IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 8, 2020, accepted September 27, 2020, date of publication October 5, 2020, date of current version October 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3028588

Fake Review Detection Based on Multiple Feature Fusion and Rolling Collaborative Training

JINGDONG WANG, HAITAO KAN, FANQI MENG, QIZI MU, GENHUA SHI, AND XIXI XIAO
School of Computer Science, Northeast Electric Power University, Jilin City 132012, China

Corresponding authors: Fanqi Meng (mengfanqi@neepu.edu.cn) and Haitao Kan (1145758763@qq.com)

This work was supported by the Science and Technology Development Plan of Jilin Province (Network Public Opinion Analysis and Dynamic Evolution Mechanism Research for Public Crisis Early Warning) under Grant 20190303107SF.

ABSTRACT Fake reviews may mislead consumers. A large number of fake reviews will even cause huge property losses and public opinion crises. Therefore, it is necessary to detect and filter fake reviews. However, most existing methods have lower accuracy in detecting fake reviews due to they just use single features and lack of labeled experimental data. To solve this problem, we propose a novelty method to detect fake reviews based on multiple feature fusion and rolling collaborative training. First, the method requires an initial index system with multiple features such as text features, sentiment features of reviews and behavior features of reviewers. Second, the method needs an initial training sample set. Thus, we designed related algorithms to extract all the features of a review. Then the classification of the review is labeled manually. Finally, the method uses the initial sample set to train 7 classifiers, and the most accurate classifier will be selected to classify new reviews. The novelty of the method lies in that the features and the classification labels of the new reviews will be added into the initial sample set as new samples. So the size of the sample set will increase automatically. The experimental results in the reviews of yelp shopping website show that the accuracy of the proposed method for detecting fake reviews is 84.45%, which is 3.5% higher than the baseline methods. And compared with the latest deep learning model, its baseline precision has increased by 5.3%. According to the Friedman test, the support vector machine (SVM) classifier and random forest (RF) classifier has been proven to be the best one by statistical means. It means our method which uses multiple features has higher accuracy than the baseline models. Meanwhile, it also resolves the problem of lacking labeled training samples in fake reviews detection.

INDEX TERMS Fake review detection, machine learning, multiple feature fusion, feature extraction, rolling collaborative training.

I. INTRODUCTION

For online shopping, there are inconsistencies between products' information and products that consumers receive offline, which leads consumers to read a large amount of reviews of target products to assist judgement [1]. Therefore, product reviews not only affect consumers' purchase intentions, but also affect the interests of enterprises [2]. Positive reviews attract more potential consumers, while negative reviews drain potential consumers. To obtain higher profits, unscrupulous merchants usually hire professional writers to write fake positive reviews for their products, so as to increase the popularity of products to attract potential consumers, and at the same time write fake negative reviews for competitors to suppress them. These behaviors not only seriously mislead potential consumers, but also are not conducive to the stable development of e-commerce platforms [3]. Relevant researches show that fake reviews are not easily recognized by consumers. To purify the online shopping platform, bring consumers good shopping experience, and obtain truthful and effective reviews, effective methods are urgently needed to detect fake reviews.

Since the reviews published by reviewers are uncertain and credibility is unknown, it is necessary to establish an evaluation index system based on the credibility of the reviews. Although some scholars had also proposed many indicators for judging fake reviews, they ignored the interaction between the features. For example, only unilateral features of the a score sheet are used to detect fake reviews [4]. 1-star reviews and 5-star reviews are more likely to be fake reviews than

The associate editor coordinating the review of this manuscript and approving it for publication was Senthil Kumar.

3-star reviews [5]. It is a one-sided approach to judge whether a review is a fake review only from the perspective of scoring. The purpose of writing a fake review is not only to improve the star rating of the product but also to influence consumers' shopping desire. Reviews with positive emotions will enhance the readers' preference of the products, otherwise, they will reduce the favorability. Therefore, it is very reasonable to add sentiment analysis to the detection task. Currently, there are few experimental datasets for detecting fake reviews [6]. Labeling data is a huge project that requires time and efforts. To solve the above problems, a fake review detection method based on multi-feature fusion and rolling collaborative training is proposed in the present study. The novelty of this article lies in the following two aspects: First, multiple factors such as sentiment and user behavior are integrated into a multi-level, multi feature evaluation system. We propose a method to quantify the intensity of emotions, to analyze whether the emotional tendency of the reviewer conflicts with their review behavior, so as to provide support for judging fake reviews. Second, in order to use unlabeled data to assist model learning, we propose a method that uses rolling decision-making to coordinate training data so that the features extracted by the model can be dynamically updated, thereby reducing the impact of time factors on the detection performance of the classification model. The details are described below.

- The text representation model and sentiment analysis are used to enrich the text features, and the user's abnormal comments are analyzed to represent the user's behavior. The evaluation index system is established based on the text features and user characteristics, and the feature extraction algorithm based on the index system is designed in this step.
- The extracted data is quantized, and the extracted features are divided into two feature sets according to attributes. These two feature sets are used as the training set of the basic classifier in the classification model. Train seven basic classifiers with a small amount of labeled data, adjust the parameters of each classifier to select two optimal classifiers as part of the integrated model, and then, during collaborative training, a large amount of unlabeled data is used to expand the training data set in chronological order to ensure that the data information will not lose timeliness due to the change of time.
- The integrated learning model implements fake review detection.
- A statistical test was performed on the basic classifiers, and the performance differences of the basic classifiers were compared.
- Compare the classification effect of this method on the two datasets.

The rest of this article is organized as follows: Section 2 reviews related works; Section 3 explains how to establish a multi-level evaluation index system and fake review detection model. Including index selection, feature extraction, classifier selection, and collaborative training, etc. Section 4 confirms the practicality of the proposed indicators and the validity of the detection model through experiments; Section 5 discusses the limitations of the current research. Section 6 summarizes the work of this article and gives possible future research directions.

II. RELATED WORK

A. IDENTIFY FAKE REVIEWS FROM THE PERSPECTIVE OF THE REVIEW TEXT

User reviews are usually short text, and fake review detection is a binary classification problem [7]. The goal of this task is to determine whether a review is a fake review. Existing methods mainly follow the work of literature [8] and use machine learning methods to construct the classifier. Jindal and Liu [9] and others divided fake reviews into three categories: reviews involving only brands, reviews without substantial content, and untrue reviews. At that time, there were no public data sets for fake reviews and they decided if a review is fake or not by judging whether the review is a duplicate review. Yoo and Gretzel [10] and others collected hotel review data including 40 truthful review data and 42 fake review data as a dataset. In terms of linguistics, they used a standard statistical method to compare truthful reviews with fake reviews, and it was found that there were indeed differences in the expression between the two. Ott *et al.* [11] and others have built the ''Golden Standard'' in the field of fake review detection through the online crowdsourcing service provided by Amazon. By analyzing the part-of-speech distribution of words and extracting part-of-speech features, they used an n-gram-based feature set, and use naive Bayes (NB) and SVM as classifiers. Through the use of Linguistic Inquiry and Word Count (LIWC) [12] software for feature extraction, SVM classification is used for bigram+LIWC mixed features. Feng *et al.* [13] studied the deeper syntactic structure of the review content, focused on the analysis of writing style, extracted the features of context-free grammar, and used SVM to classify the ''Golden Standard''. Yanfang and Zhiyu [14] and others proposed a logic model for detecting fake reviews.The model added sentiment feature information of the review text, and combined the measurement method of the sentiment outliers in the review with the research on the usefulness of the review to obtain the comprehensive ranking of the comment results, so as to obtain the credibility sequence of the reviews. In the model verification set, by comparing the correspondence between the review text sequence processed by the model, and the truthful review text sequence, the purpose of fake review detection is finally achieved. Li *et al.* [15] and others used a method based on the latent dirichlet allocation (LDA) topic model to build a model by comparing the probability distribution of the subject words of truthful reviews and fake reviews and used the SVM classifier for classification. Sun *et al.* [16] used the Naive Bayesian model to calculate

user behavior features which were selected, then the result will be combined with the review content features, and finally the model of SVM is used to identify fake reviews. As neural network algorithms have made important breakthroughs in fields such as images, natural language processing tasks based on neural networks have also made great progress. With the success of attention mechanism in the field of image classification, attention mechanism has also been introduced into natural language processing tasks. Ren and Ji [17] explored neural network models to learn document-level representations, and integrated neural network features and discrete features to detect fake reviews. Li *et al.* [18] proposed a neural network-based model (SCNN model) to learn the representation of documents, and calculate the weight of sentences to detect fake reviews. Zhang *et al.* [19] proposed a Deceptive Review Identification by Recurrent Convolutional Neural Network (DRI-RCNN) model by using word context and deep learning to identify fake reviews.

B. IDENTIFY FAKE REVIEWS FROM THE REVIEWER'S BEHAVIOR CHARACTERISTICS

Lim *et al.* [20] analyzed the behavior of reviewers and used the scoring behavior to detect fake reviews for the first time. Based on the research on the behavior of false reviewers, Wu *et al.* [21] gave some criteria for identifying fake reviewers, such as contribution weight ranking and positive outlier probability. These artificially defined criteria can help us identify fake reviews. Wang *et al.* [22] believe that the relationship between reviews, reviewers, and businesses can reveal the fake review activities of fake reviewers. It is proposed to use a heterogeneous review graph with three types of nodes to capture the relationship between reviews, reviewers, and businesses reviewed by reviewers. An effective iterative algorithm for solving these three concepts based on the graph model has also been developed, so as to find reviewers with poor credibility and regard them as fake reviewers. Based on Wang's research, Wang [23] proposed an iterative framework with three types of nodes: reviewer, review, and product based on the review graph, and combined with the reviewer's behavior and review metadata features to detect fake reviewers. Xie *et al.* [24] analyzed the difference between the time patterns of the real reviewers and the fake reviewers, and used the method of mining abnormal patterns in the time series to identify the fake reviews. Yafeng *et al.* [25] believed that fake reviews are accounted for a relatively small amount and difficult to mark, so they proposed a new PU-learning technology to detect deceptive fake reviews and achieve good recognition results. Wang *et al.* [26] proposed a neural network based on attention mechanism, and combined with language features and behavioral features to detect fake reviews. Multilayer perceptron (MLP) and convolutional neural network (CNN) are used to learn feature vectors, and language features and behavioral features are weighted through attention mechanisms. Jain *et al.* [27] have proposed two different methods – multi-instance learning and hierarchical architecture to handle the variable length review texts. Experimental results on multiple benchmark datasets of deceptive reviews performed well. Fang *et al.* [28] used dynamic knowledge graphs to detect fake reviews Masood *et al.* [29], Rastogi and Mehrotra [30], Barbado *et al.* [31] and others identified fake reviews by combining multiple features such as text features and user behavior information.

C. INSUFFICIENCY OF EXISTING RESEARCH

At present, there are several problems in the related research of fake review detection:

- Fake review detection is usually based on the classification method under the full-supervised framework. The full-supervised learning method requires a large amount of labeled data as training samples and labeled data are difficult to obtain. Manually labeling data consumes a lot of manpower and material resources, and there are inaccurate subjective labels, which will limit the progress of fully supervised learning.
- Scholars try to use unsupervised learning methods, which use unlabeled data for cluster analysis to classify through unsupervised learning, but for such more confusing detection tasks, the accuracy is not high [22].
- Semi-supervised learning well balances the main problems of fully supervised learning and unsupervised learning. However, in the current detection task, only the basic features such as part-of-speech or n-gram are used for modeling, and the factors such as the interaction between different features are ignored, which reduces the classification effect [9].
- Existing deep learning models perform well in plain text classification, but they are not effective in the field of fake review detection. The main reason is that it is difficult for fake reviews to find fixed features from plain text, and it needs to be analyzed from multiple angles, such as user information, business information and other factors. Currently, a multiple dimensions method to detect fake reviews is urgently needed [32].

In view of the above problems, this article proposes a detection method that can solve the problems of shortage of standard datasets, improper feature selection, and low detection accuracy. The method analyzes the relationship between each feature, formulates a review credibility evaluation index system, designs feature extraction and quantification methods, and finally constructs a fake review detection model based on multi-feature fusion and rolling collaborative training.

III. THE FAKE REVIEW DETECTION MODEL

A. OVERALL FRAMEWORK

The overall framework of fake review detection in this article is shown in Figure 1. Step 1 (S1). Build a multi-level indicator system that includes review text and user behavior information (see section III.B). Step 2 (S2). Use crawler technology to collect web reviews and user information. Step 3 (S3). Design a feature extraction algorithm for text and user information

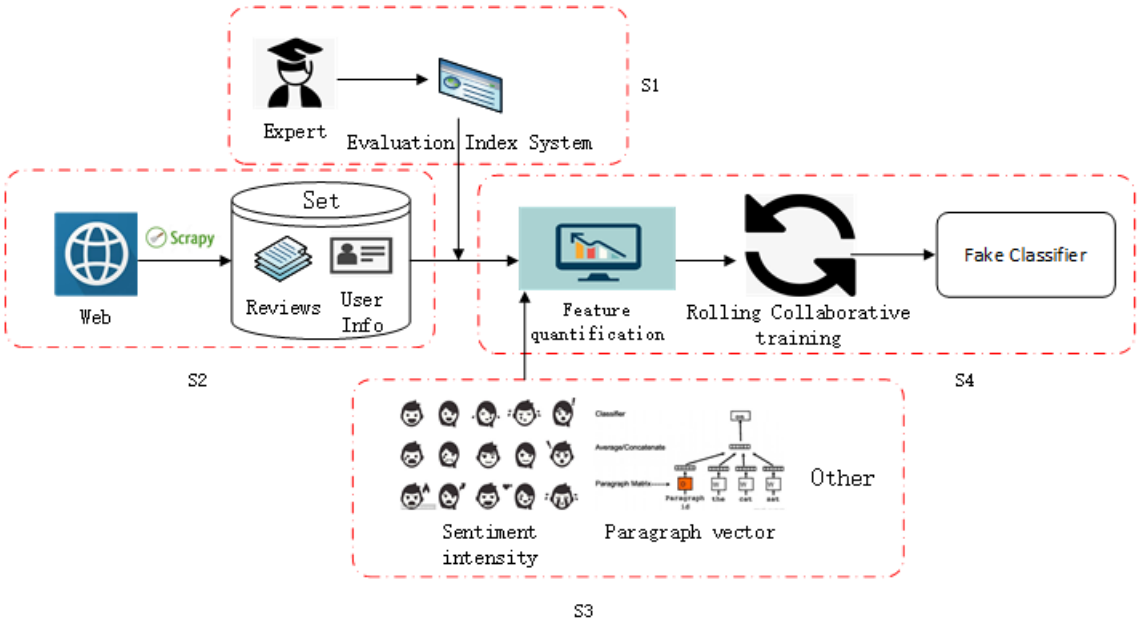

FIGURE 1. The overall framework of the fake review detection.

fusion (see section III.C). Step 4 (S4). Quantify indicators, delete irrelevant data, and build a classification model for fake review detection based on multi-feature fusion and rolling collaborative training (see section III.D).

B. THE CONSTRUCTION OF REVIEW CREDIBILITY INDEX SYSTEM

Constructing a representative feature set can effectively improve the classification accuracy and generalization ability of the model [33]. The development of the Internet makes it possible for consumers on online platforms to interact with each other. Users can share reviews and influence the purchase decisions of other consumers. The direct influence of perceived information is useful for purchase intention, and the antecedent constructs–needs of information, information credibility, and information quality–had a positive and significant impact on the perceived usefulness of online reviews. Information credibility is more relevant than information quality [34]. Due to the different emphases in various studies, the diversity of review objects and platform metadata, etc., the characteristic index systems constructed in different studies also have certain differences. This article mainly examines the credibility of the review, further refines it from the two main levels of the content of the review and the behavior of the reviewer, and builds the index set from multiple perspectives. The specific content is shown in Table 1.

C. FEATURE EXTRACTION

Pre-process unstructured data and text representation. Using the Doc2vec language model to express the text as a semantic vector, a fixed-dimensional feature vector is obtained and used as one of the features of review detection.

Doc2vec is also called Paragraph Vector, which is proposed by Le and Mikolov [35] based on the Word2vec model, which has some advantages. For example, instead of using fixed sentence lengths and accepting sentences of different lengths as training samples, Doc2vec is an unsupervised learning algorithm. This algorithm is used to predict a vector to represent different documents. The structure of the model potentially overcomes the shortcomings of the bag of words model. As shown in Figure 2.

The Doc2vec model is inspired by the Word2vec model. When predicting word vectors in Word2vec, the predicted words contain word meanings. For example, the word vector ''powerful'' mentioned above will be closer to ''strong'' than ''Paris.'' The same structure is also built-in Doc2vec. Paragraph vectors are added to the Doc2vec model, so that the Doc2vec overcomes the shortcomings of the lack of semantics in the bag of words model.

$$POS(r) = \frac{|adj(r)| + |adv(r)|}{|tw(r)|} \tag{1}$$

In formula (1), $POS(r)$ represents the part-of-speech feature value, $adj(r)$ represents the number of adjectives, $adv(r)$ represents the number of adverbs, and $tw(r)$ represents the total number of words in the review.

Calculate the intensity of the sentiment polarity for each review text. The intensity of sentiment polarity refers to the sum of the intensity expressed by the sentiment words in the review. With the help of a corpus and sentiment dictionary, extract the sentiment words and context structure of the specified part-of-speech collocation pattern, and construct a ternary sentiment unit, which is defined as: $u =< n, adv, w, >$, where u is a sentiment unit, w is a sentiment word, n is a negative word, and adv is an adverb

TABLE 1. Multi-level reviews credibility evaluation index system.

First-level Index	Secondary-level Index	Third-level Index	Index Description
Review text content	Semantic features	Paragraph vector	Deep semantic representation of text
	Lexical features	Part-of-speech frequency	The ratio of the frequency of each part of speech in the text to the total number of words
		First-person pronoun frequency	The ratio of the number of first-person pronoun words to the total number of words in the text
		Adverb frequency	The ratio of the number of adverbs of degree level in the text to the total number of words
	Sentiment features	Sentiment score consistency	Whether the sentiment polarity is consistent with the score
		Sentiment intensity	Text express sentiment intensity
		Sentiment vocabulary features	The ratio of positive and negative sentiment words in the text to the total number of words
	Character features	Review text length	The total number of words in the text
Review external information	Abnormal star	Text score deviation	The deviation of the score from the average score of the target object
		Reviewer rating deviation	The deviation of the review from the reviewer's average rating
		Extreme score	Whether it is a 1 or 5-star review
	Abnormal quantity	Maximum number of reviews on reviewer day	Maximum number of reviews posted by reviewers in a day
		Total reviews	Total number of reviews by reviewers
		Review frequency	The ratio of reviewers' daily reviews to the maximum number of reviews on historical days
	Abnormal content	Text similarity	The maximum similarity between the text and all texts of the target object

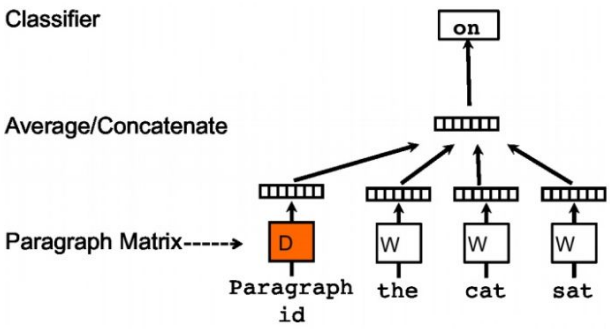

FIGURE 2. Doc2vec semantic model [35].

of degree. The formula for calculating sentiment intensity is as follows:

$$f_s = \sum_{(w_j \in re)} \left[(-1)^{(c_N)} \cdot advw(t) \cdot o(w_j) \right] \tag{2}$$

In formula (2), f_s represents the sentiment intensity of the review text, $o(w_j)$ represents the polarity of the sentiment word, re represents the review sentence, w_j represents the sentiment word in the review sentence, when the sentiment is positive, the value is 1, and the opposite is -1, $advw(t)$ represents the weight of adverbs of degree. Adverbs of different degrees have different weights, such as "more, plus, relatively, slightly more" has a weight of 1.25. "slightly, wildly, a little" weighs 0.5. and the "half-point, not big, mild, slightest" weights 0.25. c_N represents the number of negative words before each sentiment word. If there is no negative word, c_N is 0, if there is an odd number of negative words, $(-1)^{(c_N)}$ value is -1, even number of negative words then $(-1)^{(c_N)}$ value is 1. The algorithm is shown in Algorithm 1.

The time complexity of the program is $O(n^2)$. Due to the large number of words in the sentiment dictionary, the model training process consumes a lot of time. But the sentiment of a review is calculated. The value of the program running time is $0.45s$, the main time-consuming is import and export of data, and the length of the text will affect the running speed of the program.

Extract the consistency index of sentiment and score through formula (3):

$$f_{ss} = \begin{cases} 1, & (star \in (4,5)) \; and \; (f_s > 0) or (star \in (1,2)) \\ & and \; (f_s < 0) or (star = 3) and (f_s = 0) \\ 0, & other \end{cases} \tag{3}$$

In the formula (3), f_{ss} is the characteristic value of the sentiment score consistency index, *Star* is the rating of the review, *other* is expressed as other, and f_s is the sentiment polarity.

Perform text similarity calculation on the review text to be judged (reviews of a certain product) and the target text library (all reviews of the product), and record the calculation result as a characteristic index. Calculate the text-similarity between product reviews. Two reviews are characterized by Doc2Vec as paragraph vectors r, r_i, and the text-similarity

Algorithm 1 Dictionary-Based Sentiment Intensity Extraction Algorithm

Input: test text, positive dictionary, negative dictionary, negative word dictionary, degree adverb dictionary
Output: text sentiment intensity

```
{Sentences} ← review(participle)
for sentence i ∈ {Sentences} do
    {words} ←sentence i
    for all word j ∈ {words} do
        if word j ∈ {negdict} then
            if j > 0 and word(j − 1) ∈ {nodict} then
                p+ = word.weight
            else if j > 0 and word(j − 1) ∈ {plusdict} then
                p− = word.weight
            else
                p− = word.weight
            end if
        else if word j ∈ {posdict} then
            if j > 0 and word(j − 1) ∈ {nodict} then
                p− = word.weight
            else if j > 0 and word(j − 1) ∈ {plusdict} then
                p+ = word.weight
            else if j > 0 and word(j − 1) ∈ {negdict} then
                p− = word.weight
            else if j < len(word) and word(j + 1) ∈ {plusdict} then
                p− = word.weight
            else
                p+ = word.weight
            end if
        else if word j ∈ {nodict} then
            p− = α
        end if
    end for
end for
return result
```

calculation formula (4):

$$f_{cs} = max_{r_i \in R_a} \cdot cos(r, r_i) \tag{4}$$

In formula (4), f_{cs} represents the text-similarity feature value, r is the test to be tested, r_i is the review in the target text library R_a, traverses the target library and obtains the maximum value f_{cs} as the text-similarity feature value.

The external features (user behavior characteristics) of the review include the deviation of the rating of the review and the abnormal calculation of the number of reviews. Formula (5) for calculating the deviation of the review:

$$f_{RD} = \frac{|v_r - avg_{r_i \in Ra} v_{r_i}|}{r_p} \tag{5}$$

In formula (5), f_{RD} score deviation feature value, v_r is the rating level of the review, $avg_{r_i \in Ra} v_{r_i}$ is the average of all rating levels of the target product, r_p is the maximum possible rating system deviation, if the rating system of the review data source is five stars, the maximum rating deviation is 4.

The calculation formula (6) for the abnormal number of reviews:

$$f_{MNR} = \frac{Rev(a)}{max_{a \in A}?Rev(a)} \tag{6}$$

Submitting a large number of reviews in a day is anomalous behavior. By counting the ratio of the number of reviews posted on the user's day to the maximum number of posts on a historical day as the abnormal parameter of the number of reviews, the abnormal behavior characteristics of the publisher of the fake review can be obtained. $Rev(a)$ is the number of user reviews per day, a indicates user reviews during the day, A indicates the set of user reviews during the day on all historical days, and f_{MNR} indicates the frequency of reviews.

Whether the extreme score extraction formula (7):

$$f_{yn} = \begin{cases} 1, & (star \in (1, 5) \\ 0, & (star \in (2, 3, 4) \end{cases} \tag{7}$$

In formula (7), the star indicates the star rating of the review, and f_{yn} indicates whether it is an extreme review value. If it is an extreme review, the f_{yn} value is 1.

D. ROLLING COLLABORATIVE TRAINING ALGORITHM

By extracting the content features of the review text and the features of the external information of the review, a scrolling collaborative training fake review detection model is constructed.

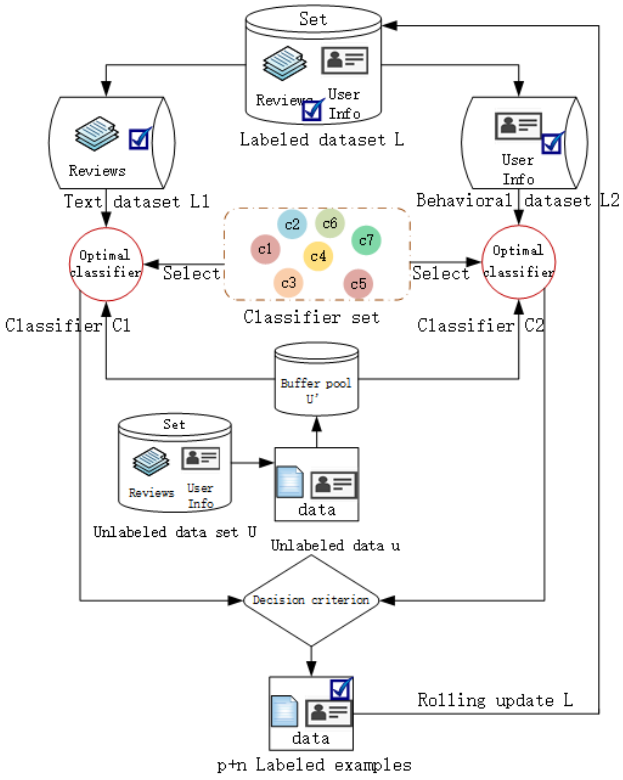

FIGURE 3. Rolling collaborative training.

The operation process is shown in Figure 3. The specific training steps are as follows:

Step1: Perform feature extraction on the review data in the labeled review data set L and quantify the feature attribute values to obtain the labeled data set L_1 based on the view of review text and the labeled data set L_2 based on the external view of the review;

Step2: Take u sample data from the unlabeled data set U in time series and add it to the buffer pool U'

Step3: Use training sets L_1, L_2 to train multiple classifiers $c1, c2, c3$, etc. in the classifier set, and select the optimal classifiers C_1, C_2 based on their classification effects on the two feature attribute sets;

Step4: Use C_1 to label all the reviews in U', add the p positive examples and n negative examples with the highest label confidence in the classification result to L, and update L_2;

Step5: Use C_2 to label all the reviews in U', add the p positive examples and n negative examples with the highest label confidence in the classification result to L, and update L_1;

Step6: Remove the above $2(p + n)$ reviews from U.

Step7: Randomly generates $2(p+n)$ new candidate reviews from U to supplement U'.

Step8: Judge whether U is empty or the number of iterations reaches the set threshold, if the conditions are met, the iteration ends, otherwise, the iteration continues.

Entering data with high confidence in the model into the model training module and constant updating the training classifier iteratively not only ensures that the detection model will not affect the detection effect due to factors such as time advancement but also uses unlabeled data to train the model. Since the classifier finally obtained through collaborative training comes from two different views, in order to reduce the final "false positive rate" of the model, this article sets the judgment basis for whether the review is fake: only when two classifiers mark it as fake at the same time, this review is fake. In order to achieve the best classification effect for each view, the classification performance of seven common classification models: random forest (RF) [36], decision tree (DT) [37], naive Bayes (NB) [38], K nearest neighbor (KNN) [39], support vector machine (SVM) [40], logistic regression (LR) [41], linear discriminant analysis (LDA) [42] were tested on the two views, and finally determine the base classifier to complete the training of the fake review detection model.

The time complexity of the model is $O(n)$, and the time cost is mainly spent on data preprocessing. The time complexity of feature extraction except the sentiment module is $O(n)$. The overall framework time complexity is $O(n^2)$, the next step is to optimize the emotional feature extraction module to reduce the running time of the program.

IV. EXPERIMENT AND ANALYSIS

A. PURPOSE OF THE EXPERIMENT

In this study, review texts in the field of e-commerce are used as the experimental data set to test the calculation method of sentiment intensity and the validity of the Doc2vec text representation network model. The optimal classifier combination is selected through experiments to complete the classification model construction. Test and verify the effectiveness of the multi-feature fusion rolling collaborative training method proposed in this research, which is more accurate than traditional text classification methods.

B. EXPERIMENTAL DATA

This article obtains the original experimental dataset YelpCHI http://odds.cs.stonybrook.edu/yelpchi-dataset/ from the yelp review website. The data includes user ID, the total number of reviews, review content, review level, review time, etc. A total of 5854 records were used as the data set for this experiment, and the fake reviews were marked with the help of the fake reviews filtering system of the yelp review website. The experimental data set is shown in Table 2. The number of training sets and test sets are divided according to the ratio of 8: 2. The final evaluation index of the experimental results adopts the comprehensive index F1 value and accuracy.

TABLE 2. Experimental data set.

Data type	Number of reviews	Number of users
Truthful	5076	4231
Fake	778	743
Total	5854	4974

C. EXPERIMENTAL PLATFORM

The algorithm used in this research uses the server operating environment as Win64; processor Intel (R) Core (TM) i5-5200U CPU @ 2.20GHz 2.20GHz; running memory 8G; Python 3.7.0 version; Tensor Flow 1.13.1 version; Gensim 3.8.0 version; Scikit-learn 0.20.1 version; Text segmentation and part-of-speech tagging are performed using NLTK tools.

D. EXPERIMENTAL SETUP AND RESULT ANALYSIS

1) FEATURE EXTRACTION

Feature extraction is performed on the numerical data through the above algorithm, and the sentiment feature is implemented with Algorithm 1. The sentiment dictionary used in this article is provided by SenticNet https://www.sentic.net/downloads/. After the data is filtered and processed, it contains 55,311 positive emotional words with sentiment intensity, 44,589 negative sentiment words with sentiment intensity, and the sentiment intensity of each sentiment word is in the range [0,1]. For example, the sentiment intensity of the positive sentiment word "acclaimed" is 0.91, while the sentiment intensity of "abbreviate" is only 0.046, which is almost neutral. The adverb of degree dictionary used is the English version of the "sentiment" data set. Among them, there are 178-degree adverbs, divided into 6 levels, namely "extreme," "very," "more," "slightly," "insufficiently," "over". For example, "absolutely" is "extreme" level, "a bit" is "slightly" level.

The text extracts the vectorized representation of sentences through the Doc2Vec language model, sets the extraction vector dimension, and selects the optimal parameters and vector dimension through iterative training. With the help of the NLTK language model package, a part-of-speech feature extraction algorithm is designed to extract part-of-speech features. Reviewer features are also achieved through extraction algorithms. Since the extracted feature values have positive and negative, we should consider the normalization method without changing the positive and negative [43] when choosing a standardized method. After the features are extracted, the scale of the data is the absolute maximum value, and the positive and negative signs are retained, that is, within the interval [-1.0,1.0], as in formula (8):

$$y_i = (-1)^n \frac{|x_i|}{max_{i \in z}|x_i|} \tag{8}$$

In formula (8), y_i is the standardized feature value, x_i is the unstandardized feature value, i is the current number, z is the total number of feature values of the feature, and n is the variable coefficient.

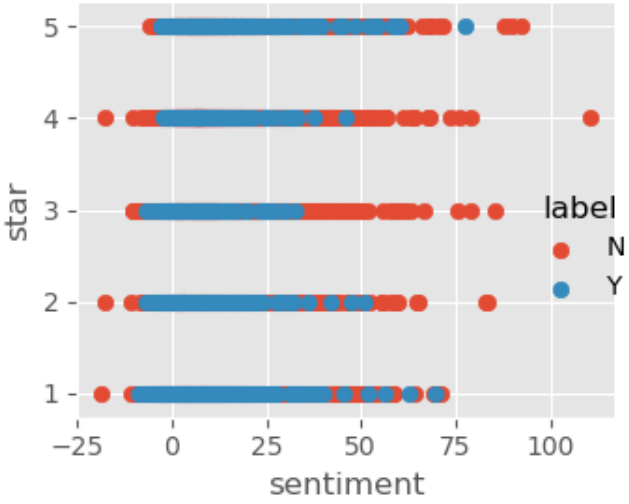

FIGURE 4. Consistent display of sentiment and ratings.

According to the statistical results of the indicator data of whether the sentiment scores are consistent in Figure 4, a large number of fake reviews exist in extreme ratings, and there are fake reviews with obvious high sentiment values in the 1-star rating. The reason may be that the reviewer copied the content of other reviews in order to improve the efficiency of writing fake reviews. The low rating is to lower the overall rating of the product to mislead consumers. This situation also exists in 5-star reviews and is very serious. From the perspective of sentiment analysis, the interval of sentiment value of fake reviews is mainly in [-10,35], and the interval of sentiment value of truthful reviews is [-20,80]. From this, it can be seen that truthful reviews and sentiments will be consistent, so sentiments cannot be used as the only indicator.

The parameter setting of the text representation model is shown in Table 3.

TABLE 3. Parameter setting table of Doc2Vec semantic model.

Doc2Vec model parameters	value
min_count	1
Window	5
Sampl	1e-3
Negative	5
Workers	4
Epochs	70

Test the influence of the vector dimension obtained by the Doc2Vec language model training on the classification effect, and conduct the classification experiment for different dimensions. The result is shown in Figure 5.

As can be seen in Figure 5, the accuracy of RF, SVM, LR, and LDA is relatively high. Within a certain range,

TABLE 4. The ten-fold cross result of each classifier based on text features.

Classifier Cross-validation	RF	LR	LDA	KNN	DT	NB	SVM
1	81.13%	83.02%	84.90%	69.81%	69.81%	67.92%	84.91%
2	86.54%	86.64%	80.77%	78.85%	78.84%	69.23%	86.54%
3	71.15%	73.18%	69.23%	67.31%	69.23%	69.23%	75.00%
4	73.08%	73.18%	73.08%	73.00%	69.33%	75.00%	75.00%
5	69.23%	71.15%	71.16%	71.15%	65.38%	67.31%	71.15%
6	78.85%	73.08%	71.15%	59.62%	59.62%	71.15%	78.85%
7	80.77%	80.78%	78.85%	71.14%	69.25%	78.85%	82.70%
8	78.86%	73.08%	71.25%	57.69%	69.23%	67.31%	80.77%
9	78.85%	84.62%	78.85%	65.28%	71.15%	65.38%	82.69%
10	71.35%	69.23%	65.38%	65.38%	61.54%	67.31%	71.15%
average value	76.96%	76.76%	74.45%	66.60%	68.33%	69.87%	78.88%
variance	0.053	0.060	0.058	0.053	0.050	0.039	0.053

TABLE 5. The ten-fold cross result of each classifier based on the behavioral features.

Classifier Cross-validation	RF	LR	LDA	KNN	DT	NB	SVM
1	84.91%	86.79%	84.91%	84.91%	81.13%	64.15%	84.93%
2	86.44%	88.46%	86.54%	88.46%	86.54%	73.08%	86.53%
3	82.69%	76.92%	78.85%	76.92%	82.69%	73.08%	75.00%
4	86.54%	86.54%	80.77%	82.69%	88.46%	71.15%	75.00%
5	88.47%	88.46%	80.76%	76.92%	84.61%	75.00%	71.15%
6	88.46%	80.77%	80.76%	75.00%	80.76%	53.85%	78.85%
7	90.38%	82.69%	86.54%	84.72%	86.55%	75.00%	82.69%
8	84.62%	80.77%	78.85%	75.00%	78.85%	76.93%	80.77%
9	84.52%	80.87%	84.62%	75.00%	76.92%	61.54%	82.69%
10	88.56%	69.23%	80.77%	73.08%	86.44%	73.08%	71.15%
average value	86.57%	79.83%	82.34%	79.26%	83.31%	69.68%	78.88%
variance	0.023	0.055	0.029	0.051	0.036	0.070	0.053

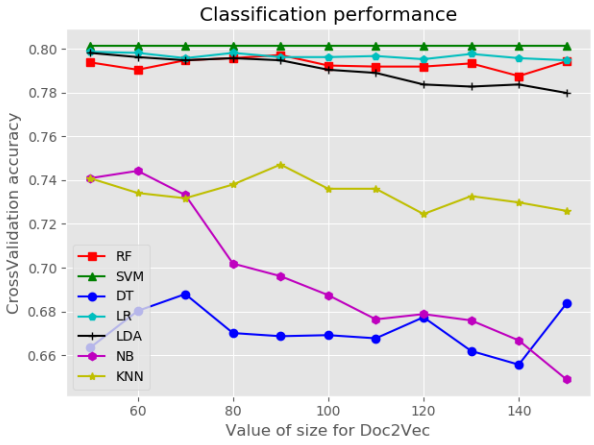

FIGURE 5. Influence of paragraph vector dimension on classification results.

as the dimension of the text vector increases, the accuracy of cross-validation also increases. SVM, LDA, and LR have maximum values in 80 dimensions, and RF has maximum values in 90 dimensions. In addition, because the previous research is based on the one-hot encoding of the N-garm language model, the vector dimension is determined by the number of words, and if the dimension is too high, it will lead to the curse of dimensionality and the disappearance of the gradient. The Doc2Vec language model is a neural network model. Mapping text to a high-dimensional vector space can well represent text content information. In order to reduce the time and resource cost of the classification model, 80 dimensions are selected as the following experimental text representation dimension.

2) SELECTION OF BASE CLASSIFIER

The cross-validation accuracy classification performance of the seven common classification models (RF, LR, LDA, KNN, DT, NB, SVM) have been separately tested on the two views to form a co-trained base classifier set. In order to ensure the stability of the experimental results, ten-fold cross-validation is used. The results are shown in Table 4 and Table 5, Figure 6 and Figure 7.

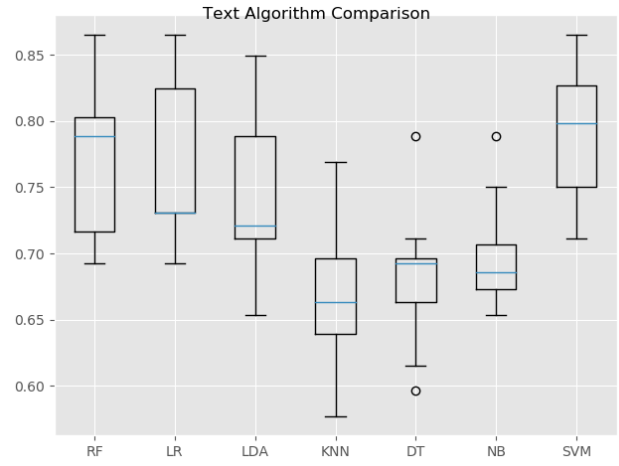

FIGURE 6. Performance of each classifier based on text features.

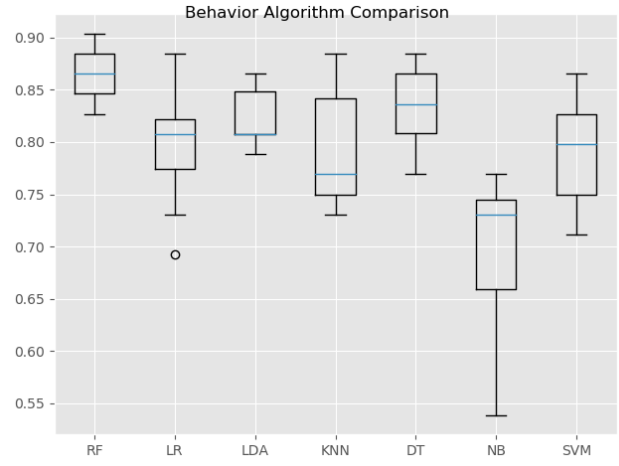

FIGURE 7. Performance of each classifier based on behavioral features.

From Table 4 and Table 5, Figure 6 and Figure 7, we can see that the overall performance of SVM and RF is better than other classification models. From the review content view only, SVM achieved the highest overall classification accuracy, followed by RF and LR. Text features include Semantic features, Lexical features, Sentiment features, Character features. In the reviewer's behavior view, the performance of RF is comparable to DT, and the overall accuracy of the former is slightly higher than the latter. Behavior features include Abnormal star, Abnormal quantity, and Abnormal content. Therefore, SVM and RF are selected as the two base classifiers in the experiment. The results show that in the false comment detection task, behavioral features have a greater impact on the classification results than text features.

3) ADJUSTMENT OF CLASSIFICATION MODEL PARAMETERS

The scale performance of the above-mentioned basic classifiers was tested with the scale coefficients of unlabeled data injection, and the effect of iteratively increasing the sample size on the classification performance was experimentally analyzed to set the scale coefficient threshold for collaborative training decision rules. The results are shown in Figure 8.

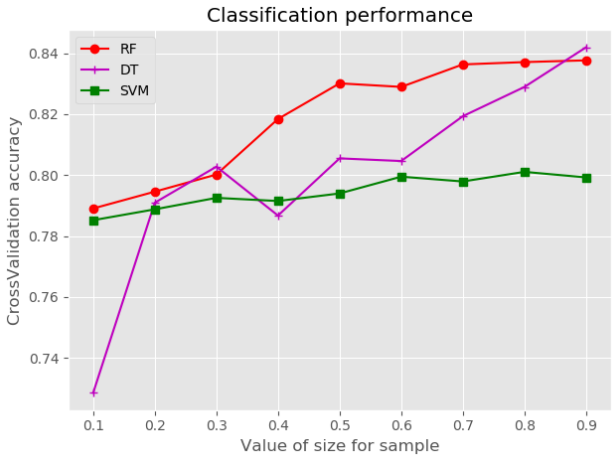

FIGURE 8. The impact of unlabeled data injection scale factor on the classification performance of each classifier.

As shown in Figure 8, with the increase of the scale factor and the sample size, the accuracy rate also increases, and the best effect is obtained when the scale factor is 0.8. As the number of samples increases, the training effect on the classifier gradually increases. However, when the number of samples is too large, it may cause overfitting and reduce the classification effect. Considering all aspects, 0.8 is selected as the scale factor.

We adjust the combination of features and selected classifiers, set up collaborative training decision rules and parameters, and perform collaborative training. The results are shown in Table 6, where C_1 represents the classifier on the feature view of the review content, and C_2 represents the classifier on the feature view of behavior. It can be seen from Table 6 that combination 1 achieves the best classification result, which is consistent with the experimental conclusion of the first step.

TABLE 6. Detection effects of different classifier combinations.

Classifier combinationC_1C_2	Accuracy	Precision	Recall	F1
1 (SVM, RF)	84.07%	83.23%	84.06%	81.56
2 (SVM, DT)	81.38%	81.32%	81.37%	81.35
3 (RF, SVM)	78.89%	62.23%	78.88%	69.58
4 (RF, DT)	83.11%	82.68%	83.11%	82.87

4) COMPARATIVE EXPERIMENT

Set up a control group for experiments to prove the effectiveness and portability of this method. A variety of methods were tested on two different datasets, one using YelpChi dataset (5854 pieces of data), the other using YelpRes dataset (15,141 pieces of data), and adjusting the experimental results

TABLE 7. Comparison of different model detection effects.

Data set	YelpChi				YelpRes			
Model	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
Supervised	81.07%	62.96%	81.06%	78.14	61.00%	66.76%	61.00%	60.97
Semi-supervised	82.53%	75.68%	82.52%	78.91	71.41%	67.75%	71.41%	71.24
Co-training	82.73%	81.34%	82.72%	79.67	79.85%	76.28%	79.85%	79.44
Co-training(multi-feature fusion)	84.45%	83.97%	84.45%	81.89	78.99%	80.16%	78.93%	78.91

obtained by adjusting the optimal parameters of each method. including:

Supervised [44]: Supervised learning of labeled data with a small number of samples. This article uses SVM, DT and RF classifiers with better classification results, compares the effects of the three classifiers, and selects the RF classifier with the best classification effect as the control group;

Semi-supervised [45]: a kind of semi-supervised learning, based on a single classifier for reinforcement learning, the classifier chooses RF;

Co-training: standard collaborative training algorithm, using the original feature set without any processing as input for model training;

Co-training (multi-feature fusion): The method proposed in this article adds text representation features and sentiment features based on original features, and trains the classifier by rolling update of the sample set. Experimental results are shown in Table 7.

It can be seen from Table 7 that the method proposed in this article has achieved good results, and its accuracy and recall rate are higher than other control groups, which proves the feasibility and effectiveness of this method in the fake reviews detection task. The recall rate in the second control group is much higher than the accurate rate, indicating that the model has a high ''fake positive rate,'' that is, some normal reviews are marked as fake reviews. The results on the two datasets prove the portability of the method in different fields. The reason why the detection accurate rate on the YelpRes dataset is lower than that the YelpChi dataset may be that the characteristics of reviews in different fields are different, but this method does achieve better detection results through multi-feature fusion.

We compare the Co-training(multi-feature fusion) with several baseline methods of opinion spam detection, including traditional approaches such as SVM, tensor decomposition methods, and some recent proposed deep learning models.

- Feature-Based Methods:

SVM + Bag of Words (BoW) [46] /n-grams + BF [47] mainly use machine learning algorithms with unigram, bigram, trigram. Behavior Features (BF) [48]are obtained from papers.

CHMM [49] is the Coupled Hidden Markov Model (CHMM) with two parallel HMMs that incorporate both the reviewer's posting behavior and co-bursting behaviors from other reviewers.

- Deep Learning Methods:

AEDA (attribute enhanced domain adaptive) [50] is a deep learning architecture for incorporating entities and their inherent attributes from various domains into a unified framework.

FCAN [32] is a Fusion Convolutional Attention Network (FCAN) to embed the user-level information into a continuous vector space, the representations of which capture essential clues such as user profiles or preferences.

TABLE 8. Experimental results on YelpChi dataset.

Model	Precision	Recall	F1
Supervised	62.96%	81.06%	78.14
Semi-supervised	75.68%	82.52%	78.91
SVM + Bag of Words (BoW)	70.97%	52.88%	60.61
CHMM	68.51%	64.58%	60.76
AEDA	68.54%	62.39%	65.32
FCAN	78.65%	67.31%	72.54
Co-training(multi-feature fusion)	83.97%	84.45%	81.89

It can be seen from Table 8 that the latest neural network-based model is better than the feature-based method. The performance improvement is mainly because the neural network can capture the semantic information of the text, but does not consider the dynamic changes of the information over time. We noticed the performance of the deep learning method. The improvement of it is mainly from the recall rate rather than the accurate rate, which indicates that the review-level semantic information may not fully reflect the difference between fake reviews and real reviews. This method combines text semantic information and other behavioral information to improve the accuracy of classification.

It can be seen from the above experimental results that the use of rolling collaborative training can improve the detection accuracy, because the rolling update sample data can dynamically adjust the classifier parameters. Explain that online reviews will change over time. The reason is that e-commerce platforms have made countermeasures against false reviews, and professional writers will also study the means to bypass platform detection. In this way, traditional

methods are difficult to capture the stable characteristics of fake reviews. The method proposed in this thesis can adjust the classifier parameters according to data changes to reduce the impact of data changes on the classification results. We added multi-dimensional features on the basis of this dynamically updated data to more accurately express data information and improve the classification accuracy. However, compared with the deep learning method, the time cost of deep learning will be less than the method in this article. The data preprocessing and feature extraction modules of this method will consume more resources than the deep learning method.

5) STATISTICAL ANALYSIS

In order to compare the different learning models used in this work, a statistical test has been applied on the experimental data. In particular, we have chosen the Friedman test [51], as this test is oriented to the comparison of several classifier methods on multiple datasets.

The Friedman test is based on a rank of each classification method in each dataset, where the best performing algorithm is assigned the rank of 1, the second-best is assigned rank 2, etc. Ties in this rank are resolved by the average of their ranks. We compare k algorithms on N datasets and let r_i denote the average order value of the i-th algorithm. In order to simplify the discussion, we don't consider the halving value, for the time being, then r_i obeys the normal distribution, and it's mean and variance are $(k+1)/2$ and $(k^2-1)/12$, respectively. Then, the Friedman statistic with $k-1$ degrees of freedom is written as follows:

$$\tau_{x^2} = \frac{k-1}{k} \cdot \frac{12N}{k^2-1} \sum_{i=1}^{k} (r_i - \frac{k+1}{2})^2 = \frac{12N}{k(k+1)} (\sum_{i=1}^{k} r_i - \frac{k(k+1)^2}{4}) \quad (9)$$

Nonetheless, it is shown that there is a more useful statistic that is distributed according to the F-distribution, and has $k-1$ and $(k-1)(N-1)$ degrees of freedom [51]. This statistic is referred to as the Friedman F, and is expressed as:

$$\tau_F = \frac{(N-1)\tau_{x^2}}{N(k-1)-\tau_{x^2}} \quad (10)$$

If the null-hypothesis of the Friedman test is rejected, post-hoc tests can be conducted, to complement the statistical analysis. In this work, we have conducted the Nemenyi tests, with the aim of gaining insight into the differences between the analyzed classifiers.

In the Nemenyi test [51], all classifiers are compared to each other. In this way, the performance of two classifiers is significantly different if their ranks differ, at least, the critical difference. The critical difference is computed with the following expression:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \quad (11)$$

On the text data set, the computation of the tests has been made as follows. In relation to the Friedman test, the ranks have been computed. For all the calculations, the α value is set to 0.05 [51]. $\tau_{x^2_{text}} = 40.5$, $\tau_{F_{text}} = 18.6$, the critical value of F distribution $F(k-1,(k-1)(N-1)) = 2.27$, so $\tau_{F_{text}} > F(6,54)$, the negative hypothesis is rejected, that is, not all classifiers have similar performance, so that they can be tested afterwards. Then perform the Nemenyi test. According to the above table query of DemŽar J [51], $q_\alpha = 2.272$, and calculate the critical value $CD_{text} = 2.19$ based on these values. According to the experimental results, the Friedman test chart is drawn as shown in Figure 9. According to the rank obtained in Friedman test, Nemenyi test shows that SVM is significantly different from NB, DT, KNN, and LDA classifiers. SVM performance is better than LR and RF.

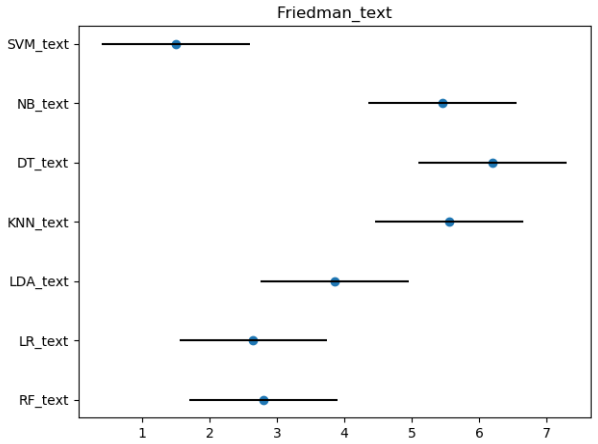

FIGURE 9. Friedman test chart of each classifier of text features. The horizontal axis is the average order value, and the vertical axis is each algorithm. Each algorithm uses a dot to display its average order value. The horizontal line segment with the dot as the center represents the size of the critical value range.

On the behavior data set. For all the calculations, the α value is set to 0.05. $\tau_{x^2_{behavior}} = 30.3$, $\tau_{F_{behavior}} = 9.2$, the critical value of F distribution $F(k-1,(k-1)(N-1)) = 2.27$, so $\tau_{F_{behavior}} > F(6,54)$, the negative hypothesis is rejected. Then perform the Nemenyi test and calculate the critical value $CD_{behavior} = 2.19$. According to the experimental results, the Friedman test chart is drawn as shown in Figure 10. According to the rank obtained in Friedman test, Nemenyi test shows that the performance of RF and SVM, NB, and KNN classifiers are significantly different, and RF performance is better than DT, LDA, and LR.

In conclusion, the statistical tests point that, between all the classifiers analyzed in this work, the SVM and RF have the best performances in the datasets. Attending to the Friedman ranks, we highlight the SVM and RF performance in the studied problem.

V. DISCUSSION

Based on the above experiments, it can be seen that the method proposed in this article is effective for fake review detection. Compared with other common supervised learning

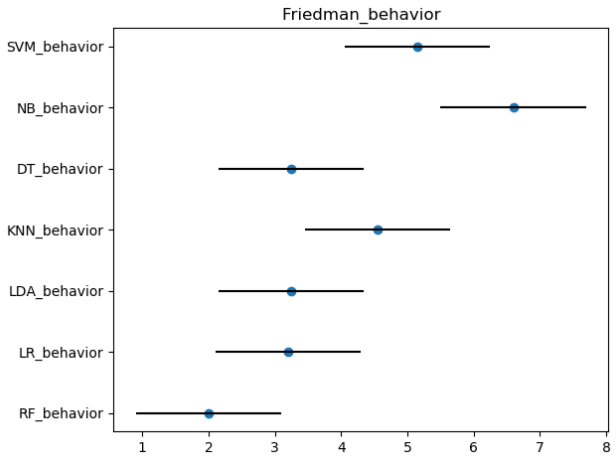

FIGURE 10. Friedman test chart of each classifier of behavior features.

models with small-labeled samples and semi-supervised learning models with large amount of unlabeled samples,this method has a better detection effect. It makes full use of a great quantity of existing of unlabeled data, and to a certain extent, reduces the huge workload of manual labeling and avoids other problems that may arise.

The method proposed in this article has the following problems to be solved:

(1) Relative to the field of information disclosure, personal information data in the e-commerce platform field is a commercial secret. It is not possible to obtain more comprehensive data for reference. It is difficult to fully describe whether a comment is true from personal information and text content, so the accuracy of fake review detection is generally not high. For example, this article can only get 1. *Date* 2. *reviewID* 3. *reviewerID* 4. *productID* 5. *Label* 6. *starrating* 7. *review*, and user dynamic IP, real-time browsing of products User key data (such as time and product browsing trajectory, etc.) can not obtain. If more data can be obtained from the user, the user's image can be portrayed to analyze the authenticity of the comment and provide new ideas for the research field.

(2) The manually compiled data set cannot fully reflect the reviews in the real world. The goal of the manually compiled data set is relatively single, but in the real world reviews, the personality of people is quite different. As for the sentiment consistency feature selected in this article, the average sentiment intensity of user ID *_gihqWuppoSHSR − qSbg5g* is 5.66 standard deviation is 5.97, and the average sentiment intensity of user ID *7clk7hJlh9U5Kkwq7IGLEw* is 13.42. The standard deviation is 12.72. It is due to everyone's economic conditions and society. Different levels also lead to different evaluation criteria. Rich and poor people may express different views on the same product, resulting in a deviation between the sentiment expressed by the user and the intensity of sentiment extracted by the algorithm. This part of the review will affect the detection result.

(3) Semi-supervised learning achieves experimental purposesis by expand the data set. Since the detection rate of labeled data is not very high, the detection rate of semi-supervised learning will not be greatly improved.

VI. CONCLUSION

In order to solve the problem that large-scale labeled datasets are difficult to obtain under the full supervision framework, this study proposed a fake review detection model based on the combination of multi-feature fusion and rolling collaborative training. Experimental results show that this method is more effective than traditional algorithms. It uses unlabeled data to improve the performance of the classification system, and has better classification accuracy. At the same time, the consistency of sentiment and score is analyzed, and the feature extraction of the review is carried out through the text representation model, and the feature fusion is combined with the external features of the text, which can effectively improve the classification effect of the classification model.

We also got a novel discovery that the characteristics of reviews will change with time. The main reason is that the writing methods of professional fake writers will be changed according to the update of the detection mechanism of the e-commerce platform, and we will try our best to bypass. Through the detection mechanism, it is more difficult to be found, and corresponding countermeasures can be made based on consumer psychology and times like shopping festivals. We hope that the next researchers can continue their research from the direction of dynamic update detection strategy. In the future, we will strive to find a more effective and accurate detection method that can detect false information in multiple fields, including fake information, fake news, and rumors. Through my research in this field, I think that future research can be studied in depth from the perspective of fake review criteria and social network models. In social networks, we can find the publisher of fake information through the fake information propagation path, and even find the key nodes of fake information propagation to solve the problem from the root cause.

REFERENCES

[1] G. Lackermair, D. Kailer, and K. Kanmaz, "Importance of online product reviews from a consumer's perspective," *Adv. Econ. Bus.*, vol. 1, no. 1, pp. 1–5, 2013.

[2] D. S. Kostyra, J. Reiner, M. Natter, and D. Klapper, "Decomposing the effects of online customer reviews on brand, price, and product attributes," *Int. J. Res. Marketing*, vol. 33, no. 1, pp. 11–26, Mar. 2016.

[3] S. Ullrich and C. B. Brunner, "Negative online consumer reviews: Effects of different responses," *J. Product Brand Manage.*, vol. 24, no. 1, pp. 66–77, Mar. 2015.

[4] S. Deng, C.-X. Wan, A.-H. Guan, and H. Chen, "Deceptive reviews detection of technology products based on behavior and content," *J. Chin. Comput. Syst.*, vol. 36, no. 11, p. 2498, 2015.

[5] D. Radovanovic and B. Krstajic, "Review spam detection using machine learning," in *Proc. 23rd Int. Sci.-Prof. Conf. Inf. Technol.*, Feb. 2018, pp. 1–4.

[6] H. Deng, L. Zhao, N. Luo, Y. Liu, G. Guo, X. Wang, Z. Tan, S. Wang, and F. Zhou, "Semi-supervised learning based fake review detection," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. Appl.*, Dec. 2017, pp. 1278–1280.

[7] H. Ahmed, I. Traore, and S. Saad, "Detecting opinion spams and fake news using text classification," *Secur. Privacy*, vol. 1, no. 1, p. e9, 2018.

[8] M. Crawford, T. M. Khoshgoftaar, J. D. Prusa, A. N. Richter, and H. Al Najada, "Survey of review spam detection using machine learning techniques," *J. Big Data*, vol. 2, no. 1, p. 23, Dec. 2015.
[9] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proc. Int. Conf. Web Search Data Mining*, 2008, pp. 219–230.
[10] K. H. Yoo and U. Gretzel, "Comparison of deceptive and truthful travel reviews," in *Proc. ENTER*, 2009, pp. 37–47.
[11] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," 2011, *arXiv:1107.4557*. [Online]. Available: http://arxiv.org/abs/1107.4557
[12] C. Tang and L. Guo, "Digging for gold with a simple tool: Validating text mining in studying electronic word-of-mouth (eWOM) communication," *Marketing Lett.*, vol. 26, no. 1, pp. 67–80, Mar. 2015.
[13] S. Feng, R. Banerjee, and Y. Choi, "Syntactic stylometry for deception detection," in *Proc. 50th Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, 2012, pp. 171–175.
[14] C. Yanfang and L. Zhiyu, "Research on product review attribute-based of emotion evaluate review spam detection," *Data Anal. Knowl. Discovery*, vol. 30, no. 9, pp. 81–90, 2014.
[15] J. Li, C. Cardie, and S. Li, "Topicspam: A topic-model based approach for spam detection," in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, 2013, pp. 217–221.
[16] W. Sun, Y. Guo, Z. Fan, and X. Xu, "False comment recognition based on the combination of content features and user behavior characteristics," in *Proc. 3rd IEEE Int. Conf. Comput. Commun. (ICCC)*, Dec. 2017, pp. 1510–1514.
[17] Y. Ren and D. Ji, "Neural networks for deceptive opinion spam detection: An empirical study," *Inf. Sci.*, vols. 385–386, pp. 213–224, Apr. 2017.
[18] L. Li, B. Qin, W. Ren, and T. Liu, "Document representation and feature combination for deceptive spam review detection," *Neurocomputing*, vol. 254, pp. 33–41, Sep. 2017.
[19] W. Zhang, Y. Du, T. Yoshida, and Q. Wang, "DRI-RCNN: An approach to deceptive review identification using recurrent convolutional neural network," *Inf. Process. Manage.*, vol. 54, no. 4, pp. 576–592, Jul. 2018.
[20] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, "Detecting product review spammers using rating behaviors," in *Proc. 19th ACM Int. Conf. Inf. Knowl. Manage.*, 2010, pp. 939–948.
[21] F. Wu, J. Shu, Y. Huang, and Z. Yuan, "Co-detecting social spammers and spam messages in microblogging via exploiting social contexts," *Neurocomputing*, vol. 201, pp. 51–65, Aug. 2016.
[22] G. Wang, S. Xie, B. Liu, and P. S. Yu, "Review graph based online store review spammer detection," in *Proc. IEEE 11th Int. Conf. Data Mining*, Dec. 2011, pp. 1242–1247.
[23] Z. Wang, S. Gu, X. Zhao, and X. Xu, "Graph-based review spammer group detection," *Knowl. Inf. Syst.*, vol. 55, no. 3, pp. 571–597, Jun. 2018.
[24] S. Xie, G. Wang, S. Lin, and P. S. Yu, "Review spam detection via temporal pattern discovery," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 823–831.
[25] R. Yafeng, J. Donghong, Z. Hongbin, and Y. Lan, "Deceptive reviews detection based on positive and unlabeled learning," *J. Comput. Res. Develop.*, vol. 52, no. 3, p. 639, 2015.
[26] X. Wang, K. Liu, and J. Zhao, "Detecting deceptive review spam via attention-based neural networks," in *Proc. Conf. Natural Lang. Process. Chin. Comput.*, 2017, pp. 866–876.
[27] N. Jain, A. Kumar, S. Singh, C. Singh, and S. Tripathi, "Deceptive reviews detection using deep learning techniques," in *Proc. Int. Conf. Appl. Natural Lang.*, 2019, pp. 79–91.
[28] Y. Fang, H. Wang, L. Zhao, F. Yu, and C. Wang, "Dynamic knowledge graph based fake-review detection," *Int. J. Speech Technol.*, vol. 15, pp. 1–15, Jul. 2020.
[29] F. Masood, A. Almogren, A. Abbas, H. A. Khattak, I. U. Din, M. Guizani, and M. Zuair, "Spammer detection and fake user identification on social networks," *IEEE Access*, vol. 7, pp. 68140–68152, 2019.
[30] A. Rastogi and M. Mehrotra, "Impact of behavioral and textual features on opinion spam detection," in *Proc. 2nd Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, Jun. 2018, pp. 852–857.
[31] R. Barbado, O. Araque, and C. A. Iglesias, "A framework for fake review detection in online consumer electronics retailers," *Inf. Process. Manage.*, vol. 56, no. 4, pp. 1234–1244, Jul. 2019.
[32] J. Li, Q. Ma, C. Yuan, W. Zhou, J. Han, and S. Hu, "Fusion convolutional attention network for opinion spam detection," in *Proc. Int. Conf. Neural Inf. Process.*, 2019, pp. 223–235.
[33] Y. Chung, P. J. Haas, E. Upfal, and T. Kraska, "Unknown examples & machine learning model generalization," 2018, *arXiv:1808.08294*. [Online]. Available: http://arxiv.org/abs/1808.08294
[34] R. de Castro Oliveira, E. C. Gonçálves Dá Rós Baldam, F. Reis da Costa, and A. S. Pelissari, "The effect of perceived usefulness of online reviews on hotel booking intentions," *Revista Brasileira Pesquisa Turismo*, vol. 14, no. 2, pp. 1–5, 2020.
[35] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 1188–1196.
[36] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[37] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification Regression Trees*. Boca Raton, FL, USA: CRC Press, 1984.
[38] A. McCallum and K. Nigam, "A comparison of event models for naive Bayes text classification," in *Proc. AAAI Workshop Learn. Text Categorization*, 1998, vol. 752, no. 1, pp. 41–48.
[39] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967.
[40] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[41] R. E. Wright, "Logistic regression," in *Reading and Understanding Multivariate Statistics*. Washington, DC, USA: American Psychological Association, 1995, pp. 217–244.
[42] S. Balakrishnama and A. Ganapathiraju, "Linear discriminant analysis-a brief tutorial," in *Inst. for Signal Inf. Process.*, vol. 18, no. 1998, 1998, pp. 1–8.
[43] C. Beeri, P. A. Bernstein, and N. Goodman, "A sophisticate's introduction to database normalization theory," in *Readings in Artificial Intelligence and Databases*. Amsterdam, The Netherlands: Elsevier, 1989, pp. 468–479.
[44] E. Elmurngi and A. Gherbi, "An empirical study on detecting fake reviews using machine learning techniques," in *Proc. 7th Int. Conf. Innov. Comput. Technol. (INTECH)*, Aug. 2017, pp. 107–114.
[45] J. K. Rout, A. Dalmia, K.-K.-R. Choo, S. Bakshi, and S. K. Jena, "Revisiting semi-supervised learning for online deceptive review detection," *IEEE Access*, vol. 5, pp. 1319–1327, 2017.
[46] Y.-R. Chen and H.-H. Chen, "Opinion spam detection in Web forum: A real case study," in *Proc. 24th Int. Conf. World Wide Web*, 2015, pp. 173–183.
[47] A. Mukherjee, V. Venkataraman, B. Liu, and N. S. Glance, "What yelp fake review filter might be doing?" in *Proc. ICWSM*, 2013, pp. 409–418.
[48] S. Rayana and L. Akoglu, "Collective opinion spam detection: Bridging review networks and metadata," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2015, pp. 985–994.
[49] H. Li, G. Fei, S. Wang, B. Liu, W. Shao, A. Mukherjee, and J. Shao, "Bimodal distribution and co-bursting in review spam detection," in *Proc. 26th Int. Conf. World Wide Web*, Apr. 2017, pp. 1063–1072.
[50] Z. You, T. Qian, and B. Liu, "An attribute enhanced domain adaptive model for cold-start spam review detection," in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 1884–1895.
[51] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.

JINGDONG WANG was born in Changchun, Jilin, China, in 1980. He received the B.E. and M.E. degrees in computer science and technology from Northeast Electric Power University, Jilin City, and the Ph.D. degree in information science from the University of Science and Technology of China, in 2017.

He has been working with Northeast Electric Power University, since 2008. From 2008 to 2011, he was a Teaching Assistant. From 2011 to 2016, he was a Lecturer. Since 2017, he has been an Associate Professor with the School of Computer Science. He is the author of more than 30 articles. His research interests include public security, natural language processing, text mining, knowledge graph, and other aspects, involve software engineering, artificial intelligence, emotional analysis, and other fields.

HAITAO KAN was born in Shandong, China, in 1996. He received the bachelor's degree from Qingdao Agricultural University, in 2018. He is currently pursuing the master's degree with Northeast Electric Power University. His research interests include artificial intelligence and data mining.

FANQI MENG was born in Tongliao, Inner Mongolia, China, in 1981. He received the B.E. degree in computer science and technology from Northwest Agriculture and Forestry University, Xianyang, in 2003, the M.E. degree in computer application technology from Northeast Electric Power University, Jilin City, in 2010, and the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, in 2018.

He has been working with Northeast Electric Power University, since 2003. From 2003 to 2009, he was a Teaching Assistant. From 2009 to 2018, he was a Lecturer. Since 2019, he has been an Associate Professor with the School of Computer Science. He is the author of two textbooks and more than 30 articles. His research interests include software safety, natural language processing, fault diagnosis of electric power equipment and other aspects, involve software engineering, artificial intelligence, data mining, and other fields. He received the Science and Technology Progress Award of Jilin Province, in 2010, 2011, 2017, and 2018.

QIZI MU was born in Datong, Shanxi, China, in 1996. She received the B.E. degree from the Ningbo University of Technology, in 2018. She is currently pursuing the master's degree with Northeast Electric Power University. Her research interests include data mining and artificial intelligence, involve influence maximization and link prediction in complex networks.

GENHUA SHI was born in Changchun, Jilin, China, in 1996. He is currently pursuing the master's degree in computer science and technology with Northeast Electric Power University. His research interests include public security and natural language processing, involve artificial intelligence, emotional analysis, and other fields.

XIXI XIAO was born in Shangqu, Henan, China, in 1995. She received the B.E. degree from the Xinlian College, Henan Normal University, in 2018. She is currently pursuing the master's degree with Northeast Electric Power University. Her research interests include online public opinion and artificial intelligence.