**PAPER • OPEN ACCESS**

# A Hybrid Intelligent Directional Push Model for Service Platform Based on Deep Learning

To cite this article: Yong Shuai *et al* 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **569** 052076

# A Hybrid Intelligent Directional Push Model for Service Platform Based on Deep Learning

**Yong Shuai[1,2], Tailiang Song[3], Yong Wang[1,2], Qing Xia[1,2], Xinyi Su[4]**

[1]Chongqing Ceprei Industrial Technology Research Institute, Chongqing 401332, China

[2]Chongqing Engineering Research Center of Electronic Information Products Reliability, Chongqing 401332, China

[3]Institute of Optoelectronics, Beijing Institute of Technology, Beijing 100081, China

[4]Faculty of Information Technology, Monash University, Melbourne 3800, Australia

alexshuai@sina.com, songtl123@126.com, wangyong@ceprei.com, davidxia25@sina.cn, 12184135@qq.com

**Abstract**. In order to improve the push precision and user satisfaction of the service platform, this paper proposes an intelligent directional push model based on the existing problems of the intelligent search model and the actual characteristics of the service platform. Firstly, use data preparation method to finish collecting, integrating, cleaning, conversion and protocol of the user association data in the service platform. Then deep learning model is used to build a local service platform semantic library. Thirdly, use jieba algorithm to match the user-associated data with the local service platform semantic library, set the weights of input data based on its importance, use the normalization algorithm to obtain the access matching matrix of the target users. Finally the collaborative filtering algorithm is used to calculate the user matching degree. The case analysis proves that the model in this paper has higher accuracy.

## 1. Introduction

Recently, many customized developed smart technology service platforms are emerging, and these platforms will provide customized services for the potential needs of customers. However, if a general retrieval model is used, not only the retrieval speed is slow, but also the matching degree is not high, the potential demand of the user can not be fully exploited, and the service push and conversion rate are low. Therefore, this paper uses the directional intelligent retrieval model to realize the intelligent push of technology services.

The directional intelligent push model has two data sources, called associated data source and target data source. The characteristics of these two data sources can be summarized into four aspects:

- The associated data source is complex and the amount of data is large, and the data preparation process needs to be evaluated for importance.
- The target data source has a small amount of data and is not suitable for a large amount of training.
- Different from the non-directionality or weak orientation of commonly used search platforms, the content of directional push is highly correlated.
- The accuracy of the content in directional push model need to be higher.

Based on the above characteristics, this paper analyzes the current research contents of intelligent retrieval, which mainly includes the three aspects. The first one is to extract the search keyword phrases to find the content that needs to be searched, such as decision tree[1,14], breadth-first algorithm[5], domain-first extension[6], text sorting method based on cross information entropy and word feature information[8], cloud computing technology[10], multi-keyword fuzzy search[13]; the second one is to use the ontological method to identify the search keywords, including identify the search content and using the identified contents to search service[2], the integrated method for retrieval[3], two-order Clustering Algorithm Based on SOM Neural Network and Hierarchical Clustering Generates Ontology[4], data fusion technology[9], intelligent semantic retrieval[7], service hybrid selection strategy based on semantic matching and quality of service attributes[12], fuzzy logic retrieval[14], ontology-based big data intelligent retrieval[15]; the third one is to study the weight allocation strategy, such as the weight division strategy based on adaptive alternating particle swarm differential evolution optimization algorithm[11].

Although the above three methods solve some of the retrieval problems, there are still three shortcomings. First of all, only the search contents input by the user are considered, and the relationship between the user's previous access content and the actual demand content is not considered. Secondly, these papers do not take into account the impact of the services provided by the platform on search results; thirdly, there is no corresponding search semantic library, or the constructed semantic library is subjective and not precise enough.

Based on the above reasons, this paper proposes a hybrid intelligent directional push model of the service platform, fully considering the user's search habits and data characteristics before access, combined with the service content provided by the local platform, uses deep learning, jieba algorithm, cosine similarity to a build a intelligent directional push model of various data sources, achieves accurate push of the service platform and improve the push accuracy of the service platform and user satisfaction.

## 2. Modeling ideas

The main work to build the intelligent directional push model of the service platform contains: user-associated data preparation (including data collection, integration, cleaning, conversion and protocol), deep learning-based local service platform semantic library creation, user text word vector digitization(including word vector matching, access data word vector importance analysis and weight setting, all user data integration and getting target user access matching degree matrix) and target user access matching degree matrix acquisition and calculation. The detailed process is shown in Fig. 1.

## 3. Modeling process

Based on the modeling ideas provided in the previous section, the following contents describe the intelligent directional push model modeling process.

### *3.1. User Association Data Preparation*

Data preparation can improve the quality of text data, thereby improving the accuracy of the algorithm, and ultimately improving the push precision of the intelligent directional push model. Therefore, data preparation work for the acquired user-related data is firstly performed. Data preparation mainly includes data collection, data integration, data cleaning, data conversion and protocol.

#### *3.1.1. Associated Data Collection and Integration*

Linked data is collected from four locations, including web browsing data, cookie data, similar APP or software data and local service platform browsing data.

Among them, the web browsing data is mainly obtained by the crawling technology, the cookies data and the similar APP and the software are obtained by parsing the corresponding database, and the local service platform browsing data is recorded in real time when the user accesses the local platform system.

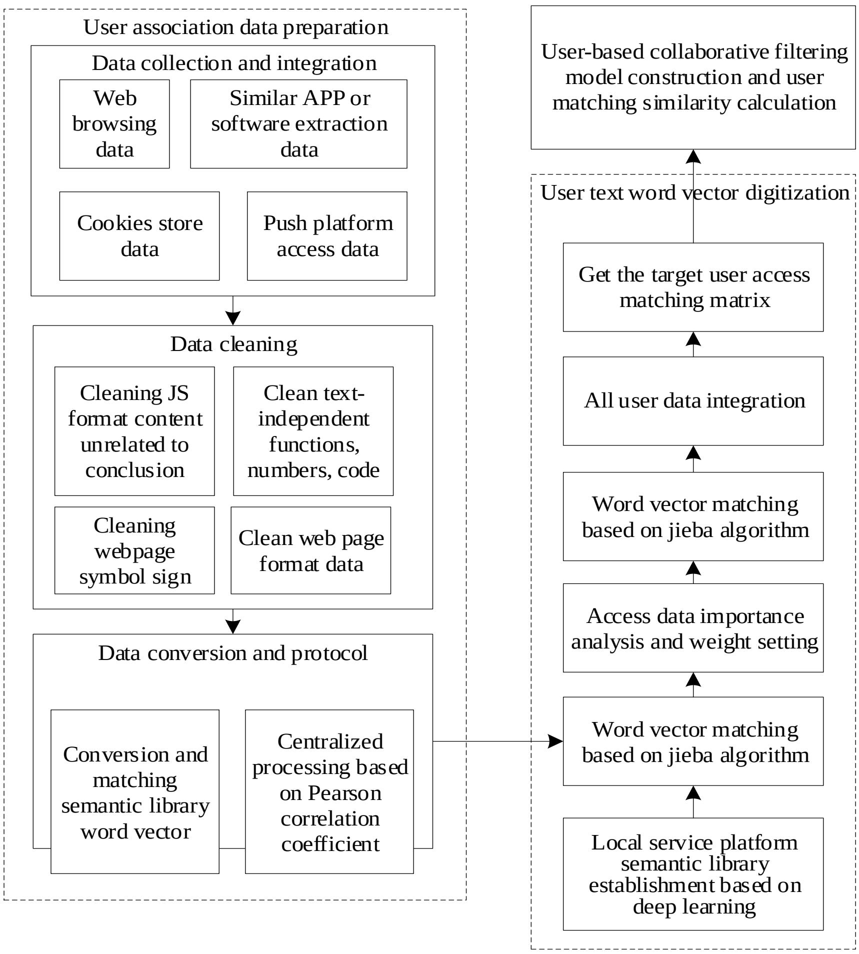


Fig. 1 Modeling process

At the same time, according to different data sources and related strengths, integrate and classify various types of data in consideration of their importance. For example, to a web page data, we need to integrate the four types of collected data , including title, keywords, and description in the head, and all the contents in the body.

### *3.1.2. Associated Data Cleaning*

The main work of data cleaning contains eliminating duplicate data, noise data and unrelated data, including:

- Clean JavaScript content that is not related to the conclusion, such as 'style', 'type', 'script', etc
- Clean web page symbol flags, such as separators, segmentation, operators, and terminators in web pages
- Clean web page format data such as '<div', '<a', '<ul'
- Cleaning text-independent functions, calculation methods, numbers, code, etc

### *3.1.3. Associated Data Conversion and Protocol*

The main work of data conversion and specification process is to convert the cleaned text data into word vectors and extract corresponding values based on the word frequency.

The steps of data conversion and protocol are as follows:

- Use jieba word segmentation mode to segment text data
- Remove stop words, where stop words use public stop words
- Extract keywords, set the number of keywords with the highest weight, and not specify part of speech

IOP Conf. Series: Materials Science and Engineering **569** (2019) 052076 doi:10.1088/1757-899X/569/5/052076

### *3.2. Deep learning based local service platform semantic library establishment*

Local semantic library establishing process based on deep learning and text mining by python is as follow[16]:

- Load TensorFlow and common data processing libraries, including numpy, scipy, pandas, sklearn
- Load pre-trained text, including service introduction text and service keyword semantic library text
- Define model readability functions to create a lookup table from integer to vocabulary list
- Set the service introduction text and service keyword word library text as input data and target data, and convert them to ids mode
- Set the hyper-parameters for deep learning, including learning rate, batch size, number of units, hidden units, embedding size, dropout value and iteration value
- All elements are superimposed to build a deep learning layer
- Build a function that retrieves the correct batch
- Start training deep learning models and save the training models
- Use a stored training model to capture key semantic libraries for different services

### *3.3. User text word vector digitization*

#### *3.3.1. Word vector match based on jieba algorithm*

To the data extracted by users from different data sources and different locations, the jieba algorithm is used to calculate the word vector matching degree with all the targeted service platform semantic libraries. The word vector matching degree is calculated as follows:

- Extract the tag value using the jieba thesaurus
- Merge labels
- Calculate the cosine similarity of the word vector by formula (1).

$$\text{cosine\_similarity} = \text{dot\_product} / ((\text{normA}^{0.5}) * (\text{normB}^{0.5})) \tag{1}$$

Where $\text{dot\_product} = \text{vector1} \times \text{vector2}$, $\text{normA} = \text{vector1} \times \text{vector1}$, $\text{normB} = \text{vector2} \times \text{vector2}$, and *Vector1* and *vector2* are two word vectors associated with each other.

#### *3.3.2. Access data importance analysis and weight setting*

To the collected associated text data, it needs to be converted into a standard word vector to match the semantic database of the local service platform, calculate the similarity and perform service push.

The importance analysis mainly considers three factors: the way to acquire data, the time difference between the access time and the current time, and the residence time per visit.

These data need to comply with the following assumptions:

**1. The importance of data acquisition methods**

Set the importance of data acquisition as Local Access Importance > Similar APP Importance > Cookie Storage Content Importance > Browsing History Importance

For ease of calculation, each importance values are defined as shown in formula (2).

$$w_1 = \begin{cases} 4, & \text{where information got from Local Platform} \\ 3, & \text{where information got from similar APP or Software} \\ 2, & \text{where information got from Cookies} \\ 1, & \text{where information got from Web History} \end{cases} \tag{2}$$

To the source data from web browsing record, because of the different structural distinctions, the words weights of different locations are defined as follows:

- The importance weights of the contents in Head and body are calculated and summed separately

IOP Conf. Series: Materials Science and Engineering **569** (2019) 052076 doi:10.1088/1757-899X/569/5/052076

- No matter how many times the word appears in the Head (or Body), the weight coefficient is calculated after integration
- When the value of the web page is used, the weight coefficient of the content in Header and Body are in accordance with the formula (3).

$$w_{14}=\begin{cases}0.4, & \text{where information got from Title in Head}\\ 0.3, & \text{where information got from Keywords in Head}\\ 0.2, & \text{where information got from Description in Head}\\ 0.1, & \text{where information got from Body}\end{cases} \tag{3}$$

The matching degree of the web page acquisition data is the sum of each position matching degree multiplies the weight value, as it is shown in formula (4).

$$S_4 = 0.4 * S_{TH} + 0.3 * S_{KH} + 0.2 * S_{DH} + 0.1 * S_B \tag{4}$$

Where $S_4$ is the matching degree of the data acquired by the web page, $S_{TH}$ is the matching degree of Title in Head, $S_{KH}$ is the matching degree of Keywords in Head, $S_{DH}$ is the matching degree of Description in Head, and $S_B$ is the matching degree of Body.

**2. Access time weight calculation method**

The access time means the time span from the time when some user visit the website to current time. It is calculated in hours, and its weight is as shown in formula (5).

$$w_2=\begin{cases}0, & \text{when interview time is greater than 360 hours}\\ (360-ivt/360), & \text{else}\end{cases} \tag{5}$$

Where *ivt* means interview time of the user.

**3. Method for calculating the weight of each visit stay time**

$$w_3=\begin{cases}0, & \text{when visit time is less than 1 minutes}\\ 1, & \text{when visit time is greater than 60 minutes}\\ (60-vt/60), & \text{else}\end{cases} \tag{6}$$

Where *vt* means visit time of the user.

Based on the above analysis, the importance weight of each access data is:

$$w = w_1 \times w_2 \times w_3 \tag{7}$$

The matching matrix of each access data is:

$$W_i = w * d_i \tag{8}$$

Where $W_i$ means the comprehensive matching degree of the i-th record, $d_i$ means the word vector matching of the i-th record

Therefore, the total access match for a single user is:

$$M = \sum_{n=1}^{N} W_i \tag{9}$$

### *3.3.3. Obtain the target user access matching matrix*

Using the above method, each user data is processed, and the data of each user is independently calculated and integrated, and all user service relevance matrix A is obtained, such as formula (10). The user actual service purchase data is collected, and the user service matching degree matrix is obtained as formula (11).

$$A=\begin{bmatrix}\text{ServiceCollection}_{11} & \cdots & \text{ServiceCollection}\\ \cdots & \cdots & \cdots\\ \text{ServiceCollection}_{M1} & \cdots & \text{ServiceCollection}_{MN}\end{bmatrix} \tag{10}$$

$$B = \begin{bmatrix} \text{ServicePurchases}_{11} & \cdots & \text{ServicePurchases}_{1N} \\ \cdots & \cdots & \cdots \\ \text{ServicePurchases}_{M1} & \cdots & \text{ServicePurchases}_{MN} \end{bmatrix} \quad (11)$$

In order to deal with the outliers and noise values in the matrix, such as a user over-reliance on the access of certain websites or the website access time is too long, the data in the matrix should be centralized so that all the data is located between [-1,1], the calculation method is:

$$U_i = \frac{U_i - \overline{U}}{Max(U)} \quad (12)$$

Where $U_i$ is the value of the i-th row in the matrix, $\overline{\mathrm{U}}$ is the average of the i-th row and *Max(U)* is the maximum value of the i-th row.

The two matrices are spliced to obtain the user comprehensive matching degree matrix C.

$$C = \begin{bmatrix} \text{User}_1 & \text{ServiceCollection}_{11} & \cdots & \text{ServiceCollection}_{1N} & \text{ServicePurchases}_{11} & \cdots & \text{ServicePurchases}_{1N} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \text{User}_M & \text{ServiceCollection}_{M1} & \cdots & \text{ServiceCollection}_{MN} & \text{ServicePurchases}_{M1} & \cdots & \text{ServicePurchases}_{MN} \end{bmatrix} \quad (13)$$

### *3.4. User matching calculation based on collaborative filtering*

Collaborative filtering is based on similarly similar evaluation algorithms, including user-based collaborative filtering and project-based collaboration. The advantages and disadvantages of the two collaborative filtering algorithms are shown in Table 1.

Table 1. Comparison of collaborative filtering algorithms

| Method | advantage | Shortcoming |
| --- | --- | --- |
| Collaborative-based collaborative filtering | There is no need to manually create features | Need a lot of project and user support algorithms<br>When the number of projects is much larger than the number that can be purchased, the utility matrix will be sparse |
| Content-based collaborative filtering | No need for a large number of users | Defining the right features can be a challenge<br>Conclusion is lack of interpretative |

Since the service platform usually has a specific service object, there is no need to manually create features, at the same time, the service platform has a large number of users and needs to satisfy the sparseness of the utility matrix. Therefore, user-based collaborative filtering can obtain good push effects. Take python as an example, the modeling steps are as follows:

- Import the necessary functions or libraries, such as pandas, numpy, sklearn, consine_similarity, etc.
- Create a DataFrame object
- Create a function that reads the user and item scoring matrix and returns a predictive score based on collaborative filtering
- Calculate the similarity between users by using the cosine similarity algorithm
- Calculate the estimated score of a given user for some service, and push the service with a high estimated score to the current user.

The cosine similarity calculation formula is:

$$S = \cos(\theta) = \frac{A \bullet B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (\mathrm{A_i})^2 \times \sum_{i=1}^{n} (\mathrm{B_i})^2}} \quad (14)$$

Where *A* and *B* represent the values of the components in the matrix that need to calculate the similarity.

Since a certain vector in the matrix is compared with other vectors and a centralized cosine similarity is generated, the value of the similarity is between [-1, 1], so after centrifugation, the cosine similarity between vectors and the Person correlation coefficient are equivalent, as shown in formula (15).

$$r_{xy} = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{\left(\sum X^2 - \frac{(\sum X)^2}{N}\right)\left(\sum Y^2 - \frac{(\sum Y)^2}{N}\right)}} \tag{15}$$

Where *X* and *Y* represent matrices whose similarity needs to be calculated.

The estimated score for some service is calculated as:

$$E_{n+1} = \sum_{i}^{n} V_i * S_i \tag{16}$$

Where $V_i$ is the evaluation value of the i-th user, and $S_i$ is the cosine similarity calculation value of the i-th user.

## 4. Case calculation and analysis

### *4.1. Case Introduction*

The case data of this paper gets from the project 'electronic product quality assurance comprehensive technology service sub-platform', which is oriented to the service demand of product quality improvement and reliability guarantee of Chongqing electronic information industry, and carries out eight types of technical services, such as system certification, product testing and certification, software evaluation, physical and chemical testing, failure analysis, information security, reliability data, metrology inspection and calibration. The relevant data has been saved to the database and processed according to the modeling process. The following is the intelligent directional push modeling process according to the above section.

### *4.2. Case calculation*

To facilitate case analysis, this paper extracts a typical user X from the sample database to demonstrate the modeling process. Firstly, analyze the data based on visited web page record.

Taking 'software evaluation' as an example, the semantic library size is defined as 5, and the semantic library obtained through the deep learning model is 'software, evaluation, process evaluation, software testing, software function'.

Firstly, analyze a single record, taking web browsing records and local platform access records as an example. For a single browsing record, such as 'http://www.cstc.org.cn/', calculate the word vector matching degree with the software testing comment library, and integrate the matching degree of the entire browsing record . The result is shown in Table 2.

Table 2. Result obtain from website

| Content | | | Single Result | Weights | Total Result |
|---|---|---|---|---|---|
| Head | Title | China Software Testing Center | 30.08 | 0.4 | 12.032 |
| | Key words | China Software Testing Center, Software Evaluation, Software Testing, Testing, Third Party Testing, Software Testing Center, Software Evaluation Center | 58.35 | 0.3 | 17.505 |

| | | | | |
|---|---|---|---|---|
| Description | China Software Testing Center (referred to as: China Evaluation), as the authoritative third-party software and hardware products and information system engineering quality safety and reliability testing institutions, is a kind of scientific research institution directly under the Ministry of Industry and Information Technology. Its business network covers more than 500 large and medium-sized cities across the country, and test reports issued in 61 countries and regions have achieved mutual recognition. | 8.61 | 0.2 | 1.722 |
| Body | IT system evaluation, ....... , intelligent transportation system evaluation service | 52.93 | 0.1 | 5.293 |
| | Total | | | 36.552 |

As it is shown in the above Table 2, the match for a single URL browsing record is calculated to be 36.552. According to this method, all the URL browsing records of this user are calculated. Calculate the access time weight and the weight of each visit stay time and summarize the URL access matching degree, as shown in the following Table 3.

Table 3. Match degree of all URL browsing records

| ID | website | Matching degree | Access time weight | weight of each visit stay time | Total degree |
|---|---|---|---|---|---|
| 1 | http://www.miitbeian.gov.cn | 36.552 | 0.45 | 0.65 | 10.69 |
| ... | ... | ... | ... | ... | ... |
| 224 | http://www.kekaoxing.com/ | 24.665 | 0.43 | 0.77 | 8.17 |
| | | | | Total Match Degree | 2287.97 |

Similarly, calculate the values of cookies, similar APPs and local platform access records, and summarize them as shown in Table 4.

Table 4. Match degree calculated from all data source

| Data source | Software detection match | Data acquisition method importance weight | Total Match Degree |
|---|---|---|---|
| Browsing history | 2287.97 | 1 | 2287.97 |
| Cookies | 1899.01 | 2 | 3798.02 |
| Similar App or Software | 3211.23 | 3 | 9633.69 |
| Local Platform | 1733.91 | 4 | 6935.64 |
| | Total Match Degree of software evaluation | | 22655.32 |

Therefore, the Match Degree of User X to software evaluation is 22655.32. Use the same method to calculate other seven match degree of Product testing certification, physical and chemical testing, failure analysis, information security, reliability data, metrology inspection and system calibration,the result is in Table 5.

Table 5. Match Degree of User X

| Service Items | X | Normalization |
|---|---|---|
| Software evaluation | 22655.32 | 0.61 |
| Product testing certification | 1244.56 | -0.33 |
| Physical and chemical testing | 13467.2 | 0.20 |
| Failure analysis | 886.52 | -0.35 |
| information security | 14562.44 | 0.25 |
| Reliability data | 980.78 | -0.35 |
| Metrology inspection and calibration | 8972.22 | 0.01 |
| System Certification | 7886.89 | -0.04 |

Calculate all current users' matching degree of service type and combine them with the actual number of purchases to generate a matching degree matrix. For the sake of display, only five users with similar similarities are listed here, as shown in Table 6.

Table 6. Match Degree of example users

| Data source \ Username | | User 1 | User 2 | User 3 | User 4 | User 5 | User X |
|---|---|---|---|---|---|---|---|
| Similar Data Match Degree | Software evaluation | 0.61 | -0.18 | 0.34 | -0.30 | 0.47 | 0.34 |
| | Product testing certification | -0.33 | -0.18 | -0.17 | -0.25 | 0.05 | -0.17 |
| | Physical and chemical testing | 0.20 | 0.15 | -0.20 | 0.45 | -0.25 | -0.20 |
| | Failure analysis | -0.35 | -0.11 | 0.46 | -0.05 | -0.03 | 0.46 |
| | information security | 0.25 | -0.19 | -0.10 | -0.06 | 0.27 | -0.10 |
| | Reliability data | -0.35 | 0.80 | 0.43 | -0.07 | -0.42 | 0.43 |
| | Metrology inspection and calibration | 0.01 | -0.11 | -0.39 | 0.20 | -0.39 | -0.39 |
| | System Certification | -0.04 | -0.17 | -0.35 | 0.08 | 0.31 | -0.35 |
| Actual purchase service data | Software evaluation | 0 | -0.09 | -0.15 | 0.25 | -0.43 | |
| | Product testing certification | -0.29 | 0.13 | 0.48 | 0.36 | -0.07 | |
| | Physical and chemical testing | -0.43 | 0 | 0.03 | 0.19 | -0.14 | |
| | Failure analysis | 0.29 | -0.35 | -0.22 | -0.44 | 0.52 | |
| | information security | -0.14 | 0.46 | -0.27 | -0.13 | -0.39 | |
| | Reliability data | 0.14 | -0.06 | 0.35 | -0.43 | 0.34 | |
| | Metrology inspection and calibration | 0.43 | 0 | 0.30 | -0.05 | -0.26 | |
| | System Certification | 0 | -0.09 | -0.51 | 0.25 | 0.45 | |

Calculate the similarity of UserX to the current user by using cosine similarity.

Table 7. Similarity of UserX to the current user

| Cosine similarity | User1 | User2 | User3 | User4 | User5 |
|---|---|---|---|---|---|
| UserX | -0.121 | 0.421 | 1.0 | -0.506 | 0.028 |

Calculate UserX's trend of service push based on cosine similarity and compare them with the actual purchase service. The relevant data is shown in Table 7 and Fig. 2.

Table 8. Comparison of UserX's calculated trend and actual purchase service

| Service type | Push trend | Actual purchase service | Match degree | Error rate(%) |
|---|---|---|---|---|
| Software evaluation | -0.327 | 29 | -0.326 | 0.31 |
| Product testing certification | 0.386 | 144 | 0.289 | 25.13 |
| Physical and chemical testing | -0.018 | 79 | -0.058 | 68.97 |
| Failure analysis | -0.165 | 57 | -0.176 | 6.25 |
| information security | -0.004 | 83 | -0.037 | 89.19 |
| Reliability data | 0.535 | 187 | 0.519 | 2.99 |
| Metrology inspection and calibration | 0.266 | 140 | 0.268 | 0.75 |
| System Certification | -0.662 | 0 | -0.481 | 27.34 |
| Average error rate | | | | 27.61 |

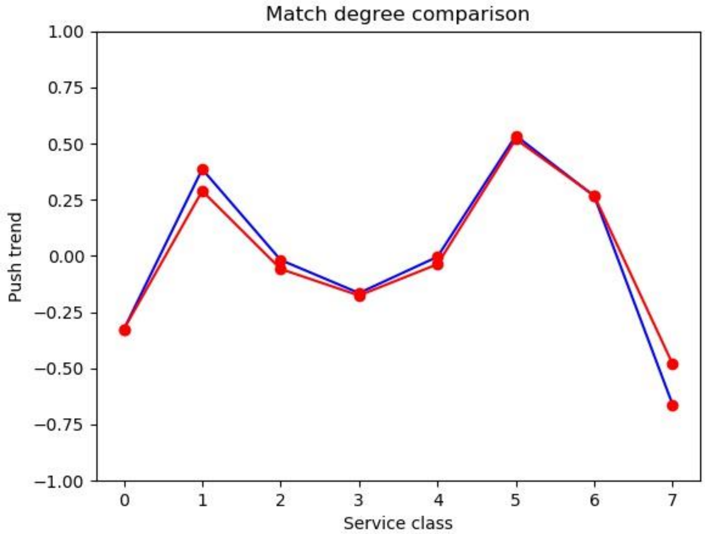


Fig. 2 Comparison of push trend

### 4.3. Case analysis

Based on the above calculations, it can be seen from *Table 9* that the most similar user to UserX is User3 and the related services of Reliability data and Product testing certification should be firstly pushed to to UserX.

From the figure 2, we can find that the matching degree calculated by the intelligent directional push model is very similar to the actual value, and the two types of data almost completely overlap.

In order to verify the accuracy of the model, some similar algorithms are used and the accuracy of the related algorithms are shown Table 9. It can be seen from the table that the intelligent directional push model used in the paper has higher accuracy and can meet the actual needs of users.

Table 9. Comparison of accuracy

| Method | Accuracy(%) | Sequence |
|---|---|---|
| Decision tree | 57.16 | ⑥ |
| Breadth-first algorithm | 62.39 | ⑤ |
| Cross information entropy | 67.83 | ② |
| Multi-keyword fuzzy search | 67.26 | ③ |
| Particle Swarm Differential Evolution Optimization Algorithm | 64.18 | ④ |
| This paper | 73.39 | ① |

## 5. Conclusions

In order to improve the push precision and speed of the service platform, this paper uses the user access association data and the user's actual purchase service data, combined with data preparation, deep learning, jieba algorithm, cosine similarity and other algorithms to construct an intelligent directional push model. By the help of case analysis, we prove that the model constructed by this paper has higher push accuracy.

At the same time, since the directional intelligent push platform needs to use deep learning and related intelligent algorithms, in order to save the push time, the operation of the directional intelligent push algorithm should be independent of the platform search operation. Since the platform contains a large number of users, the calculation of the user association matrix should be at the lowest time of platform access, such as midnight 0 to 2 o'clock.

**Acknowledgements**
This research was financially supported by the National Key Research and Development Program of China (2017YFB1401705), Intelligent Manufacturing Innovation and Development Project of the Ministry of Industry and Information Technology (MGY1704040, MGY1804080), Chongqing Science and Technology Bureau(cstc2018jszx-cyzdX0083, cstc2018jszx-cyzd0634, cstc2019jscx-fxyd0298).

**References**

[1] Liu Shuxun ,Ou Yishan, et al. The Imformation Query Mechanism of Intelligent Searching Agent in Internet[J]. Microelectronics and Computer,2000, 7(1):54-58(in Chinese).
[2] Li Faping. Design of intelligent retrieval system for ship cabin monitoring video[J]. SHIP SCIENCE AND TECHNOLOGY, 2018, 40(3):199-201(in Chinese).
[3] Chu Chunmei．Li Xiaoxia, et al. Directed query engine applications in the integrated retrieVal of chemical Web databases[J]. Computers and Applied Chemistry, 2005, 22(8):659-666(in Chinese).
[4] Gao Lin, XIA Qingguo, et al. Method ofbuilding intelligent information retrieval system based on ontology[J]. Computer Engineering and Design, 2008, 29(24):6309-6311(in Chinese).
[5] Ruan Yang, Liu Yu, et al. The Crawler-Based Directional Data Retrieval System[J]. COMPUTER ENGINEERING & SOFTWARE, 2018, 39(5):118-120(in Chinese).
[6] Wang peixia, Yu Hai, et al. Using Intelligent System to Extract Search Terms for Sci-Tech Novelty Retrieval[J]. Modern library information technology, 2016, 16(11):82-93(in Chinese).
[7] Yang Hongzhang, Fu jing. Method for constructing patent information intelligent semantic retrieval system by using patent text structured features[J]. Information Theory and Practice, 2015, 38(4):136-138(in Chinese).
[8] Zhang Ruifang, GUO Kehua. Novel retrieval intention modeling method for personalized website[J]. Computer Engineering and Applications, 2018, 54(6):37-43(in Chinese).
[9] Xia Jinsong. Data Fusion Method Based on Intelligent optimization Algorithm in Information Retrieval[J]. Software Guide, 2017,16(11):202-205(in Chinese).
[10] Zhang Qin, Zhang Huibin, et al. Design and realization of cloud inspection platform for manufacturing industry[J]. Computer Integrated Manufacturing Systems, 2017,23(8):1809-1816(in Chinese).
[11] He Heng, Xia Wei, et al. Fuzzy Multi-keyword Retrieval Scheme over Encrypted Data in Cloud Computing. COMPUTER SCIENCE, 2017,44(5):146-152(in Chinese).
[12] Hou Yongshuai，Zhang Yaoyun, et al. Recognition and Retrieval of Time-sensitive Question in Chinese QA System[J]. Journal of Computer research and development, 2013, 50(12):2012-2620(in Chinese).
[13] M.A.Rakhmatullaev1, B.B.Muminov. SYSTEM OF INTELLIGENCE RETRIEVAL FOR CORPORATE ACADEMICAL INFORMATION NETWORK[J]. International Journal of Research in Engineering and Technology,2016, 05:85-87.
[14] M. Vilares Ferro, M. Fern andez Gavilanes, et al. Intelligent Retrieval for Biodiversity[J]. International Journal on Articial Intelligence Tools, 2016,25(1):1-29.
[15] E Birialtsev, N Bukharaev, A Gusenkov. Intelligent search in Big Data[C]. Journal of Physics, 2017:1-9.
[16] Indra den Bakker. Python Deep Learning Cookbook: Over 75 parcitical recipes on neural network modeling, reinforcement learning, and transfer learning using Python[M]. China Machine Press, 2018.6:169-173