Realization of Knowledge Push System Based on Machine Learning

Yue Jiang, LiWei Wang, LeiJing Kang
R&D Center of China Academy of Launch Vehicle Technology, Beijing 100076, China
15210253626@163.com

Abstract. This paper implements the design of knowledge push system by the aid of machine learning, and details the key technologies including feature extraction of user data, method of constructing and updating user interest model, and the recommendation algorithm. By constructing the knowledge push model, knowledge resources can be pushed to users automatically and accurately, which can reduce the cost of knowledge retrieval and promote the knowledge sharing and reuse.

1. Introduction
China academy of launch vehicle technology has accumulated a lot of valuable knowledge resources since knowledge management work was conducted here in 2011. However, in general, the condition of knowledge sharing and reuse is not good and 70% of the knowledge is idle. The accumulated knowledge resources are difficult to play their proper value, which inevitably leads to low efficiency, long cycle and high cost in model developing and makes it difficult to adapt to the highly frequent development and launch of rockets. So it is necessary to realize intelligent knowledge mining and knowledge push by the aid of machine learning method, with the purpose to promote knowledge sharing and reuse.

Based on the characteristics of knowledge base for carrier rocket, this paper describes the mechanism of knowledge push based on user’s behavior. According to a user’s basic information, browsing history and operation behavior, related knowledge can be mined and pushed to the user.

2. Feature extraction of user data
User data mainly refer to the data containing the information about user’s interests, habits and preferences. Feature extraction of these data can prepare for the establishment of user interest model, which can be divided into two parts: the extraction of explicit data and the extraction of implicit data. Explicit data mainly include user basic information (e.g. job, department, technical ability, security classification, etc), current retrieval information and subscription information. Implicit data mainly include: history retrieval information, user knowledge behavior (e.g. browsing, rating, upload, download, bookmarking, etc.).

Implicit data extraction method is mainly used in this paper. In this way, user data can be obtained automatically by tracking a user’s browsing behavior without disturbing the user. On the client side, a user’s browsing behavior, such as bookmarking, printing, the speed of the scroll bar, etc, can be recorded by the aid of Ajax, JavaScript and other technologies. Meanwhile, the server can record the user’s IP, user ID, pages visited, access time and other data in the server log.
Here we use A to represent the user behavior set including uploading, downloading, bookmarking and other tagging activities. B represents the set of user’s rating behaviors, and C represents the set of user’s browsing behaviors. Feature extraction can be done according to the procedures shown in Figure 1 and Table 1, and the interest words of the user can be obtained.

3. Construction of user interest model

3.1. The representation of user interest model

The user interest model is one of the key parts of the knowledge push system, and it has apparent impact on the accuracy of knowledge pushed to the user. In fact, the user interest model is a kind of data structure, the pattern of its manifestation can directly determine the capability and computability of the model. The representation of vector space is widely used in the construction of user interest model [1-2].

The user interest model can be represent by a \( n \)-dimension vector \( \{ (t_1, w_1), (t_2, w_2), \ldots, (t_n, w_n) \} \), where \( t_i (1 \leq i \leq n) \) and \( w_i (1 \leq i \leq n) \) respectively represent the \( i \)th interest word and its weight value. The value of \( w_i \) indicates the level of interest in item \( t_i \) (hereinafter referred to as interestingness).

![Figure 1 Feature extraction logic procedures](image-url)
Table 1 Feature extraction data processing procedures

| Functions       | Descriptions |
|-----------------|--------------|
| **Input**       | User data    |
| **Process**     | For each knowledge document, distinguish the user behaviors of class A, B and C, and then calculate the scores respectively. If the total score of class A is more than the threshold, the knowledge document is marked as interest document of class A. Similarly, we can recognize the interest document of class B and class C. Based on the interest documents, the user’s interest words can be obtained by technologies such as text segmentation and feature extraction. |
| **Output**      | The user’s interest words |

3.2. The building process of user interest model

The building process of user interest model is shown in Figure 2 and Table 2.

![The construction of user interest model](image)

**Figure 2 The building process of user interest model**

Table 2 User interest data processing procedures

| Functions       | Descriptions |
|-----------------|--------------|
| **Input**       | User data    |
| Process         | 1) Read the user information, load the user instance and the corresponding interest branch items; 2) Through the information extraction of each branch item, the system automatically perceives the user's action, then intelligently handles the interest branch item, and saves the result; 3) The data is aggregated to form the user's interest model. |
| **Output**      | User Interest model |

4. Update of user interest model

The user instance in the user model is not fixed. On the contrary, the user instance changes along with the user's behavior. It is necessary to update the user instance every once in a while. Thus, in the construction process of user model, it needs to take full account of the changes of user instance to ensure the timeliness of user model.

4.1. Method to update user interest model

There are two ways to achieve the timeliness of the user interest model. One is the global numerical method. The database needs to be scanned and the data need to be recalculated before each knowledge push. This method can guarantee the timeliness of user interest model, but the calculation task is large and time-consuming, which will greatly affect the efficiency of the system. The other one is the method based on the model. The system just scans the database once, and calculates all the user
instances. The user instances can be directly called at each knowledge push with no recalculation. But this method will lead to a serious hysteresis effect, and needs an overall periodic update.

In this paper, we integrate the two methods above and use the advantages of both methods to implement the update of the user interest model. The specific procedures are shown in Figure 3.

![Figure 3 User interest model update procedures](image)

First, the model calculation is performed periodically. When periodical update time arrives, the periodical update module is triggered, then the system automatically detects and calculates the changed user instance. Second, the system monitors each user instance. If the interest update parameter in a user instance exceeds the threshold, the single-user update module is triggered and the user instance is recalculated. On the one hand, the user model can be updated periodically and entirely. On the other hand, the user instance can be fine-tuned according to the monitoring data. To some degree, the attributes of real-time and simplicity can be both satisfied through this method.

4.2. Design of update module

1) Periodical update module

When the periodical update time arrives, the system automatically detects the state of all the user's interest, and then put the changed user into the update set. The users with no interest changes do not enter the update set to reduce the repeated calculation. The system calculates the user instance for each user in the update set to form the new user interest model.

2) Single-user update module

Single-user update is more complicated than the periodical update (As shown in Figure 4). There are a variety of situations to trigger, including time trigger, quantity trigger and behavior trigger. The actual situation is often a combination of multiple triggers. Behavior trigger usually occurs along with quantity trigger, while time trigger and quantity trigger can work alone.
Figure 4 Single-user update procedures

① Time trigger
When the user has an interest word but has not involved it for more than a certain time, it is necessary to use the forgotten function to carry on the interestingness analysis for the interest word. If the user's interestingness is lower than the predetermined threshold, delete the interest point.

The user's interestingness of an interest word can be calculated using the formula of the degree of familiarity. The formula is based on the forgetting rules, where the memory is influenced by time and contents. So the calculation of the user's interestingness of an interest word can be converted into:

\[ y_c = \begin{cases} \frac{1.2 \cdot \sin(\varphi \cdot w_c)}{\sqrt{1 + x \cdot \delta}}, & 0 \leq x \leq \eta \\ \frac{\ln(x - \eta + 1)}{10}, & x \geq \eta \end{cases} \]

Where \( y_c \) is the user's interestingness of word \( c \), and its value decreases and finally stabilizes with increase of time. \( \varphi \) is the item used to adjust the rate of change with time. \( w_c \) presents the weight of word \( c \) in the interest branch item of user instance. \( \eta \) shows the length of time between initial point of knowledge being quickly forgotten and when the forgotten curve no longer changes, which is around 10 days. \( \delta \) is the parameters when the forgotten curve no longer changes, and \( \delta = 1.2 \cdot \sin(\varphi \cdot w_c)/\sqrt{\eta + 1} \).

② Quantity trigger
When the quantity of interest words exceeds the specified limit, which may be caused by some reasons such as behavioral triggering, the interestingness analysis for all the interest words on the
branch is carried out and the interest words with lower interestingness need to be removed to form a new interest word set (quantity requirement satisfied).

③ Behavior trigger

With longer time users spent on the system, the user interest words may change and the system will form new interest words through the behavior analysis according to the existing user behavior and update the old ones.

5. Recommendation Algorithm

Currently, popular personalized recommendation algorithms include the following: demographic-based recommendation, content-based recommendation, collaborative filtering based recommendation, etc. [3]. Content-based recommendation algorithm has been widely used in the text related fields, since its model is relatively easy to build and the personalized recommendation effect is good. Content-based recommendation algorithm uses the property features of the knowledge items and user's ratings of their choices over the items as the source data, and not requires other users' data. The approach makes recommendations only based on the user's own interests, so there is no user-item rating matrix sparse problem, thus can be used to do recommendation for the user who have a special interest. In the content-based recommendation system, the knowledge item is defined as a vector of feature representation, and the quality of these features is directly related to the quality of the recommendation system. For text-based content, features can be extracted automatically by means of information technology, and they can also be arbitrarily defined. The recommendation system learns the user's interest based on the characteristics of the items rated by the user and generates the user's interest document, and then finds the knowledge with the highest interest of the user by matching the interest document.

The procedures of knowledge recommendation are shown in Figure 5 and Table 3.

![Figure 5 Logic procedures of knowledge push](image-url)
### Table 3 Knowledge Push data processing procedures

| Functions | Descriptions |
|-----------|--------------|
| Input     | Current user ID |
|           | 1) Read the current user instance, and obtain the user's interest words through the user interest model; |
|           | 2) Sort the user's interest words by the weight, and get the weight list; |
| Process   | 3) According to the order of interest words, access the relevant knowledge documents; |
|           | 4) If the user has viewed the knowledge document, put key identification on the document; |
|           | 5) Push the knowledge list to the user's work surface. |
| Output    | The knowledge list quickly pushed |

#### 6. Application

Currently, machine learning based knowledge push model constructed in this paper has been used in the knowledge management system of China Academy of Launch Vehicle Technology, to provide users with personalized knowledge recommendations. By recording the operating behaviors of the users, and combining the professional background of the users and other related data, we categorize the knowledge resources of the knowledge base, and push the relevant knowledge documents to the users' personal spaces, thereby reducing the users' cost to retrieve the knowledge and improving the knowledge resource utilization, and effectively increasing the viscosity of the users. Knowledge push interface is shown in Figure 6.

![Figure 6 Knowledge push interface](image)

#### 7. Conclusion

Machine learning based knowledge push system implements user behavior based knowledge push. According to the user's basic information, browse records and other operation behaviors, the system can mine and push relevant knowledge to the users. This paper introduces some key technologies: feature extraction of user data, method of constructing and updating user interest model, and the recommendation algorithm, and explains the working principle of knowledge push. The main
characteristic of intelligent push system is that it can identify the behaviors of a user through the interaction with the user, so as to realize the personalized recommendation of knowledge. Apply the machine learning technique to the knowledge push system is the key to improve the intelligence of the system. By acquiring behavioral characteristics of the user, the proactive, accurate, and personalized knowledge push can provide better knowledge services, reduce costs of the users to retrieve information, improve work efficiency of the users, thus increasing the viscosity of the system's users.

References
[1] Li Z F, Feng X F, Zhao H Q. Research on Information Push System Based on Users Interest Model[J]. Journal of Taiyuan University of Technology, 2011, 42(5):503-509.
[2] Zou Z W, Ke Q. The Design and Optimization of an Active Pushing System Based on the Vector Space Model [J]. New Technology of Library and Information Service, 2005, 7:42-45.
[3] Tang R. Research on News Recommendation Based on Fusion of Content-Based Recommendation and Collaborative Filtering[D]. content-based recommendation and collaborative filtering recommendation fusion of news, Chongqing University of Technology, 2016.