords in documents with search queries. However, this technology cannot distinguish synonymous words from different sentences. Moreover, users have to consider a lot about how to organize query words in order to get the right information they want. This brings too much inconvenience to users. As a result, semantic search engine emerges in order to better serve users.

X. Guo · J. Ma (✉) · X. Li
College of Economics and Management, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangning District 211106, China
e-mail: majing5525@126.com

H. Yang et al. (eds.), *Smart Service Systems, Operations Management, and Analytics*, Springer Proceedings in Business and Economics,
https://doi.org/10.1007/978-3-030-30967-1_3

Recently, Long short-term memory (LSTM) networks have shown remarkable performance in several kinds of Natural Language Processing (NLP) tasks, including image captioning [1], machine translation [2] and semantic search [3]. LSTM networks have also been used to predict semantic relatedness score [4] in order to find the relevant questions from the existing questions and find the relevant answers to a new question for Community Question Answering forums. Inspired by this task, we propose an LSTM-Based Neural Network Model reformed from predicting semantic relatedness score model based on similarities between predicting the relatedness of sentence pair and semantic search. We perform experiments to validate the utility of our model in Home Depot dataset, and compare the method to other models.

The contributions of this paper are as follows:

(1) Explore a novel method to process semantic search task. In this field, user query contains less information than search result does, so this paper proposed to split search result document and splice user query and related documents many times to balance the information quantity.
(2) Based on the method mentioned in (1), this paper built a novel neural network model for semantic search field.
(3) Perform experiments to validate the utility of our model, and compare our model to other model using different variants of LSTM architecture.

2 Related Work

The traditional language model, without taking sequence factor into account, cannot capture semantic information thoroughly. For example, "look after" and "after looking" have different meaning and traditional language model cannot tell any differences from each other. Mikolov et al. [5] proposed Recurrent Neural Network Language Model (RNNLM) in order to process sequence data. Recurrent neural network (RNN), which is different from normal neural network, introduces constant circulation into its model so that it could process sequence information. RNN shows remarkable performance in processing many tasks concerning sequence, but RNN has a Long-Term Dependencies problem [6] when it processes longer passages that contain too much information. The LSTM architecture, proposed by Hochreiter and Schmidhuber [7], addresses this problem of Long-Term Dependencies by introducing a memory cell that is able to store state information over long period of time into RNN structure. LSTM has recently been used in information retrieval field for extracting sentence-level semantic vectors [8] and context-aware query suggestion [9].

Commonly used variants of LSTM architecture are the Bidirectional Long Short-Term Memory (BLSTM) Network and the Multilayer Long Short-Term Memory (MLSTM). Several scholars pay much attention to exploring novel LSTM variants, including Tree-Structured Long Short-Term Memory Networks proposed by

Tai et al. [10] and multiplicative LSTM proposed by Stephen Merity et al. [11] in order to obtain a better performance on NLP tasks. In this paper, we conducted our experiments using different variants of LSTM.

The setup of our work is closely related to what was proposed already by Nassif H et al. [4]. Nassif et al. aim to obtain the semantic relatedness score between two questions as shown in Fig. 1. The model was built on two bidirectional LSTM whose output can be augmented with extra features and fed into the multilayer perceptron. Other similar methods include the combination of recurrent and convolutional models [12]. In this paper, we regard semantic search as predicting semantic relatedness between user query and potential document aiming to find the closest result in semantic meaning. The main difference of our method compared to these models is that we balance the quantity of information between inputs by exploiting user query several times. We also compare our method to these methods.

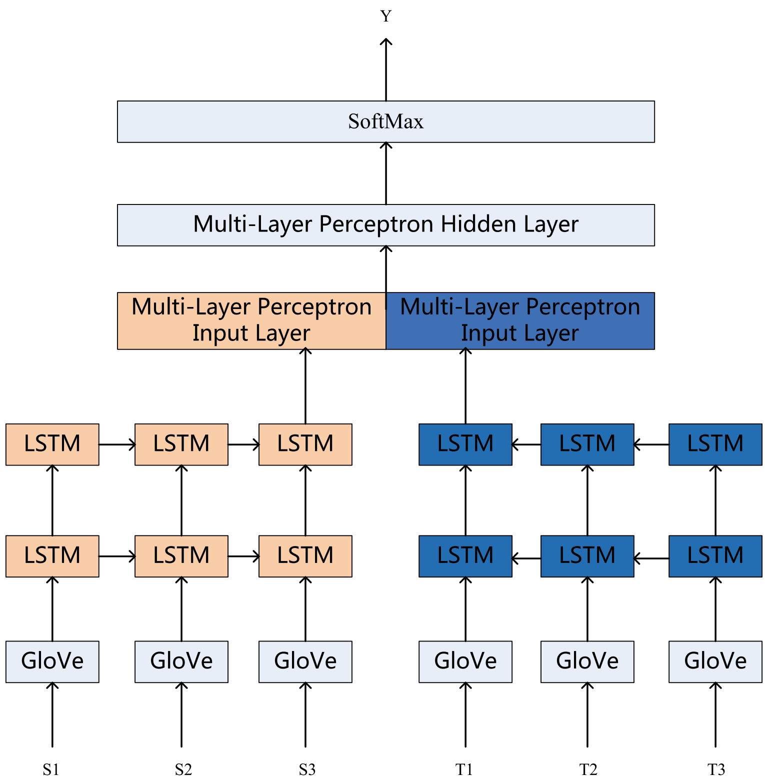

Fig. 1 The general architecture of predicting semantic similarity score model

3 Method

The essence of search engine is to calculate the relatedness score between user query and possible documents. Hence, predicting the semantic similarity bears strong resemblances to semantic search. We could see a semantic search task as described below:

$$f(Query, Doc) = SemanticScore\{a|a \in R, a \in [1, 3]\} \tag{1}$$

where Query is user input query words, Doc is a document and SemanticScore is a real number between 1 and 3.

That is to say, given user query and documents, processed by functional transformation and then output a real number between 1 and 3, in which 1 denotes "not relevant" and 3 denotes "extremely relevant".

Based on the model mentioned in 2, we proposed a LSTM-Based Neural Network Model which is suitable for semantic search, as is shown in Fig. 3. If we directly apply the model mentioned in 2 into semantic search, we would see a model shown in Fig. 2. Apparently, there exists an imbalance in information quantity between user queries and documents. That is to say, documents contain more information than user queries do. Therefore, if we directly use model in Fig. 2, we could not gain remarkable performance theoretically.

To balance the information quantity, we reform the model in Fig. 2 and the model in Fig. 3 is the general architecture of our model. There are three differences between Figs. 2 and 3. (1) We splice user query and related documents many times. The input layer of our model consists of two parts. One is LSTM representation of user query and another is LSTM representation of sentences in related documents. In this way, we make the best use of user query so that we balance the information quantity. (2) For every pair of user query and a document, we gain several semantic scores. (3) We use Ridge Regression as the last step to output the final semantic relatedness score.

The input of our model consists of two parts: one is user query words (QW) and another is Related Document. For QW, the model directly transforms them to vectors using GloVe and then encodes vectors using LSTM. For Related Document, the model splits the documents into sentences first and then processes them using the same methods as QW. After LSTM encoding, the model splice user query to every sentence twice. Then we get several semantic scores and our goal is to gain a final semantic score. Therefore, we see this task as a regression problem:

$$\begin{aligned} \widehat{\mathrm{S}} &= Xw \\ &= X(X^T X + \lambda E)^{-1} X^T S \end{aligned}$$

where S denotes actual semantic scores, $\widehat{\mathrm{S}}$ denotes the final forecasting semantic score, and X denotes a matrix consisted of sub-semantic scores.

In the last step, this model uses Ridge Regression to predict the final score. The output of our model is a final semantic score representing the semantic relatedness

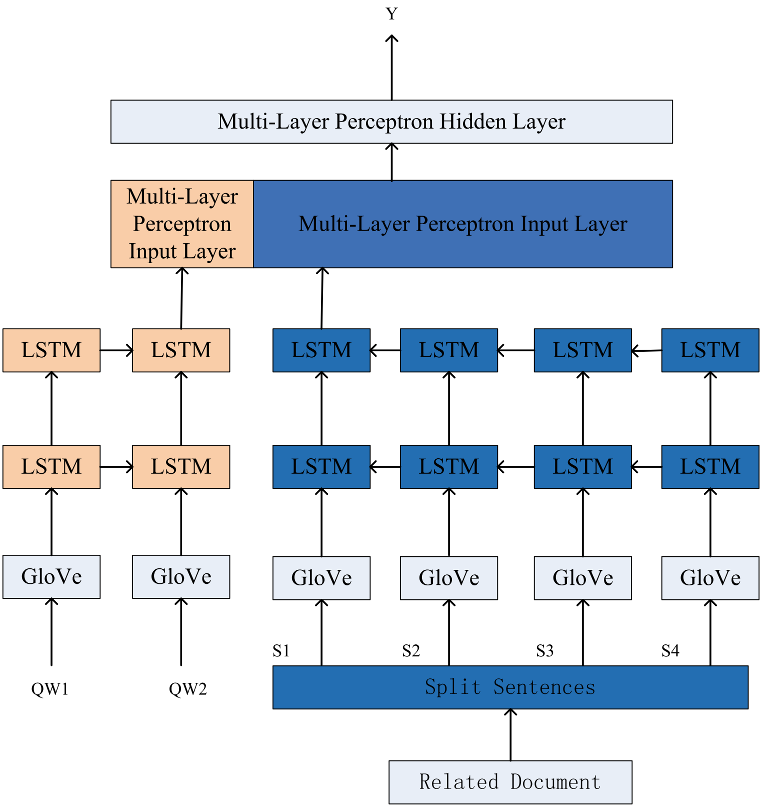

Fig. 2 Directly apply predicting semantic score model into semantic search

between user query and document. This model is designed to process English semantic search task. If you want to apply this model to other languages, all you have to do is to change the input part of the model. Take Chinese into example, we can split words first and then uses this model.

4 Experimental Setup and Results

In this section, we describe our experimental setup. The dataset and Evaluation methodology is described in Sects. 4.1 and 4.2. The experiments that we conduct to test the performance of our model is given in Sect.4.3.