Computers, Materials & Continua
DOI:10.32604/cmc.2022.022717
Article

Tech Science Press

Content Feature Extraction-based Hybrid Recommendation for Mobile Application Services

Chao Ma[1,*], YinggangSun[1], Zhenguo Yang[1], Hai Huang[1], Dongyang Zhan[2,3] and Jiaxing Qu[4]

[1]Harbin University of Science and Technology, Harbin, 150040, China
[2]Harbin Institute of Technology, Harbin, 150001, China
[3]The Ohio State University, Columbus, 43202, USA
[4]Heilongjiang Province Cyberspace Research Center, Harbin, 150001, China
*Corresponding Author: Chao Ma. Email: machao8396@163.com
Received: 16 August 2021; Accepted: 16 November 2021

Abstract: The number of mobile application services is showing an explosive growth trend, which makes it difficult for users to determine which ones are of interest. Especially, the new mobile application services are emerge continuously, most of them have not be rated when they need to be recommended to users. This is the typical problem of cold start in the field of collaborative filtering recommendation. This problem may makes it difficult for users to locate and acquire the services that they actually want, and the accuracy and novelty of service recommendations are also difficult to satisfy users. To solve this problem, a hybrid recommendation method for mobile application services based on content feature extraction is proposed in this paper. First, the proposed method in this paper extracts service content features through Natural Language Processing technologies such as word segmentation, part-of-speech tagging, and dependency parsing. It improves the accuracy of describing service attributes and the rationality of the method of calculating service similarity. Then, a language representation model called Bidirectional Encoder Representation from Transformers (BERT) is used to vectorize the content feature text, and an improved weighted word mover's distance algorithm based on Term Frequency-Inverse Document Frequency (TFIDF-WMD) is used to calculate the similarity of mobile application services. Finally, the recommendation process is completed by combining the item-based collaborative filtering recommendation algorithm. The experimental results show that by using the proposed hybrid recommendation method presented in this paper, the cold start problem is alleviated to a certain extent, and the accuracy of the recommendation result has been significantly improved.

Keywords: Service recommendation; cold start; feature extraction; natural language processing; word mover's distance

1 Introduction

As the mobile communication technology continues to grow, the coverage of intelligent mobile terminal devices continues to expand, and the number of mobile application services is showing an explosive growth trend, then the problem of information overload in the field of mobile application service comes with it [1]. The recommendation system emerged as an effective means to solve the problem of information overload. The current recommendation system of mobile application services includes both the design and research of user models, as well as the ranking and screening of mobile application services. On the one hand, the reasonable integration of the two can provide users with personalized information and improve the quality of services. On the other hand, it can make users rely on the system and reduce the loss of users.

Since the recommendation system was proposed, there are several recommended strategies recognized in this field, which mainly include the following: collaborative filtering, content-based recommendation, and social network-based recommendation. Among them, the collaborative filtering recommendation algorithm has a wide range of applications in the fields of the online retail service, online taxi-hailing service, the online travel service, and so on [2–4]. Collaborative filtering recommendation algorithm is one of the most widely used and most successful recommendation techniques in recommendation systems [5]. Collaborative filtering recommendations can be roughly divided into three types: User-based Collaborative Filtering, Item-based Collaborative Filtering Recommendation, and Model-based Collaborative Filtering Recommendation [6].

However, when a new item is not rated by any user, or when a new user has not rated any item, then the most classic cold start problem in the collaborative filtering algorithm will appear. The cold start problem refers to the general problem of a recommendation system that lacks a large amount of user behavior data and item data in the initial stage that cannot carry out effectively personalized recommendation [7]. In recent years, with the development of machine learning technology and more and more scholars' in-depth research, some results have been achieved in solving different types of cold start problems, but there is still a lack of perfect solutions [8]. The effect of solving the cold start problem of the recommended system will have a direct impact on the performance of the recommended system.

Aiming at this problem, this paper proposes a hybrid recommendation method based on content feature extraction. This method extracts service content features through natural language processing (NLP) technologies such as word segmentation, part-of-speech tagging, and dependency parsing. Then, the Bidirectional Encoder Representation from Transformers (BERT) model is used to vectorize the content feature text, and an improved weighted word mover's distance TFIDF-WMD algorithm based on Term Frequency-Inverse Document Frequency (TF-IDF) is used to calculate the similarity of mobile application services. Finally, the recommendation process is completed by combining the item-based collaborative filtering recommendation algorithm.

The main contributions of this paper are as follows: (1) Based on NLP technology, a specific method for extracting the potential content features of mobile application services is proposed. This method improves the accuracy of describing mobile application service attributes and the rationality of the method of calculating mobile application service similarity. (2) A hybrid recommendation algorithm that combines the content features of mobile application service and user ratings is proposed. This algorithm effectively alleviates the cold start problem of the collaborative filtering recommendation system.

The remainder of this paper is structured as follows. In Section 2, we discuss previous work related to the proposed approach. In Section 3, we propose a hybrid recommendation method for mobile

application services based on content feature extraction. In Section 4, we conduct experiments to verify the effectiveness of the proposed method. Finally, we present the conclusions to this study.

2 Related Works

In recent years, service recommendation has become a relatively active research direction in the field of service computing, and many scholars have done in-depth research on it. Cold start problems in the recommendation system can be divided into two types: cold start problems for new users [9] and cold start problems for new items [10]. The cold start problem of a new item is when a new item is added to the system, because it is a brand-new item that has not been rated by users, or because there are fewer users who have rated the item, the number of ratings is lower than a preset threshold, so the probability of this item being recommended is very small. Many scholars have put forward many solutions to this classic problem.

Faced with a large number of unrated new items, users often need to spend a lot of time to choose their favorite items. Many scholars have proposed different solutions to this problem. Feng et al. proposed a Deep Collaborative Latent Factor Model (DeepCLFM), which extracts deep nonlinear feature vectors of users and items from review embeddings through a bidirectional Gate Recurrent Unit (GRU). Additionally, DeepCLFM introduces the attention mechanism to measure the contribution of each review and adopts the matrix factorization module to learn latent factors according to the IDs of users and items. Finally, to fully integrate the deep nonlinear features and latent factors, DeepCLFM generates a deep interaction of them in the first and second-order fashion to predict the user's rating of the item [11]. Shen proposed a recommendation algorithm based on the combination of content and collaborative filtering. The main process of the algorithm is that the k-means clustering algorithm is used to cluster the users of the data set, then the appropriate weight of each attribute of the user is determined, and the new user is assigned to the appropriate class according to the clustering method of the user demographic characteristics, finally, the nearest neighbor of the new user is extracted. According to the item score of the nearest neighbor, calculating the pre-rating of a new user on a nonrated item and generating a list of recommendations [12]. Cai et al. used Convolutional Neural Network (CNN) to build a basic user-item matrix, and used the relationship between users and items to get user preferences, then combined user-friend preference relationships with the recommendation system to finally get the recommendation results [13]. Jiang et al. introduced the fuzzy matrix of item attributes, and used fuzzy clustering method to measure the relevance of items, then classified all item attributes. At the same time, the weight of item relevance is introduced in the calculation of user similarity, which makes the nearest neighbor search more accurate, thereby improving the accuracy of the recommendation result [14]. Liu et al. used the tag as a content of the item information, then a keyword vector is generated. At the same time, by introducing the label weight, it can avoid the influence of different items with different emphasis on the recommendation result, to further improve the accuracy of the recommendation algorithm [15]. He et al. proposed a hybrid recommendation algorithm that is provided by combining Latent Dirichlet Allocation_Matrix Factorization (LDA_MF), Latent Dirichlet Allocation_Collaborative Filtering (LDA_CF) and item-based collaborative filtering models. A large real data set is used to evaluate the proposed method [16].

The traditional solution is to ask some internal users to pre-rate the new item before the item goes online, to achieve the purpose of accumulating raw data, thereby solving the problem of cold start of the new item. However, this method cannot fundamentally solve the cold start problem of a completely new item. Yu et al. proposed the tripartite graphs that are used to picture the relations

among user-item-tag and user-item attribute. Combing information among users, tags, attributes of items, and time weights, the functions for predicting the rating are defined and a new personalized recommendation algorithm is constructed. This method is quite effective in solving the cold start problem of new items in the recommendation algorithm [17]. Wang et al. proposed a collaborative filtering recommendation algorithm based on item fusion auto-encoder. In the feature extraction, the inherent information of the item was used as the input of the auto-encoder, and the low-dimensional nonlinear feature representation was obtained by multidimensional dimensionality reduction. When constructing the model, the obtained low-dimensional features were integrated into the latent vector of the items in the matrix decomposition, and the user's historical interactive items were used to construct the user's latent vector, which alleviated the problem of cold start [18]. For solving the problem of cold start, Chen et al. proposed a collaborative filtering recommendation (CFR) algorithm combining singular value decomposition (SVD) and classification model (CM), which is called strongly connected components (SCC) algorithm. The classification model based on machine learning is used to obtain the recommended tags, and the collaborative filtering model based on SVD is used to obtain the items to be recommended [19]. For solving the problem of cold start of new users and new items in the collaborative filtering recommendation algorithm based on matrix factorization, Li et al. obtained the feature vectors of new users and new items by using the attribute-feature mapping algorithm based on K nearest neighbors. This method solves the cold start problem faced by this type of collaborative filtering algorithm [20].

It can be seen from the above literature that the current service recommendation methods have the following problems:(1) In the current mobile application service recommendation methods, the mobile application services with a high predicted usage frequency are recommended for users, while ignoring new services that are less evaluated by most users. This leads to the problem of incomplete and inaccurate recommendations. (2) When solving the issues of item cold start, the potential content feature of the service itself are ignored, and the judgment of the similarity between the services is not comprehensive, which causes the problem of recommendation deviation. In order to solve the above problems, this paper combines a recommendation method based on content feature extraction with collaborative filtering recommendation method to form a hybrid recommendation method. By taking more potential content features of services into account, this method not only improves the accuracy of service recommendation, but also, to a certain extent, alleviated the cold start problem.

3 Hybrid Recommendation Method Based on Content Feature Extraction

3.1 Hybrid Recommendation Model

As mobile application service providers provide increasingly mobile application services on the platform, there are increasingly mobile applications that users can choose from. However, in mobile application platforms, the mobile applications that are more common and easy to be selected by users have limitations. Unrated mobile apps are difficult to discover by users. In response to this problem, the method proposed in this paper will effectively solve the problem of cold start of new projects. The model flow chart is shown in Fig. 1.

The recommendation model based on content feature extraction can be divided into two parts. The left half of the model uses dependency syntax analysis technology to process version update information, function description information, and function characteristic information to form content feature text. And through the BERT preprocessing model to process the classification information and content feature text, and finally generate the content feature vector. The right half of the model adopts the idea of calculating service similarity through user ratings in the traditional collaborative

filtering algorithm and constructs a user rating matrix from existing data. The content feature vector is used in the weighted word distance algorithm TFIDF-WMD proposed in this paper to calculate the similarity of content features. User-rating matrix is used in Pearson similarity calculation method to calculate item similarity. Finally, the two similarities are weighted and accumulated to obtain the mixed similarity. And the recommendation result is obtained based on the prediction score ranking of the mixed similarity.

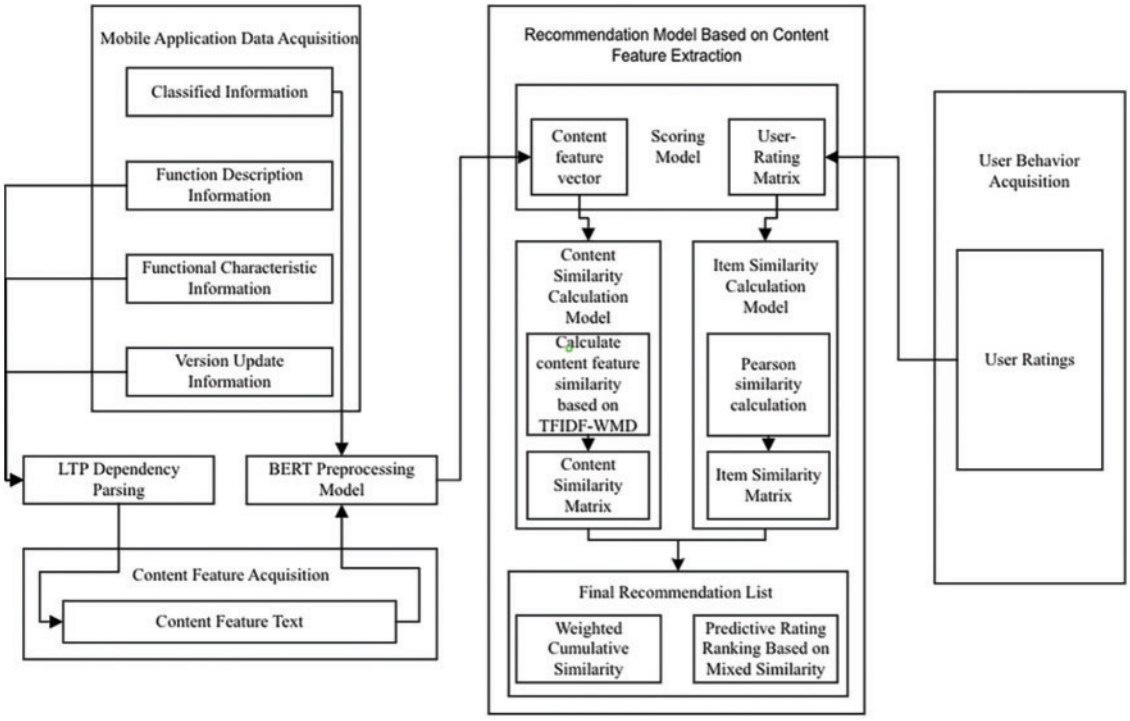

Figure 1: Recommendation model based on content feature extraction

3.2 Content Feature Extraction Method Based on NLP

3.2.1 Content Features and Content-Based Recommendation

Content features are the core basis of content-based recommendation methods. Finding suitable content features also helps to improve the accuracy and efficiency of content-based recommendation methods. Since text content is the most likely content feature to describe the content comprehensively and specifically, how to accurately extract, quantify, and participate in the calculation of the recommendation algorithm is the most important and most expansive problem in content-based recommendation methods. Therefore, in content-based recommendation algorithms, the key issue is to extract content features and calculate content features.

The basic principle of content-based recommendation is based on users' interest preferences. This can recommend items with similar content characteristics to the user's favorite items. When an item has structured information, the characteristics of each dimension can be used as a dimension of the vector. Vectorization means that the value of each dimension is not necessarily a numerical value, but it is in the form of vectorization, which is the Vector Space Model (VSM). Usually in content-based recommendation methods, the method of calculating the similarity of two items is as follows:

Suppose the vector representations of two items are $V_1 = (p_1, p_2, \ldots, p_k)$ and $V_2 = (q_1, q_2, \ldots, q_k)$. The similarity of the same-dimensional components p_t and q_t representing the same-dimensional content features is $\text{sim}(p_t, q_t)$. Then the similarity of the two items is shown in the following formula (1).

$$\text{sim}_{\text{item}}(V_1, V_2) = \sum_{t=1}^{k} w_t * \text{sim}(p_t, q_t) \tag{1}$$

where w_t is the weight of the t-th component. The value of this weight can be manually set through understanding of the business, or it can be obtained through training and learning of machine learning algorithms.

In the problem scenario of this paper, the mobile applications to be recommended to users in the mobile application platform recommendation system contain many available content features. Through multiscale principal component analysis [21,22], we found a series of useful features, such as, the classification information of the mobile application, the function description information of the mobile application, the function characteristic information, the version update information of the mobile application, and so on. For each content feature, we further get some more feature details by referring to signal decomposition method [23,24]. The content features of mobile applications is shown in Tab. 1 below.

Table 1: Content features of mobile applications

Content features	Feature details	Description
Classified information	General information	According to the actual situation, mobile applications are divided into three categories: tools, games, newspapers and magazines.
	Subcategory information	Among the above three categories, tools are divided into 24 sub-categories such as business, weather, tools, and tourism; games are divided into 18 sub-categories such as action, sports, simulation, and strategy; newspapers and magazines are divided into 26 sub-categories such as automobile, cooking, science, and art.
Function description information	Subject-verb phrase information	The subject-verb phrase used to describe the mobile application in the function description text of the mobile application. Such as: business services (“商家服务” in Chinese), food delivery (“外卖提供” in Chinese), and other phrases.
	Verb phrase information	The verb-object phrase that exists in the function description text of the mobile application to describe the mobile application. Such as: providing channels, tracking locations. And other phrases.

(Continued)

Table 1: Continued

Content features	Feature details	Description
Functional characteristic information	Fixed phrase information	The central phrase used to describe the mobile application in the function description text of the mobile application. Such as: air quality (“空气质量”in Chinese), city scenery (“城 市风景” in Chinese), and other phrases.
Version update information	Update status information	According to the actual situation, the function information of the mobile application is not only the initial function introduction part, but also includes the version upgrade information included in each update.
	Update time information	Time information for each version update of the mobile application.

By analyzing the features of all available content in this problem scenario, it can be known that all above content is semi-structured data given by the application service provider on the mobile application platform. From the perspective of the mobile application recommendation system, classification information and version update information should be directly available in the database of the system. However, for the function description information and function characteristic information used to describe the functions of mobile applications, if the function description text containing the above two kinds of information is directly vectorized, the problem of insufficient content features due to too much invalid information will occur. Therefore, how to effectively process the function description text and complete the extraction of information has become a very critical issue.

3.2.2 Content Feature Extraction Based on Natural Language Processing

(1) Phrase extraction based on dependency parsing function

In the problem scenario of this paper, according to the analysis of the previous content characteristics, it can be seen that the function description information in the function description text is usually the phrase containing the subject-verb (SBV) and the phrase containing the verb object (VOB). And the functional characteristic information contains the attribute (ATT) phrase. Therefore, this paper uses the part-of-speech tagging and dependency parsing method in the Language Technology Platform (LTP) of Harbin Institute of Technology to filter out the phrases that contain the relationship between SBV, VOB and ATT in the mobile application function description text. For the function description information in the function description text, the need to extract phrases that contain the relationship between SBV and VOB. As shown in Fig. 2 below, “customer call” (“顾客打电话” in Chinese) and “order goods” (“订购商品” in Chinese) are function description information.

For the function characteristic information in the function description text, it is necessary to extract the phrase containing the ATT relationship. As shown in Fig. 3 below, “the next four days"(“未来四天” in Chinese), “weather forecast"(“天气预报” in Chinese), and “real-time weather information"(“实时 天气信息” in Chinese) are the feature information.

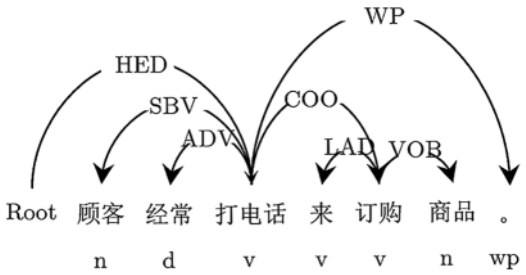

Figure 2: Dependent syntax analysis to extract functional description information

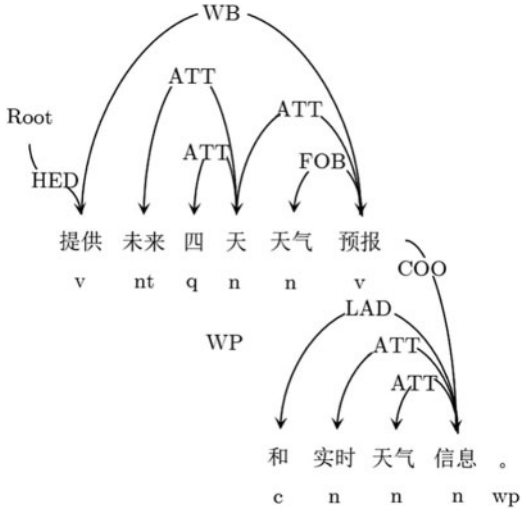

Figure 3: Dependent syntax analysis to extract functional characteristic information

Through this part of the functional phrase extraction based on dependency syntax analysis, the original functional description text with redundant and invalid information is transformed into the content feature text containing only the phrases of the above three syntactic structures.

(2) Vectorization of content feature text

Through the above method of extracting functional phrases based on dependency syntax analysis, we have obtained content feature texts that only contain SBV, VOB, and ATT syntactic structures. However, these texts are still unstructured and incalculable. Therefore, this paper uses part of the BERT-Based-Chinese pre-training model in BERT (Bidirectional Encoder Representation for Transformers) to complete the vectorized representation of Chinese content feature text. Turn the feature text into a structured and computable sparse vector.

In the problem scenario of this paper, we used the open source packaging library Bert-as-service developed by Tencent AI Lab based on the BERT-Based-Chinese pre-training model to vectorize the Chinese content feature text after LTP processing. In the Chinese prediction model, the corpus is processed in units of words. Therefore, there is no need for pre-segmentation before processing. According to the needs of the problem scenario in this paper, after configuring the corresponding functions of its service, enter the content feature text in batches. In the processed result, each vector is 768 dimensions, which becomes a structured and computable sparse vector.

3.3 Hybrid Recommendation Algorithm

3.3.1 Calculation Method of Content Feature Similarity

Term Frequency-Inverse Document Frequency (TF-IDF) term frequency weighting technology is used to evaluate the importance of words to the text in the document set or corpus. Term Frequency (TF) indicates how often a word appears in a document. Inverse document frequency (IDF) is used to evaluate the universality of words in the corpus. The Word Mover's Distance (WMD) algorithm is a method of calculating text similarity. The core idea of WMD is to move all words in a certain text to another text and minimize the problem of moving costs. Among the problems studied in this paper, based on the TF-IDF keyword weight technology to optimize the word distance algorithm, a weighted word distance algorithm TFIDF-WMD is proposed to calculate the similarity between different mobile application services. After that, the result of similarity between different mobile application services is used as a calculation factor to participate in the calculation of a hybrid recommendation algorithm based on content characteristics and item similarity. Through this method, the item could start problem caused by the single use of item-based collaborative filtering recommendation algorithm is alleviated, and the recommendation accuracy is effectively improved.

Suppose there is a trained word vector matrix $X \epsilon R^{d \times n}$, with a total of n words. The i-th column $x_i \epsilon R^d$ represents the d-dimensional word vector of the i-th word. The Euclidean distance between word i and word j is shown in the following formula (2).

$$c(i,j) = \| x_i - x_j \|_2 \tag{2}$$

And a content feature text used to describe the mobile application service a is processed by the BERT model and can be represented by the sparse vector $d_a \epsilon R^n$ as its bag of words. Assuming that in the content feature text, word i appears $n_{i,a}$ times, and the sum of the times of all words appearing in d_a is $\sum_k n_{k,a}$, then the TF value of word *i* is shown in the following formula (3).

$$tf_{i,a} = \frac{n_{i,a}}{\sum_k n_{k,a}} \tag{3}$$

Assuming that the total number of content feature texts in the corpus is |D| and the number of texts containing word i is $|\{a : i \in d_a\}|$. The IDF value of word i is shown in the following formula (4).

$$idf_i = \log \frac{|D|}{|\{a : i \in d_a\}|} \tag{4}$$

Then, in the content feature description text of the mobile application service a, the TF-IDF value of the word i is shown in the following formula (5).

$$tfidf_{i,a} = tf_{i,a} \times idf_i \tag{5}$$

There are two mobile application services a,b , let d_a and d_b respectively represent the word bag representation of the two content feature texts to be calculated, and each word i in d_a can be transferred to d_b in whole or in part. Define a sparse transition matrix $T \epsilon R^{n \times n}$, then $T_{ij} \geq 0$ represents how many words are transferred from word i in d_a to word j in d_b , $T_{ij} \geq 0$. Therefore, the sum of its transfer cost is $\sum_{i,j} T_{ij} c(i,j)$.

According to the idea of word distance algorithm, when the transfer cost sum is larger, the similarity between two content feature texts participating in the calculation is lower. That is, the similarity between two content feature texts is inversely proportional to the minimum transfer cost between the texts. After transforming the above problem into a linear programming problem, the text

similarity of content features between mobile application services a and b is $sim_{char}(a, b)$. As shown in the following formula (6). Where ω is the undetermined coefficient.

$$sim_{char}(a, b) = \frac{\omega}{\min\limits_{T\geq 0} \sum_{i,j=1} T_{ij} c(i,j)} \tag{6}$$

Make $\sum\limits_{j=1}^{n} T_{ij} = tfidf_{i,a}$, Satisfying condition $\forall i \in 1, \ldots, n$ and $\sum\limits_{i=1}^{n} T_{ij} = tfidf_{i,b}$, Satisfying condition $\forall j \in 1, \ldots, n$.

In the method described in this section, the WMD algorithm itself assumes that the importance of different words on a content feature text is the same, but this does not completely conform to the actual situation. Therefore, the TF-IDF value is used instead of a single word frequency value to participate in the calculation, which makes the method in this paper more reasonable. This improves the accuracy of content feature similarity calculation and reflects the flexibility and robustness of the WMD algorithm.

3.3.2 Item-Based Collaborative Filtering Recommendation Algorithm

In the traditional item-based collaborative filtering recommendation algorithm, it is generally assumed that there are two sets in the recommendation system: user set U, $U = \{u_1, u_2, u_3, \ldots u_m\}$ and item set I, $I = \{i_1, i_2, i_3, \ldots i_n\}$. Each individual user can rate all items, and each individual item can also be rated by all users. In the question studied in this paper, assuming that most of the mobile application services have been rated by users, after one-to-one correspondence between users and services, the following user-service (U-S) score matrix can be obtained. As shown in Tab. 2 below:

Table 2: User-service **(U-S)** score matrix

Rating	u_1	u_2	...	u_n
s_1	7	6		1
s_2	4	?		5
s_3	9	?		7
...				
s_n	2	8		?

In this method, we first calculate the similarity between the service and the service based on the user's rating of the mobile application service. Then the calculated similarity is used as the weight, and the predicted score of the user for the unrated service is obtained through weighted calculation.

There are many ways to calculate similarity based on users' ratings of services. This paper uses the Pearson similarity measurement method. This method has the advantage of eliminating calculation differences caused by different average values. Let $r_{u,a}$ represent the rating of user u for service a, $r_{u,b}$ represent the rating of user u for service b, and $\overline{r_u}$ represent the average rating of the user for the rating service. Then the similarity between service a and service b is $sim_{rating}(a,b)$, as shown in the following formula (7).

$$sim_{rating}(a,b) = \frac{\sum_{u\in U}(r_{u,a} - \overline{r_u})(r_{u,b} - \overline{r_u})}{\sqrt{\sum_{u\in U}(r_{u,a} - \overline{r_u})^2}\sqrt{\sum_{u\in U}(r_{u,b} - \overline{r_u})^2}} \tag{7}$$

From the above calculation process, it is not difficult to see that if you only consider the item similarity calculated based on the user's rating of the item and use a single variable as the weight to calculate the user's predicted rating for the unrated item, there is an obvious item could start problem. That is, when a new mobile application service s_{new} is added to the recommendation system, no historical user rating data can be used to calculate the similarity between the service and the service. This will make it impossible to recommend new services to users in time. On the other hand, due to the inability to recommend in time, the rate at which new items receive user ratings is slow. Due to the inability to recommend in time, the rate at which new items receive user ratings is slow. This will prevent new items from being accurately and effectively recommended to potential users for a long time in the future.

3.3.3 Hybrid Recommendation Algorithm Based on Content Features and Item Similarity

According to the item cold start problem in the item-based collaborative filtering recommendation algorithm described in the previous section. This paper proposes a hybrid recommendation algorithm based on content features and item similarity.

Suppose two mobile application services a and b. According to the previous content feature extraction method based on natural language processing and content feature similarity calculation method based on TF-IDF and word distance algorithm. It is calculated that its service similarity based on content feature text is $sim_{char}(a, b)$. According to the aforementioned item-based collaborative filtering algorithm. Calculate its service similarity based on user ratings as $sim_{rating}(a,b)$. Then the mixed similarity of service a and b is sim(a, b), as shown in the following formula (8).

$$sim(a, b) = \lambda \cdot sim_{char}(a, b) + (1 - \lambda) \cdot sim_{rating}(a,b) \tag{8}$$

Use the mixed similarity calculated by the above method as the weight. Calculate the user' s predicted score for unrated services through weighted calculation. Suppose pred(u,p) is the predicted score of user u for service p. Suppose sim(i, p) is the mixed similarity between the scored service i and the predicted service p. Then the predicted score is shown in the following formula (9).

$$pred(u,p) = \frac{\sum_{i \in rateditems(u)} sim(i, p) * r_{u,i}}{\sum_{i \in rateditems(a)} sim(i, p)} \tag{9}$$

This mixed method has the following advantages. When the value of the control coefficient λ is reasonable, there is no need to worry about the cold start problem of items that only have the content characteristics of the service itself and have no historical scoring data after the new service enters the recommendation system. And with the operation of the recommendation system, the new service will quickly obtain new user rating data. When the user rating data changes incrementally, the recommendation effect will gradually approach the ideal value of this method.

Based on the above algorithm ideas, a hybrid recommendation algorithm based on content feature extraction, HRACFE (Hybrid Recommendation Algorithm based of Content Feature Extraction), is described as Algorithm 1 in Tab. 3.

Table 3: HRACFE algorithm

Algorithm 1 Hybrid recommendation algorithm HRACFE based on content feature extraction
Input: target user u_i, user's rating information set for the service $Socer_{set}$, service set$Service_{set}$
Output: Recommended service set $Toplist_{set}$ of target user u_i
Begin
1: for each s ∈ $Service_{set}$ do
2: s ←DEPENDENCY_SYNTACTIC_PARSING (s);
3: s ←BERT_VECTORIZATION (s);
4: num←num + 1;
5: end for
6: for i = 1 to num do
7: for j = 1 to num do
8: if $S_i \neq S_j$ then
9: $sim_{char}(S_i,S_j)$ ←TFIDF-WMD (S_i,S_j);
10: $sim_{rating}(S_i,S_j)$ ←PEARSON (s_i,s_j);
11: $sim(S_i,S_j) \leftarrow 0.7 \cdot sim_{char}(S_i,S_j)+0.3 \cdot sim_{rating}(S_i,S_j)$;
12: end if
13: end for
14: end for
15:$Toplist_{set}$ ←TOP-N (pred(u_i, S));
16:return $Toplist_{set}$;

In the above algorithm, we perform the dependency syntax analysis on each service in the service set by step 2; and use the Bert model to complete the vectorized representation of the content feature text by step 3; then in step 9 we use the weighted word distance algorithm TFIDF-WMD which is proposed in Section 3.3.1 to calculate the similarity between different mobile application services. And from step 10 to step 11, by combining with the calculated service similarity based on user ratings, the hybrid similarity between services is obtained. Finally, we selected the top N apps as the results to recommend to users.

4 Experimental Results and Analysis

4.1 Experimental Evaluation Index

This paper chooses F-measure and Novelty as the evaluation indicators of the experiment. $F-measure$ is a commonly used index for evaluating the accuracy of an algorithm. This indicator balances the accuracy of an algorithm by considering both precision and recall. Its formula is shown in the following formula (10).

$$F-measure = \frac{2 \times P \times R}{P + R} \tag{10}$$

Use P stands for Precision, and R stands for Recall. In the problem scenario of this paper, the calculation formula is shown in the following formulas (11) and (12).

$$P = \frac{NUM_{recomture}}{NUM_{recomtotal}} \tag{11}$$

$$R = \frac{NUM_{recomture}}{NUM_{total}} \tag{12}$$

where $NUM_{recomture}$ represents the number of applications correctly recommended to users. $NUM_{recomtotal}$ represents the number of applications recommended to users. NUM_{total} represents the number of applications in the target user test set.

The Novelty reflects the proportion of new services in the recommended list recommended to users. Assuming that the number of new services in the recommendation list is N, and the number of new services contained in it is N_{new}. It is as shown in the following formula (12).

$$Novelty = \frac{N_{new}}{N} \tag{13}$$

4.2 Experimental Program

The experimental data set in this paper is obtained by crawling the mobile application service and user information data of the application pool through the Scrapy crawler framework. AppChina is a mobile application service platform that stores a large amount of mobile application service information. The service information includes the category, function information, update records of the mobile application, the number of registered users, and the user's score for the mobile application that has been used. The mobile application classification of this data set includes 6 categories, all of which are included in the mobile application classification information defined in Section 3.2.1 of this paper. The total number of mobile application services is 24,437, with 2162 registered users, and users' ratings for mobile applications are between 1–10.

The experiment randomly selected 20 groups of mobile application services from the data set, each group included 1000 cases of mobile application services, of which the training set accounted for 70% and the test set accounted for 30%. Taking into account the need to test the cold start performance of alleviating items in this method, 150 cases in each group are randomly selected and assumed to be new services, and the corresponding 150 groups of user score data are set to be empty. There are some repetitions in the sampling of 20 sets of mobile application services, and finally the average value of the 20 sets of test data is taken as the experimental result. The method in this paper is compared with the content-based recommendation method and the traditional collaborative filtering recommendation method, and the results are analyzed.

4.3 Experimental Results and Analysis

1) Determination of the value of the mixed similarity coefficient λ

In the recommendation process, the value of the mixed similarity coefficient needs to be determined. Tab. 4 shows the recommendation result F-measure when each user recommends 20 mobile application services, the content feature inverse proportional coefficient w is 1, and the mixed similarity coefficient λ is 0, 0.5, and 1.

It can be seen from the above results that the recommended performance is the best when $\lambda = 0.7$.

2) Comparison of F-measure and Novelty between different recommended methods.

Table 4: When w is 1, the **F-measure** value under different λ value

λ	F-measure
0.1	0.49
0.2	0.52
0.3	0.53
0.4	0.55
0.5	0.57
0.6	0.58
0.7	0.59
0.8	0.52
0.9	0.51

In the figure below, Figs. 4–6 respectively show the Precision rate, Recall rate and F-measure value of the recommended results of different methods. Fig. 7 shows the Novelty value of the recommendation results of different methods.

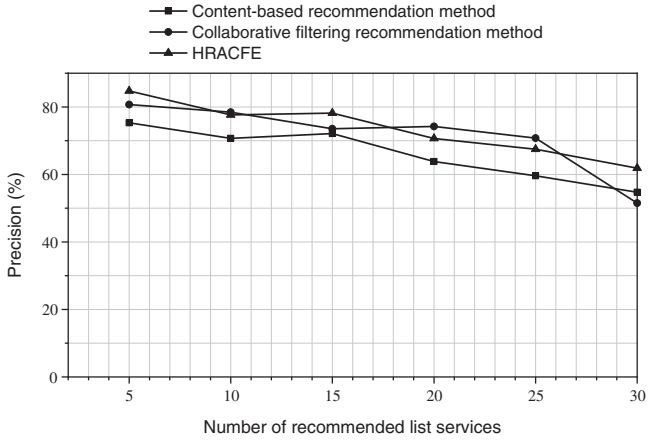

Figure 4: The Precision rate of the recommended results of different methods

It can be seen from the data results in the figure that when the number of recommendations is less than 20, the accuracy of this method is better than traditional content-based recommendation methods and collaborative filtering recommendation methods; with the expansion of the recommendation list, the decreasing trend of the accuracy of this method is similar to the traditional content-based recommendation method. The traditional collaborative filtering recommendation method cannot handle the cold start problem of new projects, so the novelty of the recommendation result is 0. In the case of considering user ratings, the novelty of the recommendation results of the method in this paper almost reaches the same performance as traditional content-based recommendation methods. Based on the above test results, the method proposed in this paper has better actual recommendation performance and has a certain ability to alleviate the cold start of items.

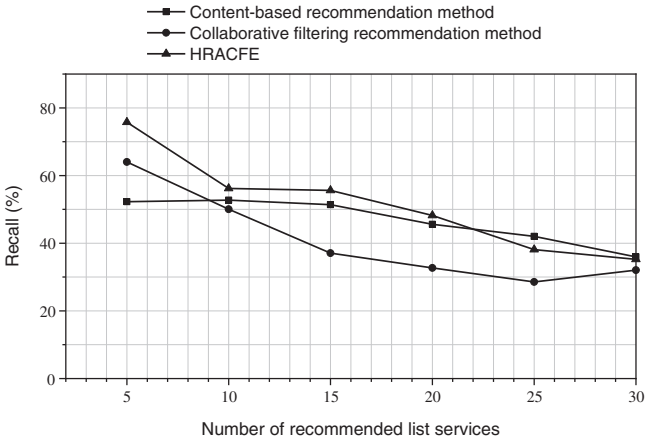

Figure 5: The Recall rate of the recommended results of different methods

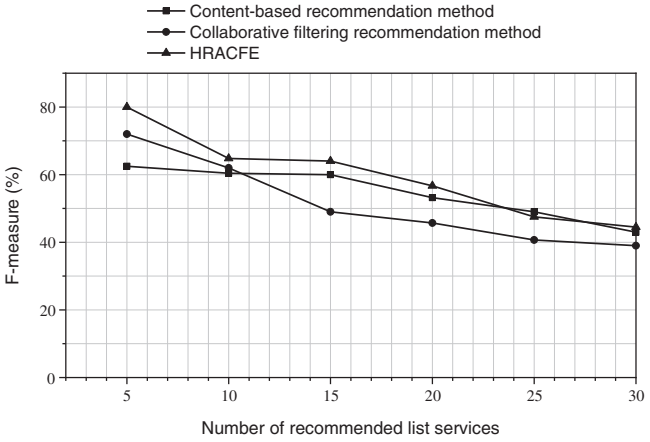

Figure 6: The **F − measure** value of the recommended results of different methods

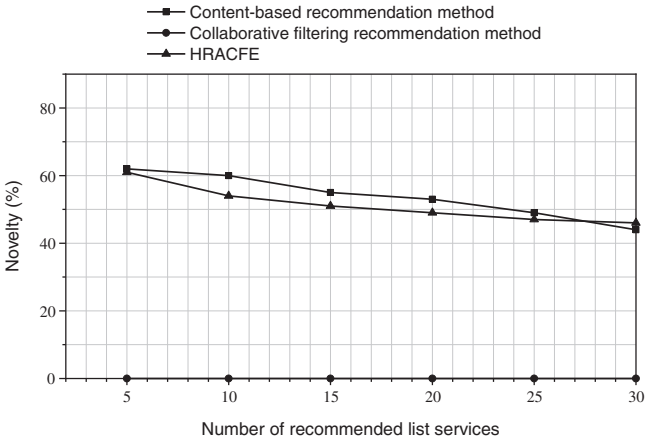

Figure 7: The **Novelty** value of recommendation results of different methods

5 Conclusion

This paper combines the content feature extraction-based recommendation method and the item-based collaborative filtering recommendation method to propose a hybrid recommendation method. Content features, as the inherent attributes of services in the recommendation system, play a pivotal role in solving the problem of service cold start. In the proposed method, the NLP technology is used for content feature extraction, and the similarity of content features and the similarity of user ratings are integrated to alleviate the cold start problem of services in the recommendation system. Combining the similarity of content features with the similarity of user ratings can also improve the performance of the recommendation system. The proposed method has been tested and proved to be effective. The proposed method is suitable as the recommendation engine of application service stores, which is because of that on the one hand, compared with traditional content-based recommendation methods, the proposed method in this paper takes advantage of more content features and better combines the similarity of user ratings in the collaborative filtering algorithm, making the recommendation result more reliable and accurate; on the other hand, compared with the traditional collaborative filtering algorithm based on user rating similarity, the proposed method in this paper solves the problem of not being able to recommend new services, and it also has good recommendation performance for services with few user reviews.

Funding Statement: Project supported by the National Natural Science Foundation, China (No. 62172123), the Postdoctoral Science Foundation of Heilongjiang Province, China (No. LBH-Z19067), the special projects for the central government to guide the development of local science and technology, China (No. ZY20B11), the Natural Science Foundation of Heilongjiang Province, China (No. QC2018081).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Q. Long, W. F. Long, Z. T. Li and K. L. Li, "A Game-based approach for cost-aware task assignment with QoS constraint in collaborative edge and cloud environments," *IEEE Transactions on Parallel and Distributed System*, vol. 32, no. 7, pp. 1629–1640, 2020.

[2] Y. Koren and R. Bell, "Advances in collaborative filtering," in *Recommender Systems Handbook*, Boston, MA, USA: Springer, pp. 77–118, 2015.

[3] H. N. Yu, J. G. Shu, X. H. Jia and H. L. Zhang, "lpRide: Lightweight and privacy-preserving ride matching over road networks in online ride hailing systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 10418–10428, 2019.

[4] H. N. Yu, X. H. Jia, H. L. Zhang and X. Z. Yu, "PSRide: Privacy-preserving shared ride matching for online ride hailing systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 68, no. 11, pp. 1425–1440, 2021.

[5] M. Y. Zhang, X. Geng and G. S. Deng, "A novel service recommendation approach in mashup creation," *Intelligent Automation & Soft Computing*, vol. 25, no. 3, pp. 513–525, 2019.

[6] M. Deshpande and G. Karypis, "Item-based top-N recommendation algorithms," *ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 143–177, 2004.

[7] Y. Qiao and L. J. Li, "Research on solution of solving cold start problem in recommender systems," *Computer Technology and Development*, vol. 28, no. 2, pp. 83–87, 2018.

[8] Z. T. Li, W. L. Li, F. Y. Li, Y. Sun, M. Yang *et al.,* "Hybrid malware detection approach with feedback-directed machinelearning," *SCIENCE CHINA Information Sciences*, vol. 63, no. 3, pp. 240–242, 2020.

[9] D. K. Duan and X. F. Fu, "Research on user cold start problem in hybrid collaborative filtering algorithm," *Computer Engineering and Applications*, vol. 53, no. 21, pp. 151–156, 2017.
[10] C. X. Ren, "Improved algorithm of alleviating item cold starting," *Computer Engineering & Software*, vol. 37, no. 8, pp. 11–15, 2016.
[11] X. J. Feng and Y. Z. Zeng, "Joint deep modeling of rating matrix and reviews for recommendation," *Chinese Journal of Computers*, vol. 43, no. 5, pp. 884–900, 2020.
[12] H. L. Shen, "Recommendation algorithm based on the combination of content and collaborative filtering," *Computer Knowledge and Technology*, vol. 14, no. 2, pp. 232–234 + 282, 2018.
[13] C. C. Cai, H. H. Xu, J. Wan, B. Q. Zhou and X. W. Xie, "An attention-based friend recommendation model in social network," *Computers, Materials & Continua*, vol. 65, no. 3, pp. 2475–2488, 2020.
[14] W. Jiang, J. Chen, Y. Jiang, Y. Xu, Y. Wang *et al.,* "A new time-aware collaborative filtering intelligent recommendation system," *Computers, Materials & Continua*, vol. 61, no. 2, pp. 849–859, 2019.
[15] Y. Liu and W. H. Zhu, "A hybrid recommendation algorithm combination of content-based and tag weight," *Computer & Digital Engineering*, vol. 48, no. 4, pp. 773–777, 2020.
[16] L. Huang, C. J. Lin, J. He and H. Y. Liu, "Diversified mobile app recommendation combining topic model and collaborative filtering," *Journal of Software*, vol. 28, no. 03, pp. 708–720, 2017.
[17] H. Yu and J. H. Li, "Algorithm to solve the cold-start problem in new item recommendations," *Journal of Software*, vol. 26, no. 6, pp. 1395–1408, 2015.
[18] D. Wang, F. Xue, K. Liu, S. Y. Chen and H. B. Zhang, "Collaborative filtering recommendation algorithm based on item fusion auto-encoders," *Journal of Computer Applications*, vol. 39, no. S1, pp. 84–87, 2019.
[19] J. X. Chen, H. Q. He, Y. F. Pan and Y. W. Wu, "Collaborative filtering algorithm based on classification model and SVD," *Electronic Measurement Technology*, vol. 43, no. 14, pp. 69–73, 2020.
[20] G. Li and L. Li, "A new algorithm of cold-start in a collaborative filtering system," *Journal of Shandong University (Engineering Science)*, vol. 42, no. 2, pp. 11–17, 2012.
[21] P. Worrajiran, "Towards development of brain-computer interface based on point to point movements," Ph.D. dissertation, University of Strathclyde, Glasgow, Scotland, United Kingdom, 2009.
[22] M. T. Sadiq, X. J. Yu, Z. H. Yuan, M. Z. Aziz, S. Siuly *et al.,* "A matrix determinant feature extraction approach for decoding motor and mental imagery EEG in subject specific tasks," *IEEE Transactions on Cognitive and Developmental Systems*, pp. 1–1, 2020, https://10.1109/TCDS.2020.3040438.
[23] M. T. Sadiq, X. J. Yu, Z. H. Yuan, Z. M. Fan, A. U. Rehman, G. Q. Li *et al.,* "Motor imagery EEG signals classification based on mode amplitude and frequency components using empirical wavelet transform," *IEEE Access*, vol. 7, pp. 127678–127692, 2019.
[24] M. T. Sadiq, X. J. Yu, Z. H. Yuan, Z. M. Fan, A. U. Rehman *et al.,* "Motor imagery EEG signals decoding by multivariate empirical wavelet transform-based framework for robust brain-computer interfaces," *IEEE Access*, vol. 7, pp. 171431–171451, 2019.