Systems and Soft Computing 6 (2024) 200109

Contents lists available at ScienceDirect

Systems and Soft Computing

journal homepage: www.journals.elsevier.com/soft-computing-letters

The design of advertising text keyword recommendation for internet search engines

Fang Wang [*], Liuying Yu

School of Design, Hainan Vocational University of Science and Technology, Hainan, 571126, China

ARTICLE INFO

Keywords:
Search engines
Advertising text
Entity naming
Feature extraction

ABSTRACT

As the growth of internet technology, human life is full of various advertisements. It is possible for individuals to obtain the advertising information they require, whether in an online or offline context. A research proposal is presented with the objective of enhancing the precision of online advertising recommendations. The proposal is based on the design of internet search engine advertising text keyword recommendation models, which integrate entity naming recognition models to facilitate tasks such as text classification and feature extraction. A recommendation algorithm based on content similarity is used to achieve keyword recommendation. Under the similarity calculation method of continuous bag-of-words model, when K is 100, the model weighted precision of the feature extraction method based on graph sorting and inverse text frequency index is 0.88, the weighted recall is 0.76, and the weighted F1-score is 0.82. In offline simulation testing, 85 % of the keyword recommendation model's recommendation time is less than 1 s, 99 % of the recommendation time is less than 2 s, and the recommendation cost can be significantly reduced by 75 %. In practical applications, the recommendation efficiency of this method can reach 96.3 %, and the recommendation precision can reach 95.8 %. The recommended satisfaction rate can reach 99.5 %. The results demonstrate that this method can provide highly accurate keyword recommendations and reduce the cost of advertising placement. Furthermore, it has been recognized and praised by users.

1. Introduction

As the growth and popularization of internet technology and online advertising, search engines become an indispensable way for humans to obtain information. It has changed human online behavior and provided different business models for many companies. Search service companies that provide users with free-search related information can obtain a large amount of data and traffic, and monetize information through advertising business. Nowadays, internet companies both domestically and internationally, such as Baidu and Google, have invested a large amount of manpower and resources to carry out corresponding advertising businesses, providing high-quality services to users and advertisers while earning profit returns. Keyword advertising is one of the important forms of search advertising. Customers query by inputting keywords, and advertising providers provide highly relevant search results based on the input keywords. Keyword search can reduce online user costs and increase access to useful information within a limited range, which has become an important component of users' online search information [1-3]. Keywords are important link hubs between users, advertisers, and search companies. The most important step for advertisers is to choose keywords, as effective keywords can reduce advertising costs and user search time policy costs [4-5]. Based on this, search companies need to continuously optimize search engines to ensure the effectiveness of keyword search advertisements. Therefore, the study proposes keyword recommendation (KR) for internet search engine advertising text, aiming to improve its precision and reduce advertising placement costs and user search time costs. The main contribution of the research lies in proposing a deep learning-based algorithm for KRs in advertising texts for internet search engines, which includes two parts: an entity naming recognition (ENR) model and a KR algorithm. The ENR model can accurately identify entities in text, including city and core business information, and extract keywords. Based on KR methods, feature extraction can be achieved through two methods: discrete word bag model and skip word model, which can achieve good recommendation results. This method has high scalability and processing efficiency, and can handle large-scale data, making it suitable for the research and application of recommendation systems.

[*] Corresponding author.
E-mail address: wangfang19850329@hotmail.com (F. Wang).

https://doi.org/10.1016/j.sasc.2024.200109
Received 25 December 2023; Received in revised form 14 May 2024; Accepted 3 June 2024
Available online 12 June 2024

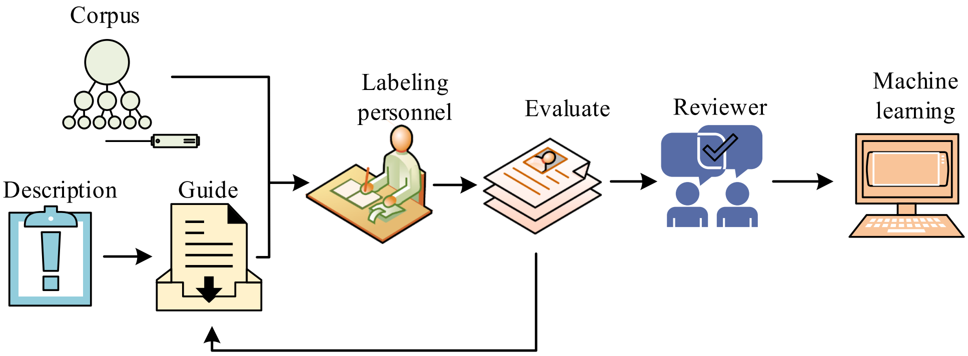


Fig. 1. Topic information extraction process.

2. Related works

Companies and businesses increase revenue through advertising placement. Whether online or offline, users or consumers will receive various personalized advertising recommendation information. To improve the efficiency of advertising, Wu research group proposed a personalized recommendation method based on a new perspective, which is improved by combining natural language processing and data mining. Experimental data showed that this method could improve the effectiveness of recommending products and reduce a certain amount of advertising investment [6]. Li and his team members proposed a personalized implicit recommendation method for learners, which calculates the trust between learners based on their interactive behavior in social networks, and used the three degree influence theory to mine learners' implicit friends. Experimental results showed that this method was reliable and effective in personalized recommendation and interpretability [7]. Cai et al. proposed a new inductive heterogeneous graph neural network model that utilized the relationship information in user cold start recommendation systems to alleviate the sparsity of user attributes. The experimental results indicated that the performance of this recommendation model was superior to state-of-the-art baselines [8]. Karn research team proposed a recommendation model based on hybrid recommendation model (HRM) and hybrid sentiment analysis (HSA), which generated a preliminary recommendation list using HRM, and then generated the final recommendation list using HRM with HSA. The experimental results indicated that this method could improve recommendation precision and user satisfaction [9]. To improve the interaction between users and projects in the simulation system, researchers such as Wang proposed a collaborative filtering recommendation algorithm based on dynamic clustering and similarity measurement in the hypergraph. A hypergraph model was utilized to capture complex relationships beyond pairwise relationships, and a hypergraph network was constructed using an extended game dynamics clustering algorithm. The experimental results showed that the algorithm had good prediction and recommendation performance [10].

Keywords are very important in the advertising recommendation, and users will obtain search results related to keywords through queries and searches. Cheng et al. proposed a keyword-citation-keyword network to analyze the subject knowledge structure. It highlighted the importance of keyword distribution in articles and analyzed the semantic relationships of keywords in different articles. The experimental results indicated that this method could identify indirect connections between keywords [11]. Duari and Bhatnagar proposed a supervised framework for automatic keyword extraction from a single document. It designed a supervised keyword extraction method based on the interaction between node attributes, and constructed a feature training set based on keyword label allocation. Experimental data showed that the traditional keyword extraction method performed better [12]. The Madon research team used a keyword ranking search method based on documents and e-commerce websites to improve the precision of keyword ranking, and imported a dataset from Kaggle and converted it into a more useful format. The results indicated that genetic algorithms had higher precision in keyword ranking than artificial neural network algorithms [13]. Sharma and Jinwala proposed a centralized token generation method for multi-write, multi-read, and searchable encryption problems. It combined keyword search with conjunctions to construct a search token without interacting with trusted institutions of the enterprise. Experiment outcomes indicated that this method could effectively resist external interference when selecting keywords [14]. The Du research group proposed an iterative encryption method to ensure data security. It implemented keyword mapping through Bloom filters and build indexes in conjunction with cloud servers. The experimental results indicated that this method had safety and effectiveness [15].

In summary, the above research has made some progress in advertising recommendation and personalized recommendation, but there are still some problems. There are still issues in advertising recommendation methods where advertising costs cannot be completely reduced. Personalized recommendation methods are limited by the social behavior of learners, and the performance of inductive heterogeneous graph neural network models still needs to be improved. HRMs and HSA methods need further validation. There are limitations in semantic relationship analysis in research related to key words. Keyword extraction methods still need to be optimized. Keyword ranking search methods need to consider other factors. Centralized token generation methods require better keyword selection strategies, and iterative encryption methods have some limitations. Therefore, the study proposed a design for KR in advertising text for internet search engines, with the aim of improving the precision of advertising recommendations and reducing advertising placement costs.

3. Design of KR for advertising text for internet search engines

3.1. Entity naming recognition model and keyword vocabulary design

The ENR model is used to complete tasks such as text classification and feature extraction, which facilitates the construction of KR models in the future [16]. Firstly, it is necessary to annotate the data, and the research adopts a distributed annotation form. The data annotation is shown in Fig. 1.

In the process of data annotation, the first step is to determine the desired target label. Then, it is necessary to understand the definition of feature, which means removing the city and core parts from the keywords. City represents the content related to the city in the text, while core represents the core, important, or valuable parts of the text. These two tags help distinguish information about cities and core content when processing text data. Next, it needs to refer to the guidelines to determine which words need to be labeled as city and core. During the annotation process, it is necessary to evaluate the annotation results. If encountering uncertain data or ambiguous situations, it is necessary to

Table 1
Original keyword data display.

Variable name	Variable Interpretation	Missing judgment
Keyword	Keyword	No
Keyword_ID	Keyword identification	No
Refer	Data sources	No
City	Place in City	Yes
Top_category	Class 1	No
Category	Class 2	No
Meta	Class 3	Yes
Length	Keyword length	No
Shows	The exposure of advertising words on Baidu	No
Clicks	The clicks generated by the placement words on Baidu	No
Cost	Consumption generated by posting words on Baidu	No
Uv	Number of users accessing the site	No
Pv	Number of visited pages on the site	No
Page_nums	Number of pages accessed by keyword dimension	No
Contact_uv	Number of users accessing the site	Yes
Contact_pv	Number of visited pages on the site	Yes
Leads	Mobile phone number retention	Yes
Label	Is it manually launched	Yes
Legal	Is it prohibited	Yes
Dt	Format date	No

add or delete guidelines. Finally, after the reviewer accepts the annotated results, the machine learning stage can be entered [17]. The expression for calculating the effectiveness of the ENR model is shown in Eq. (1).

$$f1-score = 2\times\frac{precision\times recall}{precision+recall} \tag{1}$$

In Eq. (1), *precision* represents precision, which refers to how many true case samples are predicted to be positive. *recall* represents recall rate, which refers to how many true case samples are predicted to be positive. $f1-score$ is the harmonic average of precision and recall rate, called F1-score, and its macro average and weighted average are used to evaluate the overall effect.

The KR algorithm is to recommend word information to users, so it is necessary to establish a keyword vocabulary to further improve the precision and effectiveness of advertising by collecting the features of keywords [18]. The source of the vocabulary includes historical keywords and customer search terms, and the original keyword data content is shown in Table 1.

The keyword content searched by users is complex and cannot be directly used for placement. The main problems are colloquialism, phone numbers, website addresses, garbled code, and even some contain prohibited information related to pornography and politics. Therefore, it is necessary to review and clean such keywords. If there are prohibited words in a keyword phrase, it is likely to be a prohibited keyword [19]. The study combines the naive Bayesian method to construct a prohibited keyword review model, where labeled historical keywords only require simple processing. The naive Bayesian model is shown in Eq. (2).

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)} \propto P\left(d\middle|c)P(c)=P(c)\prod_{1\le k\le n_d}P(w_k|c\right) \tag{2}$$

In Eq. (2), the length of document d is represented by n_d. The prior probability of category c is represented by $P(c)$. The probability that document d belongs to category c is represented by $P(c|d)$, and the frequency of word w_k appearing in category c is represented by $P(w_k|c)$. The calculation is shown in Eq. (3).

$$P(w_k|c) = \frac{count(w_k,c)+1}{\sum_{w\in w^*}(count(w,c)+1)} \tag{3}$$

In Eq. (3), all word sets are w^*, and the number of occurrences of w_k in c is $count(w_k,c)$. To prevent the occurrence of a probability of 0, an addition operation is required in the equation. The probability of a given document belonging to each category can be calculated using Eq. (3), and the equation for calculating the category with the highest probability is shown in Eq. (4).

$$\hat{c} = \operatorname{argmax}\underset{c\in c^*}{P}(c|d) \tag{4}$$

In Eq. (4), the category with the highest probability is $\hat{c}$, and the set of all categories is c^*.

Keyword cleaning requires a combination of regular expressions and some rule-based methods. Firstly, regular expressions are used for matching, where the matching of phone and number is relatively

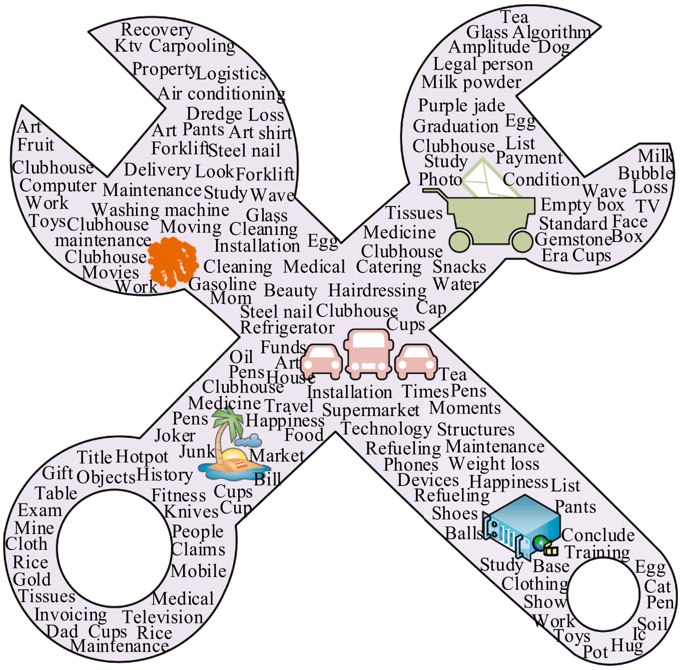


Fig. 2. Core business word cloud chart.

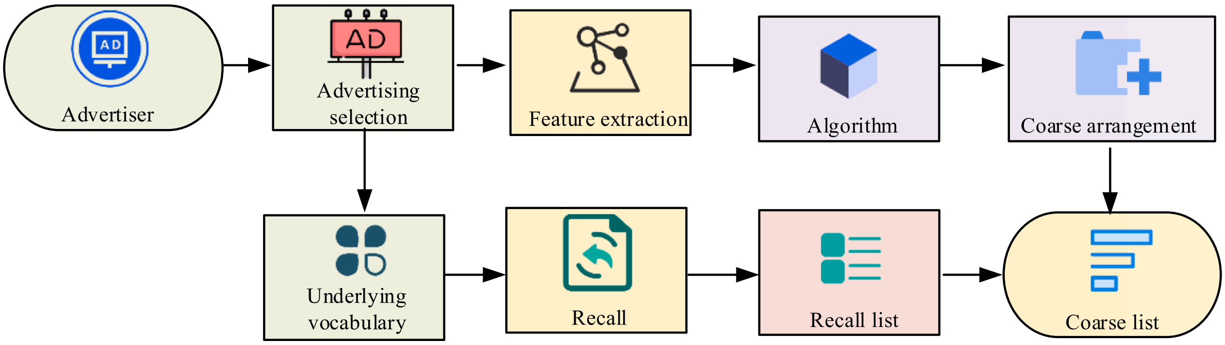


Fig. 3. A recommendation algorithm model based on content similarity.

complex, and manual sampling is used to find as many as possible [20]. Then it needs to solve the pure city problem. If a keyword only contains city or place name information and does not include business content, then it is an invalid keyword. The study uses ENR matching rules to determine whether keywords are valid and delete invalid keywords. If a keyword does not contain business content information, it cannot enter the advertising phase. Extreme level word replacement, such as keywords containing the most advanced, fashionable and other words, requires the deletion of the most advanced words to avoid exaggerated promotion. If the keywords involved in advertising have sensitive information, corresponding punishment measures are required. At this time, a trained naive Bayesian model is needed to review and filter prohibited words. Brand words need to be filtered in conjunction with the offline maintained brand word dictionary. Long or short keywords can lead to poor advertising effectiveness, so the keyword length range is 4 to 15, and the remaining keywords with long keywords should be deleted. After the above operation, the cleaned data is obtained and the missing values are filled with zero. The core business and city of keywords require word frequency statistics, and the core business word cloud diagram is shown in Fig. 2.

The core business vocabulary cloud map is a vocabulary centered graphical method used to display core business vocabulary in a specific field or industry. It can assist search engines in designing ENR models for advertising text KRs, thereby improving the precision and efficiency of text analysis and information extraction. The core business vocabulary cloud map is usually composed of multiple levels, each representing an important set of business vocabulary. It can help people quickly understand the main vocabulary and concepts in the business field, thereby better performing text analysis and information extraction.

3.2. Design and implementation of KR algorithm

The research adopts a recommendation algorithm based on content similarity to achieve KR. Firstly, the content is understood to obtain keywords, and then the similarity between word vectors is calculated to obtain a keyword list. After segmenting the original content and removing the inactive words, it needs to train the word vector. Because of the high cost of manual verification and the high time cost of online testing, offline simulation testing is used for evaluation [21]. It connects the randomly selected posts to the recommendation interface for filtering, deletion, and deduplication. If the number of recommendations is insufficient, it is necessary to expand the dataset based on search engine recommendation results. The resulting dataset format is shown in Eq. (5).

$$\left\{\left(post_i, key_{ij}\right)\middle| key_{ij} \in \left(rec_1(post_i) \cup rec_2(post_i)\right)\right\} \tag{5}$$

In Eq. (5), the ith post is represented by $post_i$. The jth keyword of $post_i$ is represented by key_{ij}. The keyword set recommended by the keyword planner is represented by rec_1, and the keyword set recommended by the search engine is represented by rec_2. The calculation for the precision of recommended results is shown in Eq. (6).

$$precision = \frac{cunt(|R(u) \cap T(u)|)}{count(|R(u)|)} \tag{6}$$

In Eq. (6), the KR list for real posts is $R(u)$; the recommendation algorithm list is $T(u)$, and the calculation for recall rate is shown in Eq. (7).

$$recall = \frac{cunt(|R(u) \cap T(u)|)}{count(|T(u)|)} \tag{7}$$

In Eq. (7), *recall* represents the recall rate. The F1-score is still the harmonic average of the two, and the calculation is the same as Eq. (1), without repeating the explanation. After determining the evaluation indicators, the recommendation algorithm model based on similar content is shown in Fig. 3.

The recommendation algorithm model based on content similarity is an algorithm that calculates and sorts the similarity between target advertisements and post content. Firstly, it selects the target advertisement selected by the advertiser and passes in four parameters: post title, post content, placement city, and secondary category. Then, it will select the secondary category as the index, filter it in the underlying vocabulary, and obtain a simple recall list. Next, it will extract key information from the target advertisement, calculate the similarity with the keywords in the recall list, and sort them in reverse order according to the similarity size. Finally, it is recommended to use the Top-K keywords as a rough list as input for the subsequent fine-tuning section.

The study used the Term Frequency Inverse Document Frequency (TF-IDF) model, Textrank model, and Textrank-IDF model for feature extraction, with the same number of extracted features. Among them, TF-IDF was utilized to calculate the significance of a certain word in the document, and the term frequency was TF, which represents the probability of a certain word appearing in the document. The TF calculation is presented in Eq. (8).

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{8}$$

In Eq. (8), the term frequency of the ith word w_i in the jth document d_j is $tf_{i,j}$. The number of occurrences of the ith word in the jth document is $n_{i,j}$, but some non representative words in the document may have a higher frequency. The inverse document frequency is IDF. If a word appears in many documents, its importance will decrease. The calculating for the IDF is shown in Eq. (9).

$$idf_i = \log \frac{|D|}{|\{j : w_i \in d_j\}| + 1} \tag{9}$$

In Eq. (9), the IDF value of the ith word w_i is idf_i, and the number of documents in the corpus is $|D|$. The denominator plus one operation is to avoid the situation where the denominator is zero. When IDF performs feature extraction alone, it tends to look for rare words and cannot achieve completely correct feature extraction. Therefore, it is necessary to combine TF and IDF to form the TF-IDF model, and its calculation is

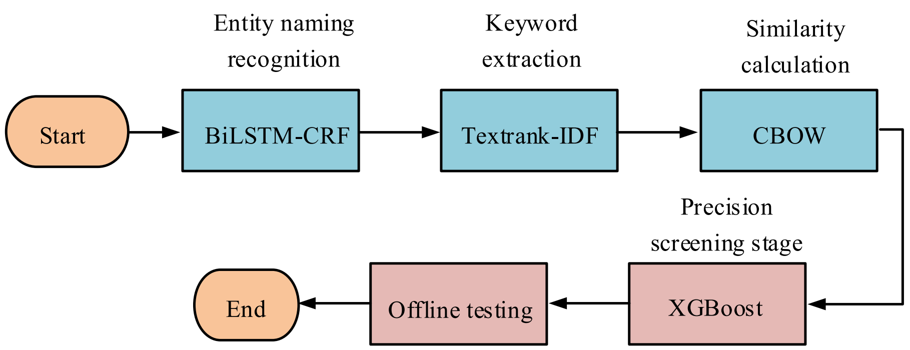


Fig. 4. Keywords recommend the workflow steps.

shown in Eq. (10).

$$fidf_{i,j} = tf_{i,j} \times idf_i \tag{10}$$

In Eq. (10), the TF-IDF value of the ith word in the jth document is $fidf_{i,j}$, and the larger its value, the higher its importance. The calculation for Textrank is shown in Eq. (11).

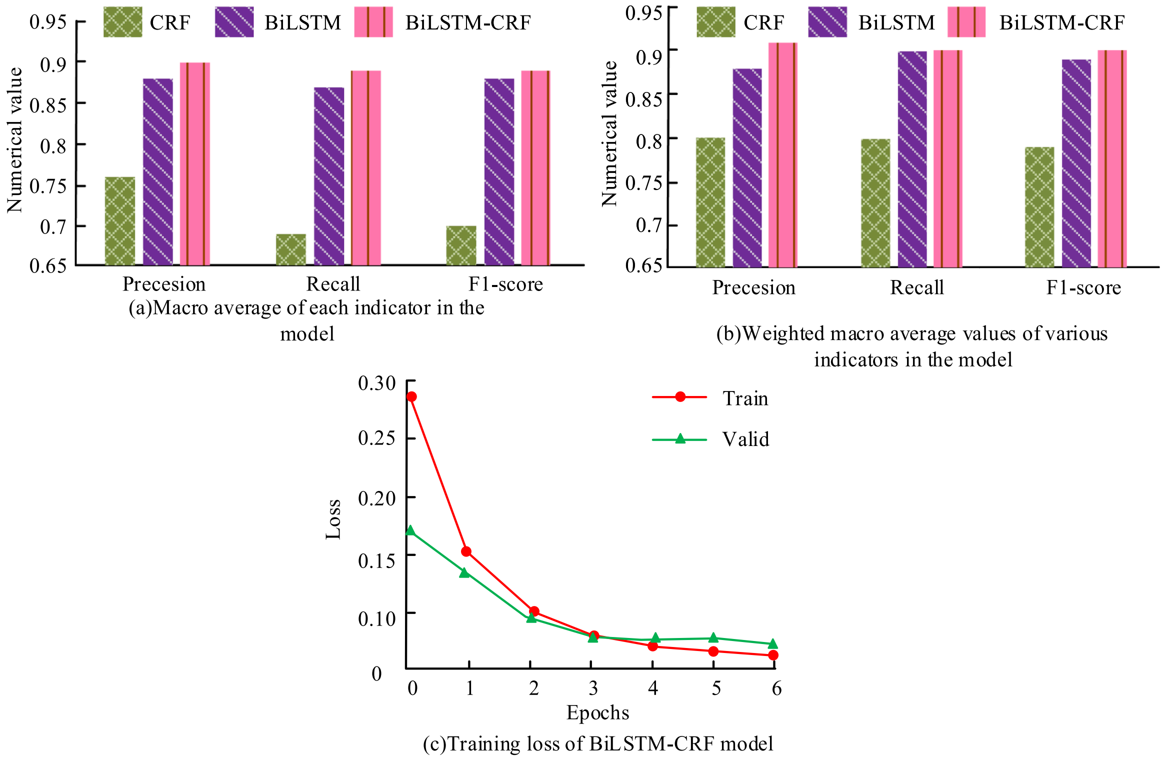
$$TR(w_i) = (1-d) + d \times \sum_{w_j \in \ln(w_i)} \frac{Q_{ji}}{\sum_{w_k \in Out(w_j)} Q_{jk}} TR(w_j) \tag{11}$$

In Eq. (11), the Textrank score for word w_i is $TR(w_i)$. The damping coefficient is d, and the importance of words w_i and w_j is represented in Q_{ji}. Due to the fact that the Textrank algorithm does not consider the relationships between documents, a new Textrank-IDF model is developed by combining the Textrank algorithm with the IDF algorithm. The calculation is shown in Eq. (12).

$$TRIDF(w_i) = TR(w_i) \times idf(w_i) \tag{12}$$

In Eq. (12), the Textrank-IDF score of the word w_i is represented by $TRIDF(w_i)$. In similarity calculation method, assuming A is a feature sequence extracted from advertisements and B is the sequence after keyword segmentation in the recall sequence, the Jaccard similarity calculation method utilizes the idea of perfect matching for similarity calculation. TF-IDF encodes sequences A and B and calculates the cosine similarity between the two sequences. Both methods in Word2vec calculate cosine similarity in a word vector environment.

The estimation of click through rate is the first step in fine-tuning, which is an evaluation and settlement indicator for the effectiveness of search advertising and directly affects the revenue of advertising companies. The click through rate of KR is the average that of advertisements displayed on a webpage, which is equal to click through divided by exposure. After calculating the estimated click through rate, this data cannot be directly used and requires feature engineering and transformation processing. Firstly, it groups according to the keyword dimension, denoted as Rank. Then it uses the click through rate value with the keyword Rank of 1 as the target label. It deletes the row data with Rank 1, calculates the monthly average values of each indicator, and aggregates them with the target label. Finally, it compares the target keywords of the same category, and the final variables are screened by the regularization rules and cross tested to determine the five variables before and after the coefficient ranking. The average click through consumption estimation is the second step of fine-tuning, using the same



(a)Macro average of each indicator in the model

(b)Weighted macro average values of various indicators in the model

(c)Training loss of BiLSTM-CRF model

Fig. 5. Experimental data of entity naming recognition model training.

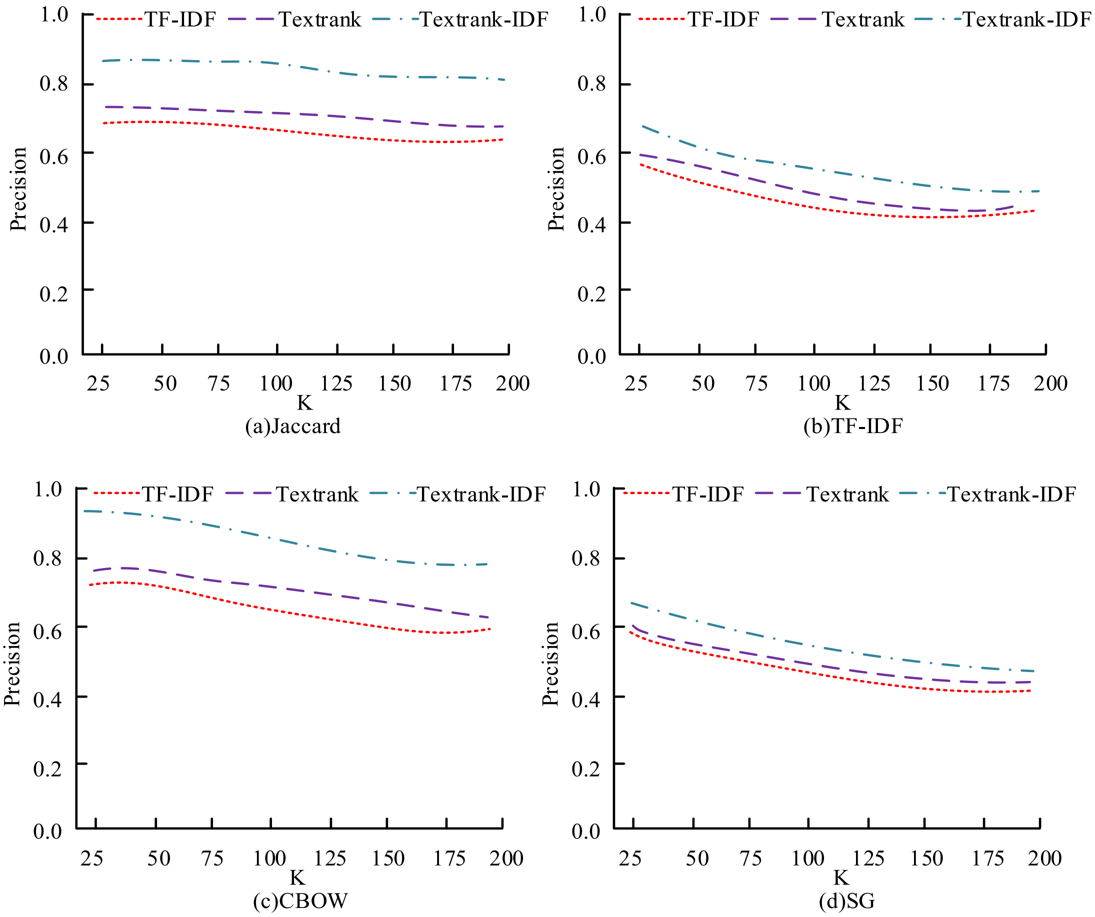


Fig. 6. Macro average precision of feature extraction methods.

data as the click through rate estimation. Firstly, it calculates the click through consumption value of the keyword, then creates a format date and calculates the average click through consumption value of each target month. It marks the click through cost value greater than or equal to this average as 1, and the remaining part as 0. It converts the average click through consumption estimation problem into a binary classification problem through the above steps. There are many binary classification algorithm models used in research. Firstly, the naive Bayesian model is one of the traditional statistical models. Assuming conditional independence between features, the word model is computationally simple, but there are also situations where it cannot solve classification problems. To improve the traditional regression model, a set of logical functions is added to the linear regression to transform the range from (0,1). The XGBoost algorithm model has higher precision and the advantages of parallelism and portability. Another integration method for decision trees is the gcForest algorithm, which consists of two parts: cascaded forest and multidimensional forest.

4. Simulation and analysis of KR algorithms for advertising text for internet search engines

4.1. Verification of entity naming recognition model effectiveness

The experiment used the first 70 % of the data in the corpus as the training set, the remaining 30 % as the testing set, and then took 10 % of the data in the training set as the validation set, introducing the early stop method. The steps of keyword recommendation are shown in Fig. 4.

In the workflow of keyword recommendation, the first step is to preprocess the data, including cleaning, segmenting, and removing stop words; Then, entity naming recognition is performed, and the model is trained using the BiLSTM-CRF method. Next, extract keywords based on the Textrank IDF feature extraction method. Afterwards, calculate the similarity and use the CBOW model. On this basis, determine the optimal parameters recommended by Top-K based on the experimental results. In the fine-tuning stage, compare the LR model, gcForest model, and XGBoost model. Finally, conduct offline simulation testing to evaluate the recommended response time and cost. The experiments used the Conditional Random Field (CRF), the Bi-directional Long Short Term Memory (BiLSTM) and the BiLSTM-CRF methods for model training, and deleted the non entity results. The experimental data is shown in Fig. 5.

Fig. 5(a) and Fig. 5(b) show the macro average and weighted average values of each indicator of the model trained using each method. The precision of the CRF method training model was 0.76, and the weighted precision was 0.80. The precision of the BiLSTM method training model was 0.88, and the weighted precision was 0.88. The precision of the BiLSTM-CRF method training model was 0.90, the weighted precision was 0.91, and the recall rate and F1-score were higher than those of the CRF method and BiLSTM method. The experimental results indicated that the best method for training the model was BiLSTM-CRF. Fig. 5(c) shows the training loss values of the BiLSTM-CRF training model. Starting from the third epoch in the validation set, the loss value tended to stabilize and the model fitting effect was good. The results showed that the BiLSTM-CRF method trained the best model and met the practical application requirements, with an precision and recall rate of 90 % for named entity models.

4.2. Verification of KR algorithm effectiveness

The experiment used Jaccard, TF-IDF, and Word2vec's continuous bag-of-words model (CBOW) and Skip Gram (SG) similarity calculation

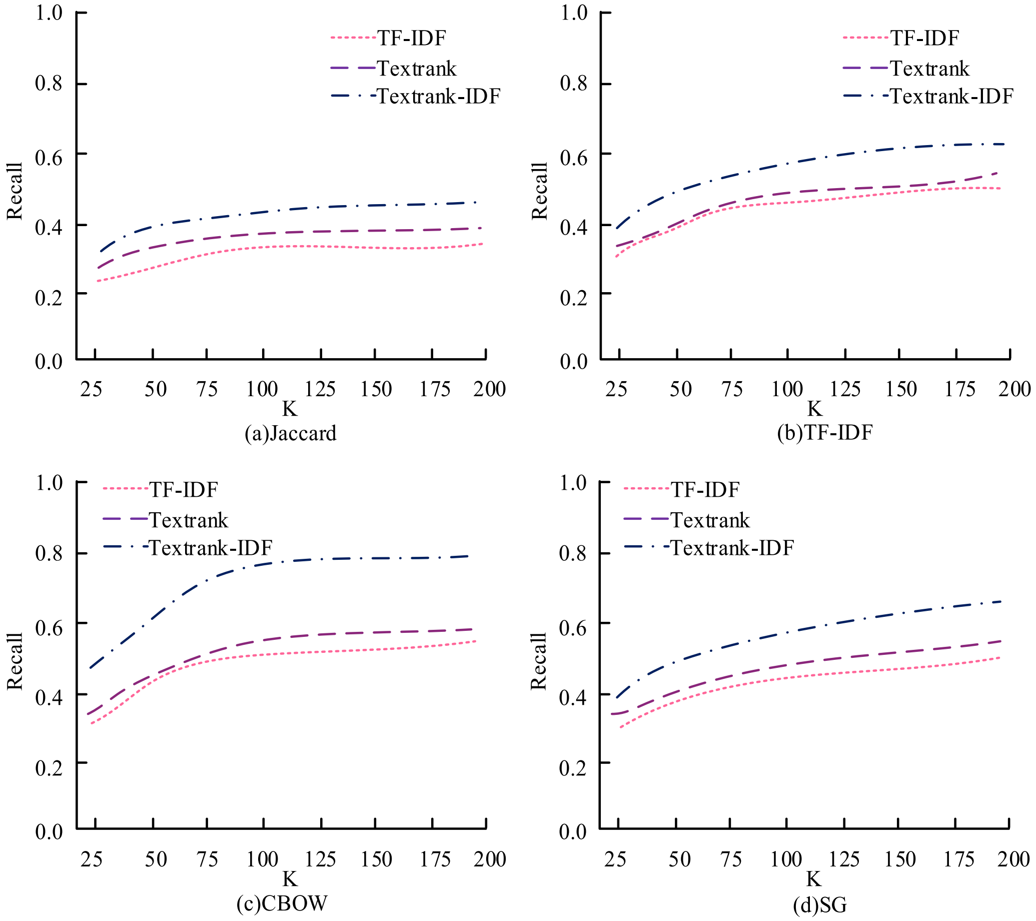


Fig. 7. Macro average recall rate of feature extraction methods.

methods to construct word vectors and calculate the similarity, and it set the threshold value of Word2vec method to 0.6. After completing the parameter setting, the experiment recommended Top-K from 25 to 200 with a gradient of 25. The macro average precision of different feature extraction methods under different similarity calculation methods is shown in Fig. 6.

As shown in Fig. 6, in all four cases, the precision of the Textrank-IDF feature extraction method was higher than that of TF-IDF and Textrank feature extraction methods. Among various similarity calculation methods, the CBOW model had the best performance. Under this algorithm model, the Top-K recommendation precision of the Textrank-IDF feature extraction method was higher than 0.8. As the K value increased, the recommendation precision showed a downward trend. The macro average recall of different feature extraction methods under different similarity calculation methods is shown in Fig. 7.

As shown in Fig. 7, the recall rate would increase with the raising of K value. Regardless of the similarity calculation method, the recall rate of the Textrank-IDF feature extraction method was higher than that of the TF-IDF and Textrank. Moreover, under the CBOW model similarity calculation method, the Top-K recommendation recall rate of the Textrank-IDF mostly exceeded 0.6, while the recall rates of the TF-IDF and Textrank were both below 0.6. The macro average F1-scores of different feature extraction methods under different similarity calculation methods are illustrated in Fig. 8.

In Fig. 8, as the K value increased, the F1-score first increased and then decreased. The Top-K recommended F1-score for the Textrank-IDF feature extraction method under the CBOW model was higher than 0.6, while the F1-score for both TF-IDF and Textrank feature extraction methods was less than 0.6. Analysis of experimental data shows that the CBOW model has the excellent training performance. When K was 100, the model had the best training effect. The Textrank-IDF feature extraction method had a macro average precision of 0.87, a recall rate of 0.75, and an F1-score of 0.81. Under the CBOW similarity calculation method, each feature extraction method weighted the macro average of each indicator, and the results are shown in Fig. 9.

In Fig. 9, the trend of each indicator was the same as the macro average result, with a slight increase in the precision indicator. Under the CBOW similarity calculation method, when K was 100, the model weighted precision of the Textrank-IDF feature extraction method was 0.88, the weighted recall rate was 0.76, and the weighted F1-score was 0.82. The experimental data showed that among the four similarity calculation methods, the CBOW model had the best training effect. In the experiment on the pass rate of the first click in the fine-tuning stage, 80 % of the dataset was randomly selected as the training set, and the remaining 20 % was used as the testing set for model training. The experiment compared the traditional LR model, gcForest model, Support Vector Machine (SVM), Decision Tree (DT), and XGBoost model. The model comparison experiment results are presented in Fig. 10.

In Fig. 10, the feature vector dimension sensitivity of the LR model and SVM were relatively high. The gcForest model and XGBoost model were basically not affected by this, and the optimal model was XGBoost, with an auc value close to 0.95. It introduced the city and core business information of the named entity, and selected different dimensions K for the word vector. The results are expressed in Fig. 11.

In Fig. 11, as K increased, the auc values of each model showed an upward trend. The XGBoost model still had the highest auc value. When adding a 20 dimensional word vector, its auc value was 0.802. When adding a complete 400 dimensional vector, the auc value rose to 0.823. The precision of the gcForest model was not as good as that of the traditional LR model at this time. Therefore, the XGBoost model was

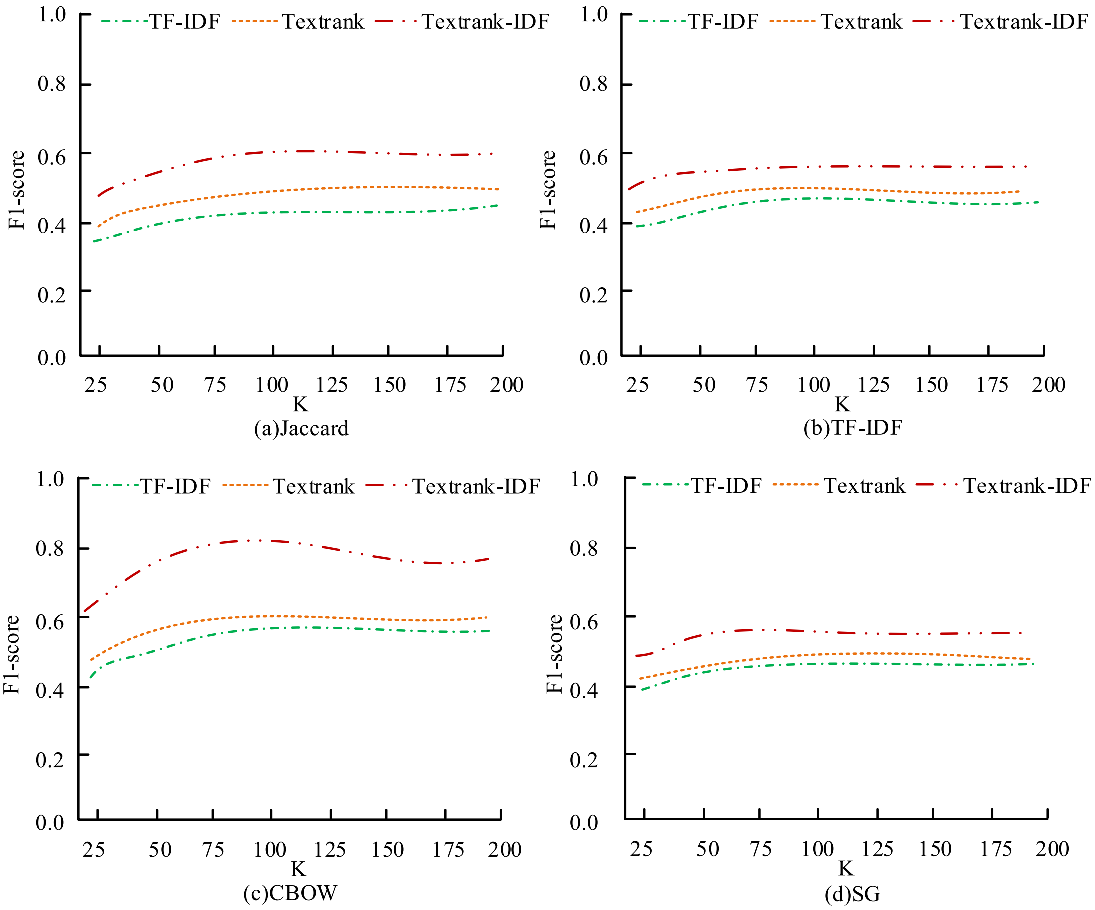


Fig. 8. Macro average F1-score of feature extraction method.

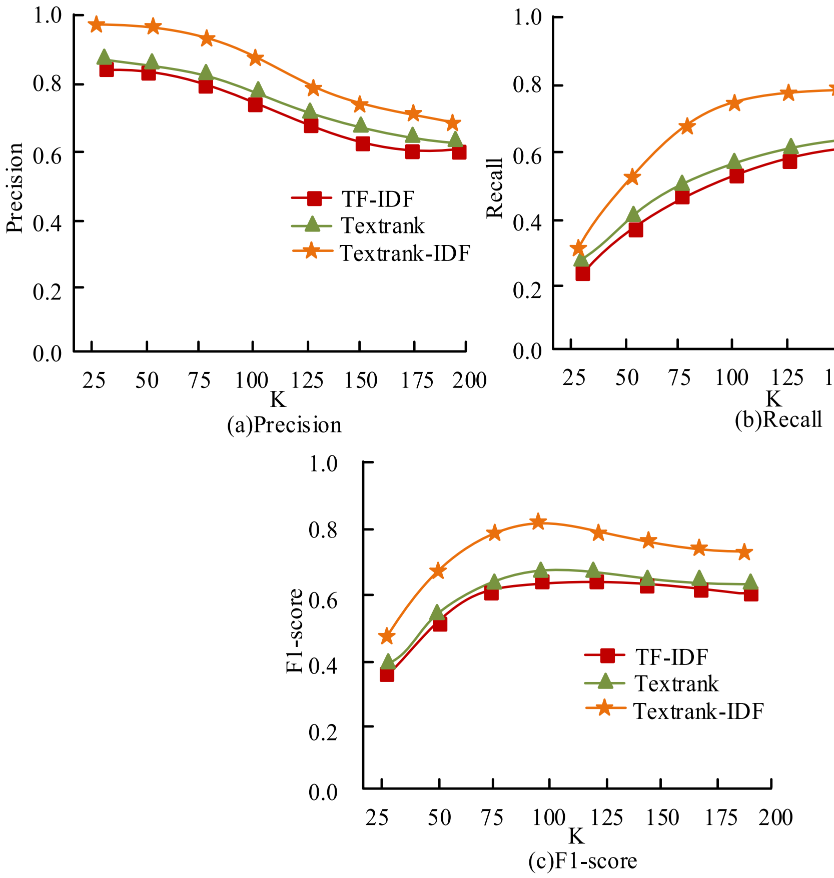
ultimately selected for the experiment to estimate the click through rate. Next, offline simulation testing would be conducted to recommend the data of 5000 posts that have already been formed. The recommendation effect is shown in Fig. 12.

Fig. 12(a) shows the recommendation response time. 85 % of the recommendation time was less than 1 s, and 99 % of the recommendation time was less than 2 s. According to the analysis of experimental data, the overall mean was 0.5 and the variance was 0.27. The hypothesis test was conducted for its mean value. The original hypothesis was that the average response time was greater than or equal to 1 s, and the alternative hypothesis was that the average response time was less than 1 s. Under the condition of large samples, based on the central limit quantification, it was assumed that the mean value follows normal distribution. Through calculation, the *P* value was less than 0.001, indicating that the original hypothesis was not tenable. This indicated that the response time of the model met the online requirements. In addition to response time, research also needed to focus on recommendation costs. Fig. 12(b) shows the distribution of cost difference, which basically follows the normal distribution. The same hypothesis test results as the response time hypothesis test results could significantly reduce costs by 75 %, indicating that the method met the design requirements. In practical applications, the recommendation efficiency of this method could reach 96.3 %, and the recommendation precision could reach 95.8 %. This method demonstrated excellent performance in recommendation and could effectively provide users with valuable recommended content. This method also performed well in terms of recommendation precision, and could provide more accurate recommendation suggestions based on the user's historical behavior and preferences. User feedback showed that the recommendation satisfaction of this method could reach 99.5 %, indicating that the method could meet the expectations and needs of users, and has received recognition and praise from users.

The experimental results showed that the Textrank-IDF feature extraction method had higher precision than TF-IDF and Textrank feature extraction methods under all four similarity calculation methods. Under the CBOW model similarity calculation method, the Top-K recommendation precision of the Textrank-IDF feature extraction method mostly exceeded 0.6, while the recall rate was all above 0.6. The research method performed well in terms of recommendation precision and efficiency, and could provide relatively accurate recommendation suggestions based on user's historical behavior and preferences. User evaluations showed that their recommendation satisfaction could reach 99.5 %. These results indicated that the method performed well in recommendation and could effectively provide valuable recommendation content to users.

5. Conclusion

Advertisers can achieve fast and convenient service and product promotion through advertising, becoming an important source of commercial monetization for advertisers. To improve the precision of advertising placement and reduce costs, a KR model for internet search engine advertising text was proposed. It adopted a recommendation algorithm based on content similarity to achieve KR. The experimental results showed that the precision, recall, and F1-score of the BiLSTM-CRF method training model were higher than those of the CRF method and BiLSTM method, and the model fitting effect was good. In addition, the research on this work is of great significance for improving the precision of advertising placement and reducing placement costs. And in the first click through rate experiment of the precision sorting

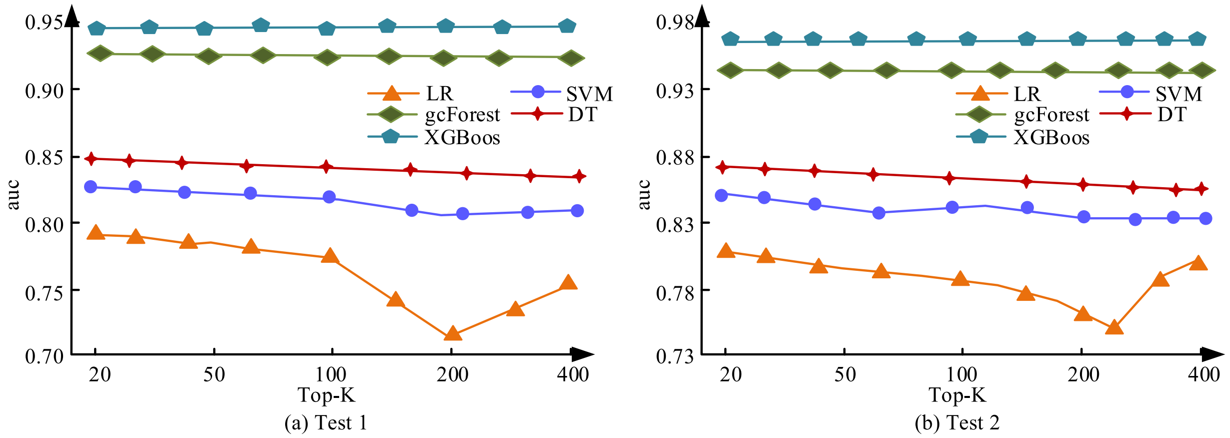


(a)Precision

(b)Recall

(c)F1-score

Fig. 9. Macro average index weighted average for feature extraction methods.



(a) Test 1

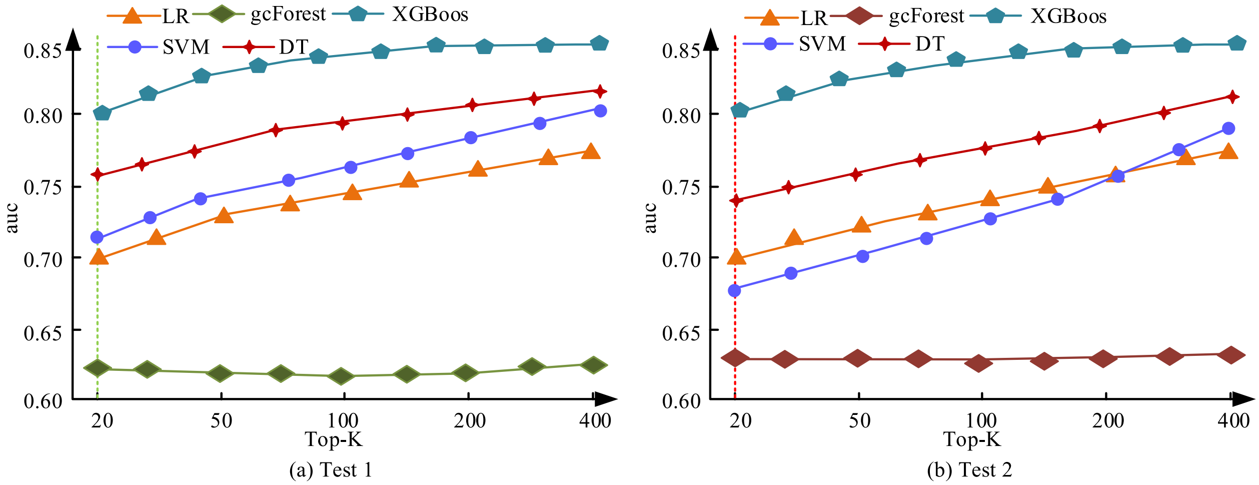
(b) Test 2

Fig. 10. Click through rate experimental data.

stage, the feature vector dimension of the LR model was more sensitive, but the XGBoost model was not affected by this, and its auc value was close to 0.95. In offline simulation testing, 85 % of the recommendation time of the KR model was less than 1 s, 99 % of the recommendation time was less than 2 s, and the recommendation cost could be significantly reduced by 75 %, indicating that the method met the design requirements. The user satisfaction evaluation could reach 99.5 %, indicating that this method met the expectations and needs of users. Therefore, this work provides effective methods and models for the advertising placement field, and important support for commercial monetization. The limitation of this study lies in the named entity recognition model stage. Although annotated data is obtained, the precision of the model still needs to be improved. In terms of the construction of the underlying vocabulary, keyword cleaning currently relies more on rule cleaning, and online automation only fixes rules. Future research can further increase annotated data and attempt to add convolutional layers to the BiLSTM-CRF model to enhance the internal feature learning ability of the model.

CRediT authorship contribution statement

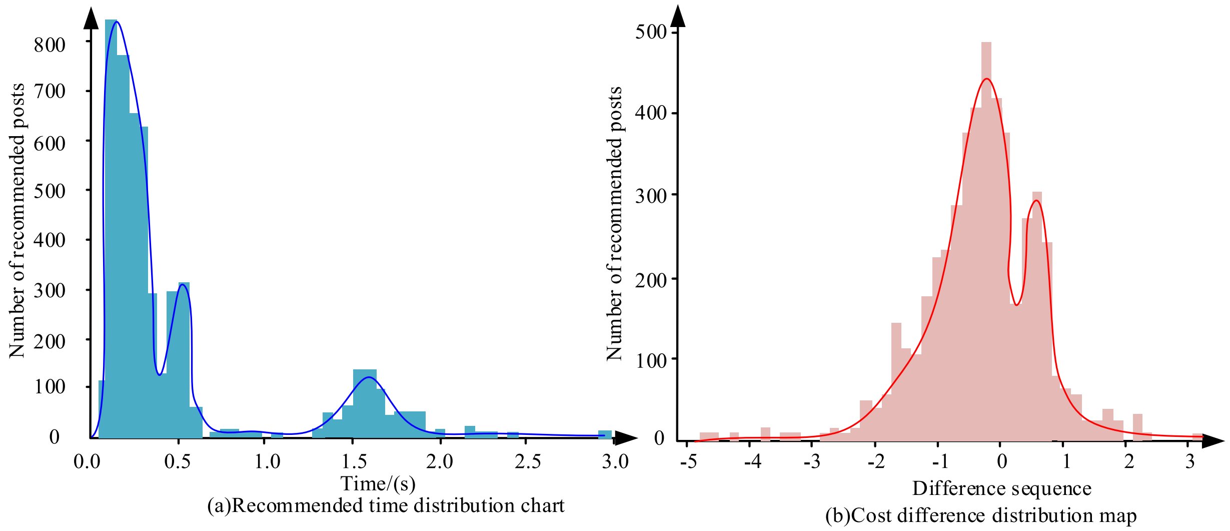
Fang Wang: Writing – original draft, Resources, Formal analysis. **Liuying Yu:** Writing – review & editing, Methodology.



(a) Test 1

(b) Test 2

Fig. 11. Experimental data on click through rate after introducing cities and core businesses.



(a)Recommended time distribution chart

(b)Cost difference distribution map

Fig. 12. Offline simulation test data for recommendation.

Declaration of competing interest

The authors have no relevant financial or non-financial interests to disclose.

Data availability

No data was used for the research described in the article.

References

[1] Y. Guo, Z. Mustafaoglu, D. Koundal, Spam detection using bidirectional transformers and machine learning classifier algorithms, J. Comput. Cognit. Eng. 2 (1) (2022) 5–9.

[2] V. Marotta, Y. Wu, K. Zhang, A. Acquisti, The welfare impact of targeted advertising technologies, Inf. Syst. Res. 33 (1) (2022) 131–151.

[3] W. Rodgers, T. Nguyen, Advertising benefits from ethical artificial intelligence algorithmic purchase decision pathways, J. Bus. Ethics 178 (4) (2022) 1043–1061.

[4] B.R. Gordon, K. Jerath, Z. Katona, S. Narayanan, J. Shin, K.C Wilbur, Inefficiencies in digital advertising markets, J Mark 85 (1) (2021) 7–25.

[5] L. Qi, Q. He, F. Chen, W. Dou, S Wan, Finding all you need: web APIs recommendation in web of things through keywords search, IEEE Trans. Comput. Social Syst. 6 (5) (2019) 1063–1072.

[6] C. Wu, F. Wu, Y. Huang, X. Xie, Personalized news recommendation: methods and challenges, ACM Trans. on Inf. Syst. 41 (1) (2023) 1–50.

[7] C. Li, B. Zhou, W. Lin, Z. Tang, Y. Tang, Y. Zhang, J. Cao, A personalized explainable learner implicit friend recommendation method, Data Sci. Eng. 8 (1) (2023) 23–35.

[8] D. Cai, S. Qian, Q. Fang, J. Hu, C. Xu, User cold-start recommendation via inductive heterogeneous graph neural network, ACM Trans. Inf. Syst. 41 (3) (2023) 1–27.

[9] A.L. Karn, R.K. Karna, B.R. Kondamudi, G. Bagale, D.A. Pustokhin, I.V. Pustokhina, S. Sengan, Customer centric hybrid recommendation system for E-Commerce applications by integrating hybrid sentiment analysis, Elect. Commerce Res. 23 (1) (2023) 279–314.

[10] Z. Wang, J. Chen, F.E. Rosas, T. Zhu, A hypergraph-based framework for personalized recommendations via user preference and dynamics clustering, Expert Syst. Appl. 204 (Oct) (2022) 117552–117565.

[11] Q. Cheng, J. Wang, W. Lu, Y. Bu, Keyword-citation-keyword network: a new perspective of discipline knowledge structure analysis, Scientometrics 124 (3) (2020) 1923–1943.

[12] S. Duari, V. Bhatnagar, Complex network based supervised keyword extractor, Expert Syst. Appl. 140 (Feb) (2020) 112876.

[13] M.M. Madon, S.M Yasin, A comparative study of artificial neural network and genetic algorithm in search engine optimization, J. Soft Comput. Data Mining 4 (1) (2023) 42–52.

[14] D. Sharma, D.C Jinwala, Multi-writer multi-reader conjunctive keyword searchable encryption, Int. J. Inf. Comput. Secur. 15 (2–3) (2021) 141–162.

[15] J. Du, J. Zhou, Y. Lin, W. Zhang, J.L Wei, Secure and verifiable keyword search in multiple clouds, IEEE Syst. J. 16 (2) (2021) 2660–2671.

[16] C.M. Segijn, I. Van Ooijen, Differences in consumer knowledge and perceptions of personalized advertising: comparing online behavioural advertising and synced advertising, J. Market. Communicat. 28 (2) (2022) 207–226.

[17] M. Poongodi, M. Malviya, M. Hamdi, V. Vijayakumar, 5G based Blockchain network for authentic and ethical keyword search engine, IET Commun. 16 (5) (2022) 442–448.

[18] P. Wang, A. Chakravarty, J. Yang, Can emotions be used as keywords for text-based, search-engine advertising? J. Interact. Adv. 21 (3) (2021) 159–172.

[19] W. Gong, C. Lv, Y. Duan, Z. Liu, M. Khosravi, L. Qi, W. Dou, Keywords-driven web APIs group recommendation for automatic app service creation process, Software: Pract. Exp. 51 (11) (2021) 2337–2354.

[20] T. Xiao, D. Han, J. He, K.C. Li, R.F. Mello, Multi-Keyword ranked search based on mapping set matching in cloud ciphertext storage system, Conn. Sci 33 (1) (2021) 95–112.

[21] F. Masood, J. Masood, H. Zahir, K. Driss, N. Mehmood, H. Farooq, Novel approach to evaluate classification algorithms and feature selection filter algorithms using medical data, J. Computat. Cognit. Eng. 2 (1) (2023) 57–67.