IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 19, 2019, accepted August 14, 2019, date of publication August 26, 2019, date of current version September 16, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2937339

MOPSO-Based CNN for Keyword Selection on Google Ads

JING LIANG[1], (Member, IEEE), HAOTIAN YANG[1], (Student Member, IEEE), JIAJIA GAO[2], CAITONG YUE[1], (Student Member, IEEE), SHILEI GE[1], (Student Member, IEEE), AND BOYANG QU[3], (Member, IEEE)

[1]School of Electrical Engineering, Zhengzhou University, Zhengzhou 450001, China
[2]School of Statistics and Big Data, Henan University of Economics and Law, Zhengzhou 450001, China
[3]School of Electric and Information Engineering, Zhongyuan University of Technology, Zhengzhou 450001, China

Corresponding author: Boyang Qu (qby1984@hotmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61876169, Grant 61473266, and Grant 61673404, in part by the Scientific and Technological Project of Henan Province under Grant 152102210153, and in part by the China Postdoctoral Science Foundation under Grant 2017M622373.

ABSTRACT Google Ads is an advertising agency that provides ads to advertisers. Advertisers match the user's search terms and push ads by selecting keywords related to their ad content. Keywords can determine the type of users an advertiser pushes, the effectiveness of the ad promotion, and the sales of the ad product. Automatically selecting keywords that are satisfactory to advertisers from a large number of keywords provided by Google Ads is the main task of this paper. But there is not too much time for the model to judge whether keywords are selected, choosing correct keywords in the shortest time is another task of this paper. Therefore, a structure of the model that can get some useful keywords for advertisers is designed and an improved multi-objective particle swarm optimization algorithm is proposed to achieve this multi-objective task. These are also the main contributions of this paper. To accomplish this multi-objective task, many technical issues need to be overcome, such as the mixed language problem, the imbalance problem, the problem of extracting features from corpora and so on. This paper proposes a corpus selection method to solve the mixed problem of Chinese and English in keywords, word embedding method to solve the representation of keywords, re-sampling to solve data imbalance problem, improved convolutional neural network (CNN) to solve classification problem, and a multi-objective particle swarm optimization algorithm (MOPSO) to achieve neural structure search of CNN so that the effect of the classification is improved and the training time is reduced. The keyword selection problem is solved with the combination of evolutionary computing, deep learning, machine learning, and text processing techniques. Experimental results show that the proposed algorithm greatly improved the accuracy of keyword selection and shortened the time of selecting keywords. Therefore, this algorithm has a good application value.

INDEX TERMS Google Ads, corpus selection, word embedding, re-sampling, CNN, MOPSO, keyword selection, neural structure search.

I. INTRODUCTION

Google Ads is the advertising agency of Google that provides advertising services to advertisers. The previous name was Google Adwords. Google Ads can push advertisers' ads to Google and Google-related sites and some social media. Such as Google, Facebook, and so on. It is profitable through the advertising of advertisers. The form of the advertisement includes text, pictures, and videos. After advertisers promote their ads through Google Ads, Google Ads will quickly respond to advertisers with information of ads about clicks, impressions, cost per click, and more. Later, when the advertisers get the information, some more reasonable solutions are adjusted to increase the competitiveness of the advertisement [1].

Before advertisers push their ads to Google Ads, they need to fill in the keywords related to the content of the ads that need to be served on Google Ads, and the search terms that users search for in the search engine match those keywords. Their match determines the position of the ad on

The associate editor coordinating the review of this article and approving it for publication was Alberto Cano.

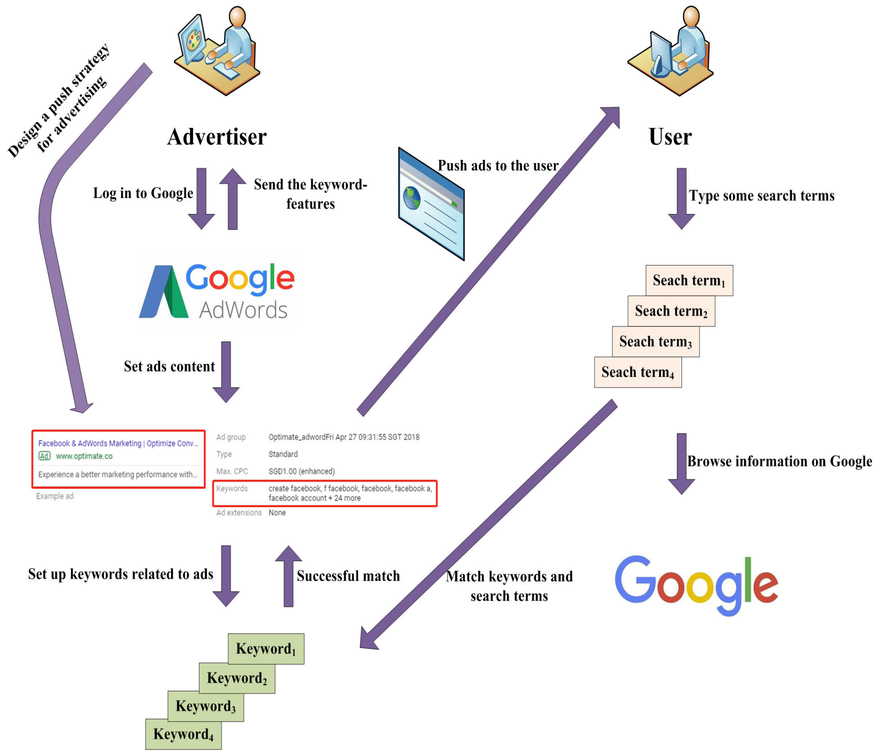


FIGURE 1. The Google Ads workflow chart.

the website that the user is browsing. The degree of matching can be determined by the advertiser's bid and the relevance of keywords and search terms. For advertisers, they push their advertising messages to the public through keywords. On the one hand, they may only want to push ads to users who are related to advertising products. Because if an ad is pushed by keywords to users who are not related to the ad product, even if the user clicks on the ad but is not interested in the ad product, this behavior will waste the advertiser's the cost of clicks. On the other hand, they want to choose efficient keywords. Efficient means that ads pushed out by keywords can get better impressions, clicks, or even purchase of advertisers' products. Therefore, it is not a simple problem to select keywords that satisfy advertisers from among the hundreds of keywords that Google Ads offers to advertisers by keywords planer [2]. In fact, the position ranking of the advertisement is also related to factors such as the advertiser's investment price and the quality of the advertisement content. For the sake of research, this article sets all factors except keywords to be consistent.

The work process of Google Ads is mainly divided into five measures. First, the advertiser sets the content of the advertisement according to their own needs. Second, the advertiser selects keywords on the system according to the advertisement content. Third, the set keywords are matched with the terms searched by the user in the search engine. In the case of a high degree of matching, Google Ads will promote the ads to more relevant users. Fourth, Google Ads will send the keywords-features to the advertiser when the keywords are input the Google Ads. Fifth, the advertisers adjust the strategy according to the features of the keyword. If these keywords are not 'good', they will be removed by advertisers from the Google Ads. The specific process is shown in Fig. 1.

As mentioned above, keywords have their features. Some keyword-features are owned before the keyword is put into Google Ads. Such as the matching degree. The matching degree can be divided into four categories. The specific details are shown in Fig. 2, namely broad, broad match modifier, phrase, exact. The other keyword-features are generated after the keyword is put into the Google Ads. These features are sent by Google Ads to advertisers and used to analyze the effect of these keywords. The features of keywords mainly include cost per click (CPC), cost per 1000 impressions (CPM), page rank, quality score and so on. Advertisers can select or set the keywords according to these features.

Different advertisers have different advertising needs. When a keyword is put into Google Ads for a while, Google Ads will send back the keyword-features for each keyword. When the advertiser receives these keyword-features, they will choose and select the keywords that match their needs. Some keywords that do not perform well will be removed from Google Ads. For example, an advertiser puts a keyword into Google Ads, and one day later, Google Ads will send a series of keyword-features such as CPC, CPM, Page Rank, and Quality Score to the advertiser [3]. The advertiser thinks that CPC and CPM are the most important standard for their

Reach

Match Type	Symbols	Matches to:	Example Keyword	Matches to:
Broad	example keyword	Misspellings, synonyms, related searches, close variations	Sell house	Sale home
Broad Match Modifier	+example +keyword	Search terms that include all words preceded by '+', or a close variant	+Sell +house	Sell china house
Phrase	"example keyword"	Search terms that include that keyword phrase without any words in between, or a close variant	"Sell house"	Sell house in singapore
Exact	[example keyword]	Search terms are only that keyword, or a close variant	[Sell house]	Sell house

Relevance

FIGURE 2. **The match types of keywords.**

ads, and the rest of the features are negligible. And keywords with CPC and CPM meet the standard will be retained, and keywords with failed CPC and CPM metrics will be removed. Therefore, whether keywords are retained depends on the different goals and needs of the advertiser. In this paper, the keywords that can be retained by the advertiser are used as positive samples in the training data, and the keywords removed by the advertiser are used as negative samples in the training data.

However, the features of a keyword require that the keyword is put into Google Ads for a period of time before they can be generated. And each keyword takes a lot of money to be put into Google Ads. For advertisers in different industries, Google Ads provides related keywords to this industry. However, there are too many keywords available for advertisers to choose from. If they put each of the available keywords for a period of time, then choose the available keywords based on the keyword features. It will spend a lot of money and time. Therefore, how to find the keywords available to advertisers from the huge selection of keywords is one of the core tasks of this paper.

There is a large percentage of references to analyze how to select high-quality keywords on Google's advertising pages. Among them, the most representative ones are related research methods in the economic field, such as economic analysis of advertising content structure [4] and economic analysis of bid selection [5]. However, because algorithms are required to get the best results in the shortest time when solving engineering problems, few researchers use deep learning to select the keywords on Google Ads. Therefore, getting the best results in a short time is another core task of this paper.

But achieving fast keyword selection is not a simple task. So this paper proposes some solution to the following problems and cleverly solves the keyword selection problem.

1) How to choose keywords if there are few data for studying? In fact, it is the problem of the short text classification [6]. However, in reality, due to factors such as the cost of advertising and the number of vocabularies, advertisers can not directly access a large number of keywords, which results in keyword classification not being available in one step. In order to optimize this problem, this paper proposes the word embedding method [7] to better pick up the features of these keywords.
2) A keyword may contain both Chinese and English words. The keyword language types in the research case are mixed, and the most common one is the intersection of English keywords and Chinese keywords, so the corpus needs to be carefully selected.
3) The number of keywords that meet the needs of advertisers and the number of keywords that do not meet the needs of advertisers is not balanced. When an advertiser chooses keywords, the possibility of selecting high-quality keywords is small, this is because there are fewer keywords with good quality and top-ranking features. In order to improve this indicator, an unbalanced data set needs to be resampled [8]. There are usually two ways to resample, one by creating a composite example from a few categories (over-sampling) and the other by dropping the original example from most categories (under-sampling). In this study, the two methods are mainly used to improve the data imbalance, and the best method is selected for the following experimental process [9].
4) The performances of traditional classifiers are not satisfactory. Many experts have studied and elaborated on the classification of short texts [6], [10]–[13], but few experts have studied how to choose keywords. In practical applications, it is very difficult to extract the correct characteristics of the keywords and quickly make the classification of the keywords high. To optimize this problem, this paper proposes improved convolutional neural networks. Convolutional Neural Networks (CNN) are widely used in various fields due to their advantages in feature extraction [14], which can effectively solve text classification problems.
5) Using deep learning will take a lot of time. In fact, if a more complex network structure is encountered in practical applications, most of the parameters need

to be manually adjusted to improve the classification level. And the complex structure also takes lots of time. In response to this problem, this paper proposes a multi-objective particle swarm optimization algorithm (MOPSO) to optimize CNN. The MOPSO algorithm can optimize the network structure of the convolutional neural network to improve classification and reduce training time [15].

For the selection of keywords on Google Ads, most researchers choose to use economic methods to analyze and solve. This paper explores from other perspectives. Combining evolutionary computation with deep learning and assisting with some traditional text and data processing methods greatly improves the accuracy of classification results and reduces time, finally completes the problems in engineering applications. For advertisers, it will reduce a lot of time and money, and the economic benefits will be huge.

II. RELATED WORK

A. KEYWORD SELECTION ON GOOGLE ADS

In this paper, keyword selection is a problem with the classification that some suitable keywords for advertisers can be selected from many recommended keywords. Because the number of words in a keyword is small, many experts use short text classification methods to solve this problem [6], [10]–[13].

In addition, when the application is keyword selection on google ads (formerly Google AdWords), the short text classification method may no longer be applicable due to many other factors that need to be considered. In order to solve the keyword selection problem of google ads, Lu *et al.* proposed to solve keyword selection by Analyzing the market [16], Desai *et al.* proposed to analyze competitor's keywords [5], Mostafa used Term Frequency to extract keywords [17].

However, in this paper, some of the keywords that the company previously-stored and selected and satisfied by past advertisers are used as the train test, and the methods of machine learning and deep learning are used to solve the problem of keyword selection. But deep learning is considered by the author to waste time and not achieve good effectiveness. Therefore, using MOPSO to design a network can make the network get the best results in the shortest time. This is also one of the main contributions of this article.

B. MULTI-OBJECTIVE PSO

The particle swarm algorithm (PSO) is based on the principle of bird flocking. PSO is proposed by Kennedy in 1995 [18]. The idea of the algorithm is to first randomly initialize all the particles whose size is N and dimension is D. In the k^{th} iteration, the position of the i^{th} particle is $x_i = (x_i, 1, x_i, 2, \ldots, x_i, D)$, the particle updates its position by its personal best (*pbest*) and global best (*gbest*) [19]. Finally, the optimal solution is output if the stopping condition is met. Through continuous iteration and updating, the optimal solution is finally found. The updating process of particles is mainly based on the following two formulas:

$$x_{i,d}^{k+1} = x_{i,d}^{k} + v_{i,d}^{k+1} \tag{1}$$

where $x_{i,d}^{k}$ is the position of the i^{th} particle at k^{th} generation and d^{th} dimension; $v_{i,d}^{k}$ is the velocity of the i^{th} particle at k^{th} generation and d^{th} dimension. The velocity of the i^{th} particle is updated according to:

$$v_{i,d}^{k+1} = wv_{i,d}^{k} + c_1 r_1 (pbest_{i,d}^{k} - x_{i,d}^{k}) + c_2 r_2 (gbest_{d}^{k} - x_{i,d}^{k}) \tag{2}$$

where w is inertia weight, c_1 and c_2 are parameters to control the trade-off between *pbest* and *gbest*, r_1 and r_2 are random numbers between 0 and 1 [20].

However, in practical applications, there is often more than one objective. Therefore, after a period of development, The first MOPSO was proposed by Moore in 1999 [21]. For the multi-objective optimization algorithm, just as there is no free lunch in the world, it is necessary to analyze according to the specific situation. Firstly, be clear about what the objective of the problem is. Secondly, need to know if there is a dominant relationship between the various objectives. Finally, when multiple suitable solutions emerge, how to choose the most appropriate solution for all problem. Currently, various evolutionary algorithms are proposed in the field of multi-objective optimization such as NSGA, SPEA, NSGA II, AMALGAM, SMPSO, and CMOPSO, which will be discussed in the succeeding sections [22]. For this paper, MOPSO is also improved for this application, the details are introduced in PROPOSED METHOD.

C. EVOLUTION ALGORITHM BASED CNN

Deep learning models have achieved remarkable results in computer vision and NLP in recent years [23], [24]. In deep learning, convolutional neural networks (CNN) can not only share convolution kernels to process high-dimensional data but also automatically extract features and train weights to obtain better classification results. In addition, For CNN, the more training data is input, the better the features extracted by the model, and the higher the accuracy of the classification. In this application, the keyword information used by the advertiser each time will be stored. The information for these keywords includes which keywords are satisfactory to the advertiser and which are not satisfactory to the advertiser. So as time goes on, data will be accumulated more and more. The effect of CNN being used to select keywords will get better and better. Therefore, CNN has great potential, so CNN is used as a classifier for this application.

CNN consists of convolutional layers, pooling layers, activation layers, and fully connected layers. In each layer, there are some hyperparameters, as shown in [25]. In the natural language processing problem, CNN is constantly being improved. In 2014, Yoon Kim *et al.* proposed the CNN improved to achieve sentence classification [14]. In 2018, Yijing Li *et al.* proposed the CNN improved to solve the imbalanced text sentiment classification [8]. In 2018, Ying Shen *et al*. proposed the CNN improved to solve medical short text classification [6].

But there are still defects in CNN, such as too many parameters, long training time and over-fitting. Recently, more and more researchers started to use evolutionary computing to tackle these problems. For example, Dahou *et al*. used CNN and evolution algorithm (EA) for sentiment classification [26], Sun *et al*. proposed the PSO based flexible convolution autoencoder for image classification [27], which is capable of automatically searching for the optimal architectures of the proposed FCAE with much less computational resource and without any manual intervention. Giovanni *et al*. proposed a methodology that uses a CNN in conjunction with the particle swarm optimization to reduce lung nodule false positive on computed tomography scans [15]. Using evolutionary computing to improve CNN is becoming more recognized in deep learning solutions. And Zhang *et al*. proposed that using PSO evolve image classification architectures [28]. It can be seen from the results of these papers that PSO can optimize the CNN network structure to achieve better results.

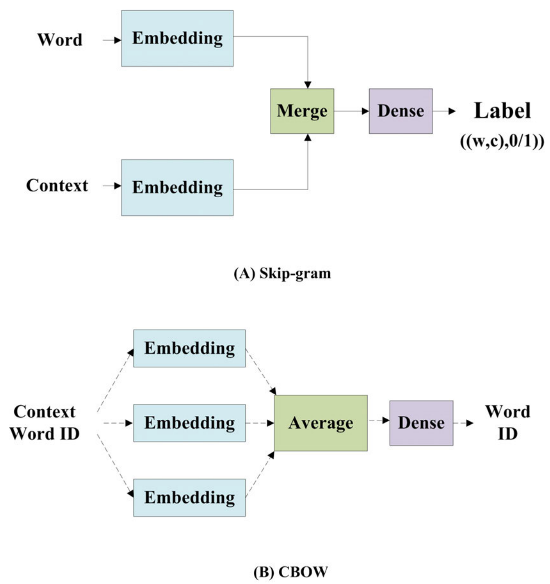


FIGURE 3. Skip-gram and CBOW Word2Vec models.

D. WORD EMBEDDING

When a model is built, the natural language cannot be directly imported into the model, because the input to the model can only be numeric type data. So the concept of word embedding was introduced. Word embedding is a way to convert words into word vectors. Word embedding is used to express natural language in the form of numbers. In this way, some language problems can be studied. Currently, word embedding has become a common method in dealing with natural language processing problems. It can be used in text classification, document clustering, speech tagging, entity identification, emotion analysis, and many other natural language processing tasks. In word embedding, Word2Vec and GloVe are the most popular methods [29], [30]. In addition, in recent years, FastText launched by Facebook has been widely used as a word embedding tool because of its fast speed and good performance [31].

For the three most commonly used word embedding methods, their principles are roughly the same, so only a detailed introduction to Word2Vec is given here. As is shown in Fig. 3, there are two types in the Word2Vec: Skip-gram and Continuous Bag of Words (CBOW). Both of the structures have advantages. The training speed of CBOW model is faster and the skip-gram model is better in predicting infrequent vocabulary [32]. So the skip-gram model is more suitable for obtaining pre-trained word embedding in this study. Google proposes the tool of Word2Vec in 2013. This tool adopts two main model architectures based on neural network, continuous bag-of-words (CBOW) model and continuous skip-gram model, to learn the vector representations of words [29]. The CBOW architecture predicts the current word based on the context, and the skip-gram predicts surrounding words given the current word [18]. The algorithm is described in detail as shown in [33].

E. CLASS IMBALANCE ISSUE

In practical applications, most of the data is not balanced. For example, the proportion of illness and health in the population, the proportion of damaged and non-damaged products in the factory, and so on. For the classification of such data imbalance, if the data is not processed, the classification effect is generally not particularly good. Therefore, there are currently many ways to deal with data imbalance problems. These methods can be roughly divided into two types. Depending on the type of data, if the number of data is large, under-sampling the majority class can be used to balance the data [34]. In this way, data diversity may be reduced. Another approach is oversampling the minority class to make the data balanced. But this approach may lead to the overfitting problem due to poor configurations [35]. There are some excellent ways to increase the number of data [36], such as SMOTE, and Borderline-SMOTE [9], [37].

F. EVALUATION CRITERIA

When the data is imbalanced, the classification effect cannot be evaluated only by the accuracy rate. For example, if there are 10 samples, 8 are correct, 2 are wrong, then accuracy is 0.8, so this is a very intuitive evaluation method, but this is valid only when the data is well balanced. In the case of imbalanced data, if there are 10 samples, 9 positive examples, and 1 negative case, then if the classifier classifies everything to be positive, the accuracy will still be as high as 0.9. In the imbalanced data test scenario, there is no doubt that the classifier has no effect. In this situation, using accuracy as evaluation criteria will not be sufficient.

In this case, the recall rate, precision rate, accuracy rate, receiver operating characteristic (ROC), and area under the curve (AUC) are often used as performance metrics to assess the performance of imbalanced data classification [38].

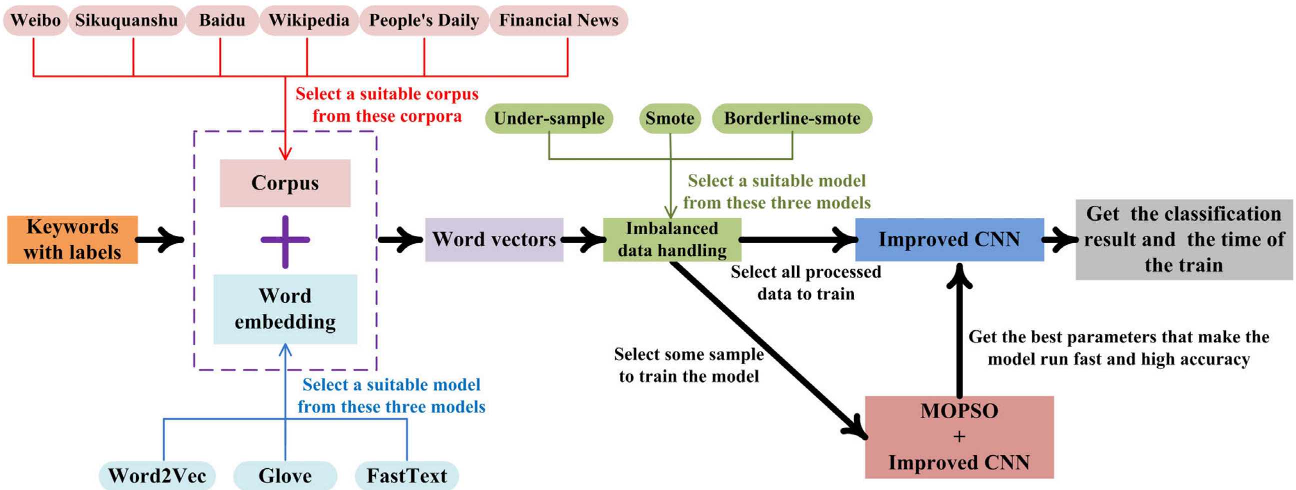


FIGURE 4. The flowchart of the algorithm.

In (3), recall rate represents the true positive samples divided by the sum of the true positive samples and the false negative samples.

$$RECALL = \frac{TP}{TP + FN} \tag{3}$$

In (4), precision represents the true positive samples divided by the number of positive samples predicted.

$$PRECISION = \frac{TP}{TP + FP} \tag{4}$$

In (5), accuracy (ACC) represents the correct sample divided by all samples.

$$ACC = \frac{TP + TN}{TP + TN + FN + FP} \tag{5}$$

The area AUC (Area Under ROC Curve) value below the ROC curve is used to measure the algorithm performance: if the model is perfect, then its AUC = 1, if the model is a simple random guess model, then its AUC = 0.5, if one model is better than the other, the area under the curve is relatively large [39].

III. PROPOSED METHOD

The challenges of the keyword selection in Google Ads are listed as follows.

1) The number of keywords in an ad cannot be too big due to cost constraints.
2) There could be a mixture of different language in the keywords.
3) There is the imbalance issue between 'good' and 'bad' keywords.
4) The effect of the typical keyword classification approach [8], [15], [40] can not produce a satisfactory result.
5) The keywords selection method need to be not only of high accuracy but also of high speed.
6) There are multiple objectives to be optimized.

To deal with the above challenges, a combination method is proposed. This method is divided into five steps. 1. Using the word embedding to get the word vector so to improve the number of keywords features. 2. Selecting the corpus to deal with the language blending question in the keyword. 3. Using resampling to balance the data. 4. Proposing an algorithm for improving CNN. 5. Optimizing the CNN network structure by an improved MOPSO.

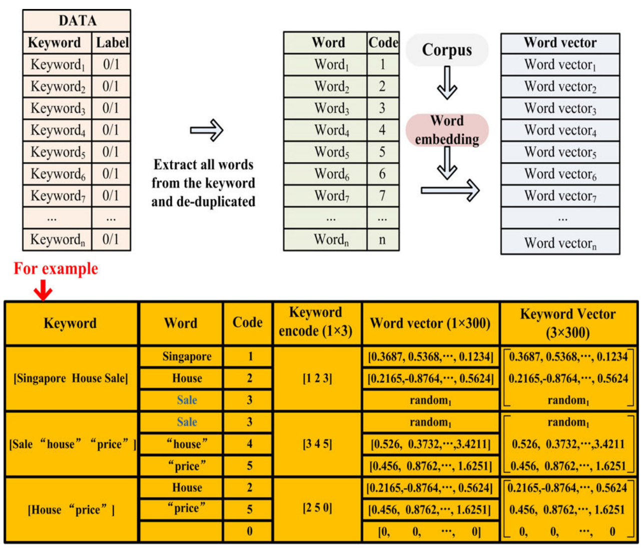
The flow chart of the proposed method is shown in Fig. 4 and the details of each step are introduced in the following subsections.

A. THE USE OF WORD EMBEDDING

In the problem of natural language processing, it is often the problem of processing multiple articles or multiple sentences. However, this study processes keywords which are difficult to deal with. First, each keyword only contains a few words, at most 15 words, usually only about 5. Second, because of cost issues, the number of available keywords for training is far less than the number of texts in other natural language processing problems. Therefore, methods in traditional natural language processing, such as Word Frequency, DF, TF-IDF and so on, are not applicable. So this paper selects the popular word embedding method in recent years to convert keywords into word vectors, and it is introduced in related work [33]. In this way, not only the keyword feature can be added, but also the features of the keyword can be extracted. It should be mentioned that the most commonly used methods for word embedding are Word2Vec, Glove, and Fasttext [31]. Different word embedding methods are selected according to different application needs. For the convenience of comparison, this paper compares these three methods and chooses the best word embedding method to prepare for the next experiment.

B. THE SELECTION OF CORPUS

The keywords in Google Ads contain the language of more than one country. In order to solve the problem that different languages exist in the same corpus. This paper turns these keywords into word vectors by selecting corpora which contain different national languages and using the above word embedding methods. In this way, even if a keyword contains several different languages, they can be put together to extract features. There are Chinese and English in a keyword used in this paper, so some corpora containing both languages are searched, such as Sikuquanshu, Weibo, Baidu, Wikipedia and so on. In addition, the task of this paper is about the issue of advertising keywords, so these corpora are required to contain enough advertising information. However, using other corpora to convert words of a keyword into word vectors may result in that the words of keywords are not in these corpora. In this case, the word vectors need to be represented by the random value. It should be noted that the number of words contained in the keyword is not uniform, so the length needs to be set to a uniform length. If the length does not reach the set length, 0 can be used to fill it up. This problem is supplemented in the improved CNN described below. In addition, different applications need to choose different corpora. This paper compares and selects the best corpus to prepare for the next experiment. The flow chart converting keywords into keywords vectors is shown in Fig. 5.



Note: A. The length of the keyword encoding should be consistent. For keywords that are not long enough, 0 is used to fill as the code and its word vector is also 0.
B. If the word is not in the corpus, the word vector can be randomly generated. Such as $random_1$.
C. The English word in " " indicates the language of other countries.

FIGURE 5. The flow that converts keywords into keyword vectors.

C. THE PROCESS OF THE IMBALANCED DATA

As mentioned in the introduction, the number of keywords that can satisfy the needs of advertisers is very small. However, the keywords that Google Ads offers to advertisers are large. Direct classification, in this case, will inevitably result in the poor effect due to the data imbalance. So this paper uses resampling to improve this problem. The under-sampling [8], smote [9] and borderline-smote [37] are the three most widely used problems to solve this situation. By experimenting with these three methods separately, the best method is used as the method of resampling.

D. THE IMPROVED CNN

Convolutional neural network is widely used to deal with natural language processing problems because of its ability to automatically train network and extract features, so this paper chooses CNN to select keywords. However, the keywords studied in this paper not only need to extract the characteristics of their own language, but also need to add some related features of the keywords in the network, so it is necessary to improve the traditional CNN.

After the description of the above three steps, this subsection is described how to select keywords in detail.

Firstly, because the length of the keywords is inconsistent, the length of the keyword needs to be limited here. The length of the keyword needs to be set according to the actual application. Because the keyword length in this application is up to 15, the keyword length is uniformly set to 15. When the length of the keyword is less than 15, it is filled with 0.

Moreover, after the length limit of the keyword is completed, each word in the keyword needs to be encoded. To complete the encoding of the keywords, all the words contained in all the keywords need to be collected and then de-duplicated, and then the different words are encoded from the one, such as 1, 2, 3, 4, 5···. In this way, each keyword can be converted into a vector of length 15 that is represented by different codes.

Next, the method of word embedding will be used. In the operation of the previous step, different code numbers correspond to different words. Then, after selecting the corpus and using the word embedding, each code can be converted into a word vector. The word embedding method is used to convert each code into a 300-dimensional word vector. There is a possibility that the corpus does not contain words in the keyword, and in this case, the word is converted into a random 300-dimensional word vector. Therefore, after using the word vector method, each keyword is converted into a matrix of 15×300.

The previous three steps are shown in Fig. 5.

Then, this matrix can be used as the input of the improved CNN. It should be noted that the word vector of the word is a word vector based on the corpus, and the corpus is not related to the application. Therefore, the purpose of referencing CNN is to get the model of the keyword selection according to the continuous training of the weight of each neural layer. And this model can select the appropriate keywords for the advertiser. There are two points to be proposed. 1. All neural layers of CNN are randomly initialized. 2. Before the final classification, adding the keyword-related features together with the features extracted by the CNN for the final softmax layer. Because keywords contain some keyword-features that advertisers value, in this study, the selected keyword features are cost per click (CPC), cost per thousand impressions (CPM), and the matching degree mentioned in the introduction. The reason why these three features are selected is that

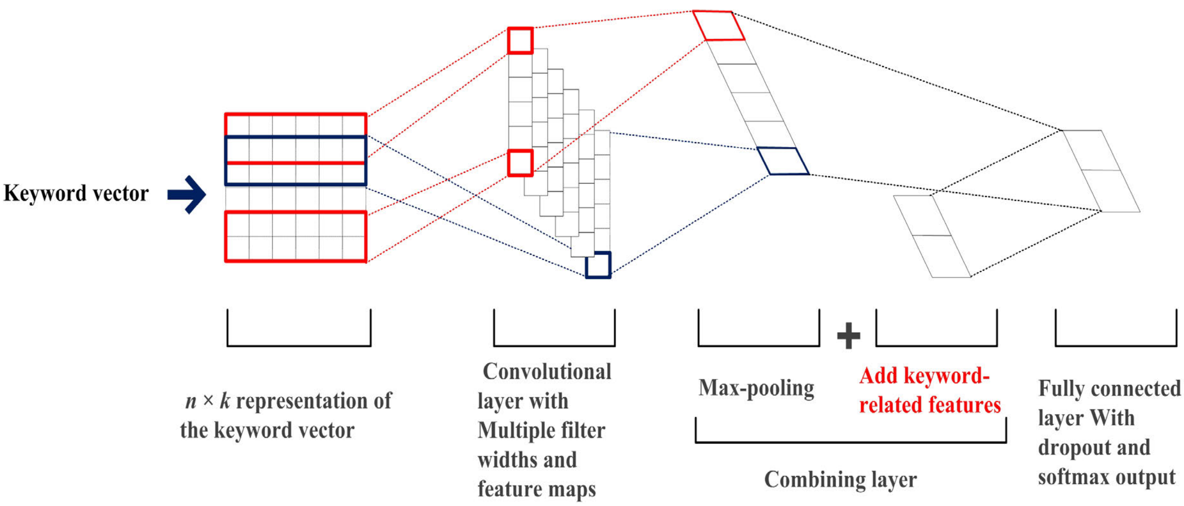


FIGURE 6. The basic structure of the proposed convolutional neural network.

these features can be obtained quickly, without having to wait for a long time, and advertisers prefer these features to judge the quality of the keywords [16]. The basic structure of CNN is shown in Fig. 6. The specific parameters and layer selection methods will be described in the next step.

E. THE MOPSO BASED IMPROVED CNN

With the improved CNN described above, the selection of keywords can be done. However, when CNN is used for keyword selection, it has a complex network structure with many parameters. How to choose the network structure that can get keywords faster and more accurately from many parameters is the key problem to be solved. Therefore, MOPSO is used to optimize the network.

In this subsection, how to optimize CNN with MOPSO is introduced.

The traditional MOPSO algorithm is improved according to this application. There are two objectives in this application, and there are conflicting relationships between the two objectives. In this application, the effect of the classification is the first objective and the training time of the model is the second objective, and the first objective is more important than the second objective. As shown in Fig. 7, the improved MOPSO needs to make the following improvements: 1. Before the criteria are met, all the particles in the Pareto front need to be obtained and sorted according to the first objective. The middle particle is selected as the best particle (knee point) in the Pareto front. The middle particle will be introduced later (**MO selection**). 2. After the number of iterations is reached and the range of the first objective is specified, the best particle in the second objective that satisfies the first objective range is found in the front. The best particle is used as the parameters of the CNN (**Final MO selection**).

As shown in Fig. 7A, the **MO selection** method in this paper is to find the non-dominated solution first. Then the

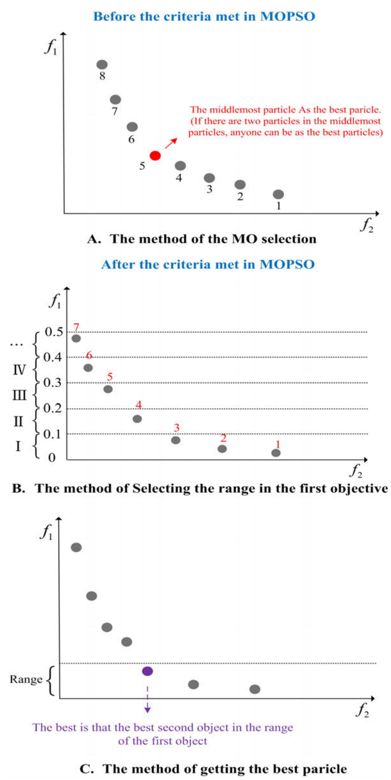


FIGURE 7. The steps of selecting the best particle.

particles in the non-dominated solution are sorted in the first object and the middle particle as the best particle. If there are two particles in the middlemost particles, anyone can be as the best particle. In this way, the best particle can be found quickly and the method is easy, which is very suitable for this application. As shown in Fig. 7B and Fig. 7C, **Final MO selection** is to get the best particle from the Pareto front when the termination condition is satisfied. the first objective (f_1) is divided by 10 parts, such as I, II, III and so on. Of course, this number of parts depends on the requirements of applications. The higher the requirement for the first objective, the more parts should be selected. In this application, it is necessary to select the particle that the train time is shortest in the error rate within 10%, and 10% is the maximum tolerance in this application. If the best particle in the f_1 is particle$_1$, the I will be selected as the range. If the best particle in the f_1 is particle$_5$, the III will be selected as the range. For example, when the criteria are met, there are some particles in the Pareto front. If the best-classified error of the particles is 1%, then the error between 1% and 10% is the range of the first object. After the range is set, there is a particle whose train time is the shortest in the range and this particle is the best one as the input of the CNN.

As shown in **Algorithm 1**, firstly, the particles x are randomly generated in the feasible area. ***GHA*** and ***PHA*** are respectively as the global best (***gbest***) historical archive and the personal best (***pbest***) historical archive. Then the ***gbest*** is selected from ***GHA*** and the personal best particle in the ith generation ($\boldsymbol{pbest_i}$) is selected from $\boldsymbol{PHA_i}$ by the **MO selection** method. Moreover, the $\boldsymbol{x}$ is updated according to (1) (2). Next, $\boldsymbol{PHA_i}$ and ***GHA*** are updated. Particles in $\boldsymbol{PHA_i}$ and ***GHA*** are firstly sorted according to the non-dominated relationship. If the number of non-dominated solutions is larger than the predefined number, **MO selection** method is employed to remove extra solutions [20]. In addition, in this paper, the terminal criterion is the number of iterations. when the iteration reaches the set number of times, **Final MO selection** is used to get the best particle and the best parameter set from the best particle will be sent to the CNN.

After MOPSO is introduced, MOPSO is used to optimize CNN. As shown in Fig. 8, the improved CNN includes two convolution layers (C1, C2), two pooling layers (P1, P2), two activation layers (A1, A2), a keyword feature layer (K1), and a dropout layer (D1).) and a fully connected layer (F1). The order of the network structure is C1, A1 (RELU), P1, C2, A2 (RELU), P2+K1, D1, F1. The fully connected layer includes an input layer, a hidden layer, a RELU activation layer, the dropout layer and the output layer containing softmax activation. The kernel size of the pooling layer is 3×1, and the type of the pooling layer is VALID. How to select the parameters will be introduced in the MOPSO.

As shown in TABLE 1, a_1, a_2, a_3, a_4, a_5 as features of a particle in the MOPSO. These particles are used to represent the structures in CNN, a_1 is the number of filters in C1, a_2 is the number of filters in C2, a_3 is the number of neurons in the hidden layer, a_4 is the probability of dropout in convolutional layer and a_5 is the probability of dropout in the fully connected layer.

These particles will be randomly initialized when the PSO algorithm is executed. Among them, a_1, a_2 and a_3 are between 20 and 150, a_4 and a_5 are between 0.1 and 0.99.

The importance of AUC is higher than the importance of ACC when it is necessary to process imbalances. Therefore, the weight of the AUC should be set higher in the fitness

Algorithm 1 MOPSO

```
Step 1: Initialize
  Randomly generate x in feasible region
  GHA = x
  PHA_i = x_i
Step 2: Select gbest and pbest_i
  gbest = MO_selection(GHA)
  pbest_i = MO_selection(PHA_i)
Step 3: Update x_i(t) according to (1) and (2)
Step 4: Update PHA_i and GHA
 For i=1:popsize
   PHA_i(t) =non-dominated-selection(PHA_i(t-1),
     x_i(t))
     if size (PHA_i(t))>maximal number of PHA
       Remove the extra particles from PHA_i(t)
     end
   GHA(t) =non-dominated-selection(PHA(t))
     if size (GHA(t))>population size
       Remove the extra particles from GHA(t)
     end
if the termination condition is satisfied
 Terminate
or
  Repeat Step 2-5
end
Step 6: Select the optimal solution
  Output the best particle in GHA by
   Final_MO_selection
```

TABLE 1. Particle definition.

Particle coordinate	Hyperparameter
a_1	Number of filters in C1
a_2	Number of filters in C2
a_3	Number of neurons in the hidden layer
a_4	Probability of dropout in the convolutional layer
a_5	Probability of dropout in the fully connected layer

of the particles. Equations 6 and 7 are two proposed fitness functions. For better comparison. The two adaptive equations need to be normalized.

$$Fitness_1 = 1 - (2 * AUC + ACC) \tag{6}$$

$$Fitness_2 = Time \tag{7}$$

These particle's position is updated by its personal best (***pbest***) and global best (***gbest***). When the number of iterations reaches the set times, the optimal parameters are passed into the improved CNN to get the optimal results.

IV. EXPERIMENT RESULTS

In this section, the experimental result of each step is demonstrated. In addition, the specific parameter settings of the algorithm are also described in detail in this section. In order to prove the reliability of the algorithm, all experiment are the results of 10 cross-validations.

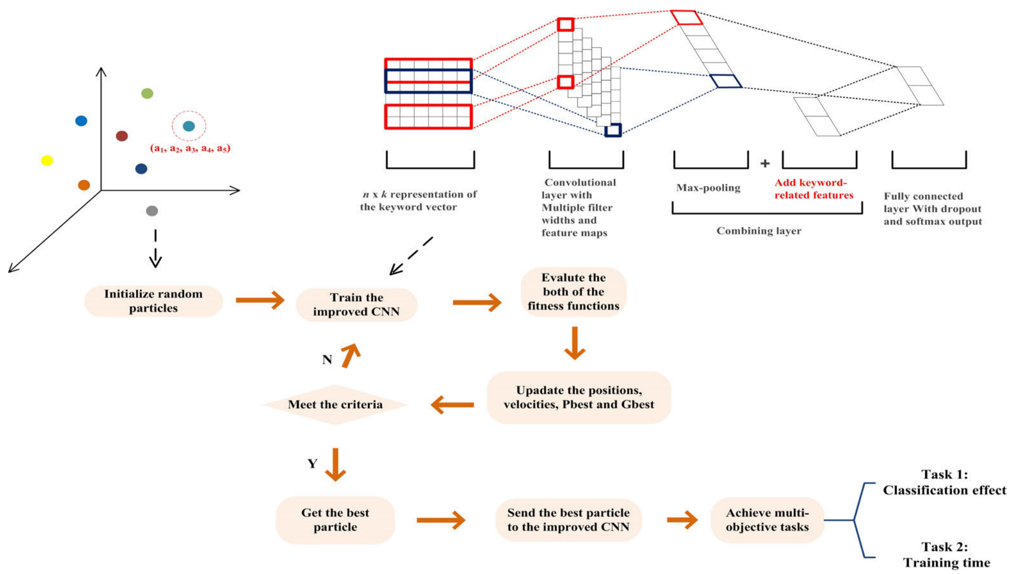


FIGURE 8. The flowchart that MOPSO optimize CNN.

A. INTRODUCTION OF THE DATA

The characteristics of the data sets are shown in TABLE 2. Advertiser ID refers to the different ad types. Positive refers to the number of keywords in the Google Ads that meet the advertiser's requirements. Negative refers to the number of keywords that do not meet the advertiser's requirements. The last two indicates are the number of Chinese and English keywords contained in the keywords.

TABLE 2. Information on the data.

Advertiser ID	Positive	Negative	Total Number	Number of keywords (English)	Number of keywords (Chinese)
Real-estate$_1$	4520	12620	17140	7110	10030
Real-estate$_2$	3210	5510	8720	0	8720
Retail$_1$	200	350	550	550	0
Retail$_2$	470	184800	264100	76600	187500
Retail$_3$	2250	3410	5660	5610	50
Retail$_4$	2000	2640	4640	4620	20
Retail$_5$	1120	2310	3430	3310	120

After the keywords are input in Google Ads, a lot of money and time will be spent. The following seven sets of data are data results that take 6 months and approximately \$10,000. Therefore, these actual data are very precious. The data of Real-estate1 is the most complete and comprehensive, with a large amount of data and representative, so in the next experiment, the Real-estate1 is selected as the experimental data.

B. SELECT CORPUS AND WORD EMBEDDING TYPE

Corpora are often used with word embedding. Because the type of this research is about keyword selection for advertising, there are both Chinese and English in the keywords and the keywords are related to the advertisement. Therefore, corpora are required to have both Chinese and English and contain advertising information. The corpora that meet these requirements include Weibo, Wikipedia, Baidu, Financial Times, People's Daily, and Sikuquanshu Corpus. The most popular three word embedding methods are Word2Vec, Glove, Fasttext. Because the classifier is CNN in this paper, in this part of the experiment, CNN used by Wang *et al.* is selected [41]. Because it is simple and quick to get results so that the word embedding can be checked quickly. The data are not balanced, so AUC is used as the evaluation standard. As shown in TABLE 3, the Weibo corpus and Word2Vec method can get the best results, so this combination is used as the keyword processing solution for the next experiment.

TABLE 3. Experimental results of word embedding combined with the corpus (AUC).

Word Embedding / Corpora	Word2Vec	Glove	FastText
Weibo	81.25%	79.68%	80.32%
Wikipedia	69.24%	78.42%	73.35%
Baidu	76.54%	76.35%	78.25%
Financial News	73.51%	74.38%	77.24%
People's Daily	68.23%	78.32%	62.54%
Sikuquanshu	79.38%	78.33%	72.33%

C. IMBALANCED DATA HANDLING

For imbalanced data sets, if there is not imbalanced data handling, the classification results may not be very effective. Therefore, this section the three most common methods for balancing datasets are used, namely Under-sample, Smote, and Borderline-smote. In this part, by combining the best corpus in the above experiment with the word embedding method, the suitable method of imbalanced data handling for this application will be selected by conducting the experiment. The CNN of the upper experiment is still used for this classification. AUC is used as an evaluation indicator for the experiment. As shown in TABLE 4, it can be seen from the experimental results that the effect of the Smote is the best. Therefore, the Smote method is used as a solution for balancing data in the next part of the experiment.

TABLE 4. Experimental results of re-sample (AUC).

Re-sample \ Method	Word embedding (Word2Vec) + Corpus(Weibo) + CNN
Under-sample	73.56%
Smote	84.34%
Borderline-smote	79.91%

D. MOPSO AND IMPROVED CNN

1) THE SETTING OF THE MOPSO AND IMPROVED CNN

In this section, the settings of MOPSO and CNN are described in detail.

In MOPSO, the inertia weight w is set to 0.7298, c_1 and c_2 are all set to 2.05, r_1 and r_2 are random numbers between 0 and 1 according to [42]. The maximum number of iterations is set to 30, the swarm size is set to 10 particles. The maximal number of GBA is equal to the population size and the maximal number of PBA is one-fifth of the population size [20].

In the improved CNN, the method proposed by Kim [14] is used as the model architecture. The word vector of a word is 300. The length of a keyword's code is 15. The type of pooling is max pooling. 3 keyword features are added to the combining layer to train with the features after the pooling layer, and the activation function is Relu. Softmax is used to classify in the last layer. The maximal iteration number of improved CNN is 1000 [14].

2) THE RESULT OF MOPSO-BASED CNN

According to the above three experiments, the best treatment plan is obtained. The Weibo corpus, the word embedding method by Word2Vec and Smote as the method of balancing the data set are selected. The improved CNN algorithm optimized by MOPSO proposed in this paper is used to test. In this part of the experiment, CNN is compared with MOPSO based CNN, and the CNN in TABLE 5 is the CNN proposed in this paper. As shown in TABLE 5, the AUC of using MOPSO is better and the time spent is shorter. Among these clients, the effect has a significant increase by using MOPSO in the Real-estate1, but the effect is not obvious in Retail1. It can be seen from the experimental data of the previous part that the data information of Real-estate1 is the most complete, and Retail1's data information is very little, so it can be concluded that the completeness of the experimental data information can also affect the experimental results. Overall, all of the clients' data, the classification effect and the time spent, both of which have made progress in using MOPSO, it can be proved that using MOPSO is helpful.

TABLE 5. Experimental results before and after using MOPSO.

Client \ Method	CNN (AUC)	Time spend by CNN(h)	MOPSO-based CNN (AUC)	Time spend MOPSO based CNN(h)
Real-estate$_1$	73.32%	87.24	90.14%	67.36
Real-estate$_2$	67.25%	62.53	83.42%	54.65
Retail$_1$	51.22%	4.47	53.46%	4.45
Retail$_2$	54.36%	25.34	55.88%	21.29
Retail$_3$	65.67%	56.32	78.49%	50.45
Retail$_4$	59.98	67.42	75.39%	47.29
Retail$_5$	63.24%	61.85	70.28%	58.45

3) COMPARISON WITH OTHER RELATED WORKS

In this section, the proposed algorithm is compared with another state of the art algorithms proposed in recent years. Among them, da Silva *et al.* [15] proposed the method that after CNN is optimized by PSO, then it is used to solve the problem of the imbalanced classification. Li *et al.* [8] proposed that the improved CNN is processed the problem of the imbalanced text sentiment classification. Yang and Eisenstein [40] proposed that improved CNN solve the issue of sentiment analysis by embedding. These three methods have an excellent effect in dealing with natural language processing, and some of the techniques they use are similar to the technique mentioned in this paper. Therefore, because of the representativeness of these three algorithms, they are used for comparative experiments.

In order to ensure the fairness of the experimental comparison, the same data set is selected. It can be seen from TABLE 2 that Real-estate1 has the most comprehensive data and it is suitable for experiments and comparisons, so this data is selected as a training sample for these algorithms. For ease of analysis, Precision, Recall, F1, Accuracy, AUC are used as the evaluation criteria in TABLE 6.

As shown in TABLE 6, the proposed method is compared with the other three methods. First, the two most important indicators are analyzed, AUC and time of the trained model. The algorithm proposed in this paper has made significant progress in AUC and time. The proposed algorithm can take only 67 hours in a time when the classification effect

TABLE 6. Compare the performance of various methods.

Method	Result	Positive			Negative			Accuracy	AUC	Time
		Precision	Recall	F1	Precision	Recall	F1			
Deep Learning	**Proposed Methodology**	**74.25%**	76.33%	**79.24%**	**88.33%**	**93.25%**	**93.85%**	**88.45%**	**90.14%**	**67.36h**
	CNN-based PSO by Da *et al.* [15]	62.54%	**78.25%**	61.52%	81.72%	81..21%	81.00%	75.62%	77.5%	98.34h
	Imbalanced text sentiment classification by Li *et al.* [8]	61.32%	65.41%	62.24%	82.15%	82.45%	84.34%	71.65%	79.6%	77.25h
	CNN+Social attention by Yang *et al.* [40]	42.38%	65.16%	60.14%	79.24%	75.16%	80.15%	69.82%	77.5%	86.51h

reaches 90.14%. This result can be applied in the project and approved by the advertisers. Followed by Li's algorithm, it takes 77 hours when AUC reaches 79.6%. Da's and Yang's algorithms have the same AUC of 77.5%, but Yang spends 86 hours, while Da spends 98 hours. As can be seen from other indicators, in addition to Recall, all the indicators of the method proposed in this paper are better than other algorithms. Therefore, it can be concluded that the algorithm proposed by the paper has a good effect in the keyword selection problem, and it has high commercial value in advertising keyword pushing and can be applied in practice.

V. CONCLUSION AND FUTURE WORK

Google Ads is very popular with advertisers. Because it helps advertisers push ads. But a large number of advertisers have caused a lot of losses in advertising investment because they are not smart enough to choose the keywords that are suitable for them. However, when solving the keyword selection problem, it is necessary to face some technical problems such as the keyword length is too short, the language type is not single, the data is unbalanced, the classification effect is poor, and the speed of classification is slow. Therefore, the method proposed in this paper is to study these problems.

According to the experimental results, the combination of machine learning, deep learning, evolutionary computation, and text embedding methods, as well as basic knowledge such as resampling, particle swarm optimization algorithm and convolutional neural network, effectively solve the problem of difficulty in selecting keywords. And to a certain extent, the efficiency of selecting keywords and the quality of selected keywords have been improved in a short time, a series of challenges in selecting advertising keywords are successfully overcome.

In the actual application, if the AUC of the model is less than 66.8%, which means that a lot of time is wasted, and it will be difficult to select keywords with high relevance and favorable price. On the contrary, if the model created in this article is used well in the application, it will bring great benefits to the advertisers, and the advertiser will effectively obtain more economic benefits at a lower cost and faster.

There are still several problems in this paper that need to be improved in depth. First, the amount of data used in this model is relatively small, because it takes a lot of money and time to generate data. Second, the main goal of this paper is to select the right keywords in a shorter time, so it needs to be assumed that other influencing factors are the same. This has certain limitations. In practice, there are many factors to be considered when choosing keywords, such as the type of ads, the format of the ad that the advertiser needs and the characteristics of the recipients of the advertising message. In fact, these factors can also have a significant impact on the performance of the selected keywords. Therefore, these factors will also become the direction of future research.

In future research, with the number of data increases and the depth of research promoting, the problems in selecting keywords for Google Ads will be further resolved and promoted. This will help advertisers save a lot of capital costs and will bring them greater economic benefits and value.

REFERENCES

[1] A. L. Murphy, S. Peltekian, and D. M. Gardner, "Website analytics of a Google Ads campaign for a men's mental health website: Comparative analysis," *JMIR Mental Health*, vol. 5, no. 4, Oct. 2018, Art. no. 12428.

[2] A. Zink, B. Schuster, M. Rüth, M. P. Pereira, W. G. Philipp-Dormston, T. Biedermann, and S. Ständer, "Medical needs and major complaints related to pruritus in Germany: A 4-year retrospective analysis using Google AdWords Keyword Planner," *J. Eur. Acad. Dermatol. Venereol.*, vol. 33, no. 1, pp. 151–156, Jan. 2019.

[3] W. C. Serrano, M.-M. Chren, N. N. Aji, S. Pagoto, E. Linos, and J. S. Resneck, Jr., "Online advertising for cancer prevention: Google Ads and tanning beds," *JAMA Dermatol.*, vol. 152, no. 1, pp. 101–102, Jan. 2016.

[4] G. Szymanski and P. Lininski, "Model of the effectiveness of Google Adwords advertising activities," in *Proc. IEEE 13th Int. Sci. Tech. Conf. Comput. Sci. Inf. Technol. (CSIT)*, vol. 2, Sep. 2018, pp. 98–101.

[5] P. S. Desai, W. Shin, and R. Staelin, "The company that you keep: When to buy a competitor's keyword," *Marketing Sci.*, vol. 33, no. 4, pp. 485–508, Apr. 2014.

[6] Y. Shen, Q. Zhang, J. Huang, Y. Lu, K. Lei, and J. Zhang, "Improving medical short text classification with semantic expansion using word-cluster embedding," in *Proc. Int. Conf. Inf. Sci. Appl.*, vol. 514, Jun. 2018, pp. 401–411.

[7] S. M. Rezaeinia, R. Rahmani, H. Veisi, and A. Ghodsi, "Sentiment analysis based on improved pre-trained word embeddings," *Expert Syst. Appl.*, vol. 117, pp. 139–147, Mar. 2019.

[8] Y. Li, H. Guo, M. Gu, J. Yang, and Q. Zhang, "Imbalanced text sentiment classification using universal and domain-specific knowledge," *Knowl.-Based Syst.*, vol. 160, pp. 1–15, Nov. 2018.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, no. 1, pp. 321–357, Jun. 2002.

[10] Q. Li, J. Dong, Q. Li, C. Wang, and J. Zhong, "A neural model for type classification of entities for text," *Knowl.-Based Syst.*, vol. 176, pp. 122–132, Jul. 2019.

[11] F. Li, Y. Yin, X. Mao, R. Shi, and J. Shi, "Method of feature reduction in short text classification based on feature clustering," *Appl. Sci.*, vol. 9, no. 8, p. 1578, 2019.

[12] O. Edo-Osagie, I. Lake, B. De La Iglesia, and O. Edeghere, "Attention-based recurrent neural networks (RNNs) for short text classification: An application in public health monitoring," in *Proc. Int. Work-Conf. Artif. Neural Netw.*, 2019, pp. 895–911.

[13] R. Türker, L. Zhang, H. Sack, and M. Koutraki, "Knowledge-based short text categorization using entity and category embedding," in *Proc. Eur. Semantic Web Conf.*, 2019, pp. 346–362.

[14] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*. [Online]. Available: https://arxiv.org/abs/1408.5882

[15] G. L. F. da Silva, T. L. A. Valente, A. C. Silva, A. C. de Paiva, and M. Gattass, "Convolutional neural network-based PSO for lung nodule false positive reduction on CT images," *Comput. Methods Programs Biomed.*, vol. 162, pp. 109–118, Aug. 2018.

[16] S. Lu and S. Yang, "Investigating the spillover effect of keyword market entry in sponsored search advertising," *Marketing Sci.*, vol. 36, no. 6, pp. 976–998, 2017.

[17] L. Mostafa, "Webpage keyword extraction using term frequency," *Int. J. Comput. Theory Eng.*, vol. 5, no. 1, p. 174, 2013.

[18] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of Machine Learning*. 2010, pp. 760–766.

[19] X. Yan, J. Yan, and Y. Feng, "Gradient and particle swarm optimization based hierarchical cluster algorithm in WSN," *J. Zhengzhou Univ., Eng. Sci.*, vol. 2, no. 2, p. 7, Feb. 2016.

[20] C. T. Yue, J. Liang, H. Song, G. Li, Y. Han, and B. Qu, "A knee point driven particle swarm optimization algorithm for sparse reconstruction," in *Proc. 11th Int. Conf. Simulated Evol. Learn.*, Nov. 2017, pp. 911–919.

[21] J. Moore and R. Chapman, "Application of particle swarm to multiobjective optimization," Dept. Comput. Sci. Softw. Eng., Auburn Univ., Auburn, AL, USA, 1999, p. 32.

[22] S. P. Kapse and S. Krishnapillai, "An improved multi-objective particle swarm optimization based on utopia point guided search," *Int. J. Appl. Metaheuristic Comput.*, vol. 9, no. 4, pp. 71–96, Oct. 2018.

[23] S. Matiz and K. E. Barner, "Inductive conformal predictor for convolutional neural networks: Applications to active learning for image classification," *Pattern Recognit.*, vol. 90, pp. 172–182, Jun. 2019.

[24] J. Coquet, S. Bozkurt, K. M. Kan, M. K. Ferrari, D. W. Blayney, J. D. Brooks, and T. Hernandez-Boussard, "Comparison of orthogonal NLP methods for clinical phenotyping and assessment of bone scan utilization among prostate cancer patients," *J. Biomed. Inform.*, vol. 94, Jun. 2019, Art. no. 103184.

[25] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mech. Syst. Signal Process.*, vol. 115, pp. 213–237, Jan. 2019.

[26] A. Dahou, M. A. Elaziz, J. Zhou, and S. Xiong, "Arabic sentiment classification using convolutional neural network and differential evolution algorithm," *Comput. Intell. Neurosci.*, vol. 2019, Feb. 2019, Art. no. 2537689.

[27] Y. Sun, B. Xue, G. G. Yen, and M. Zhang, "A particle swarm optimization-based flexible convolutional autoencoder for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 8, pp. 2295–2309, Aug. 2019.

[28] B. Fielding and L. Zhang, "Evolving image classification architectures with enhanced particle swarm optimisation," *IEEE Access*, vol. 6, pp. 68560–68575, 2018.

[29] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2013, pp. 3111–3119.

[30] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.

[31] Y. Wang, J. Wang, H. Lin, X. Tang, S. Zhang, and L. Li, "Bidirectional long short-term memory with CRF for detecting biomedical event trigger in FastText semantic space," *BMC Bioinf.*, vol. 19, no. 20, Dec. 2018, Art. no. 507.

[32] B. Ay Karakuş, M. Talo, G. Aydin, and R. Hallaç, "Evaluating deep learning models for sentiment classification," *Concurrency Comput., Pract. Exper.*, vol. 30, no. 21, p. e4783, Nov. 2018.

[33] Z. Su, H. Xu, Y. Xu, and D. Zhang, "Chinese sentiment classification using a neural network tool—Word2vec," in *Proc. Int. Conf. Multisensor Fusion Inf. Integr. Intell. Syst. (MFI)*, Beijing, China, 2014, pp. 1–6.

[34] B. Krawczyk, "Learning from imbalanced data: Open challenges and future directions," *Prog. Artif. Intell.*, vol. 5, no. 4, pp. 221–232, Apr. 2016.

[35] Z. Sun, Q. Song, X. Zhu, H. Sun, B. Xu, and Y. Zhou, "A novel ensemble method for classifying imbalanced data," *Pattern Recognit.*, vol. 48, no. 5, pp. 1623–1637, May 2015.

[36] S. Barua, M. M. Islam, K. Murase, and X. Yao, "MWMOTE–majority weighted minority oversampling technique for imbalanced data set learning," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 2, pp. 405–425, Feb. 2014.

[37] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," in *Proc. Int. Conf. Intell. Comput.* Berlin, Germany: Springer, 2005, pp. 878–887.

[38] Z. Hu and P. Jiang, "An imbalance modified deep neural network with dynamical incremental learning for chemical fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 66, no. 1, pp. 540–550, Jan. 2019.

[39] M. H. Zweig and G. Campbell, "Receiver-operating characteristic (ROC) plots: A fundamental evaluation tool in clinical medicine," *Clin. Chem.*, vol. 39, no. 4, pp. 561–577, Apr. 1993.

[40] Y. Yang and J. Eisenstein, "Overcoming language variation in sentiment analysis with social attention," 2015, *arXiv:1511.06052*. [Online]. Available: https://arxiv.org/abs/1511.06052

[41] P. Wang, B. Xu, J. Xu, G. Tian, C. Liu, and H. Hao, "Semantic expansion using word embedding clustering and convolutional neural network for improving short text classification," *Neurocomputing*, vol. 174, pp. 806–814, Jan. 2016.

[42] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, vol. 6, no. 1, pp. 58–73, Feb. 2002.

JING LIANG received the B.E. degree from the Harbin Institute of Technology, China, and the Ph.D. degree from Nanyang Technological University, Singapore. She is currently a Professor with Zhengzhou University. She has published about 140 articles in computational intelligence area, which are cited more than 9300 times according to Google Citation and her h-index is 29. Her main research interests include evolutionary computation, swarm intelligence optimization algorithm, differential evolution, multi-objective optimization, multi-modal optimization, and neural networks. Her dissertation received the IEEE CIS Outstanding Ph.D. Dissertation Award. She was an Associate Editor of the *IEEE Computational Intelligence Magazine*, from 2012 to 2017. She has been an Associate Editor of *Swarm and Evolutionary Computation*, since 2015, and IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, since 2017, and Executive Editor of the *Journal of Zhengzhou University (Engineering Science)*.

HAOTIAN YANG received the bachelor's degree from the Henan University of Technology, Henan, China, in 2017. He is currently pursuing the M.E. degree with Zhengzhou University. His current research interests include deep learning, and natural language processing and classification.

JIAJIA GAO received the bachelor's degree from the Henan University of Economics and Law, Henan, China, in 2017. She is currently pursuing the M.E. degree with the Henan University of Economics and Law. Her current research interests include demography, statistics, economics, and big data.

CAITONG YUE, photograph and biography not available at the time of publication.

SHILEI GE, photograph and biography not available at the time of publication.

BOYANG QU received the B.E. and Ph.D. degrees from the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He is currently an Associate Professor with the School of Electric and Information Engineering, Zhongyuan University of Technology, China. He has authored more than 70 SCI/EI research articles in peer-reviewed journals and international conferences, which are cited more than 3600 times according to Google Citation and her h-index is 24. His research interests include evolutionary computation, multi-modal optimization, and multi-objective optimization. He is also a Regular Reviewer of some journals, including the IEEE Transactions on Evolutionary Computation, IEEE Transactions on Systems, Man, and Cybernetics: Part B, *Information Sciences*, *Neurocomputing*, *Swarm and Evolutionary Computation*, and *Multidimensional Systems and Signal Processing*.