Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 9034773, 11 pages
https://doi.org/10.1155/2022/9034773

Research Article

Analysis of Internet Public Opinion Popularity Trend Based on a Deep Neural Network

Min Gong

King's College London, London WC2R 2LS, UK

Correspondence should be addressed to Min Gong; k21040342@kcl.ac.uk

Received 28 March 2022; Revised 3 May 2022; Accepted 5 May 2022; Published 6 July 2022

Academic Editor: Gengxin Sun

In the context dominated by Internet communication, people's various emotions can be clearly reflected through network public opinion, whether it is the view of political affairs, the preference for entertainment, or the demand for life. This also allows management or providers to meet their needs more specifically. Based on today's need to understand the trend of Internet public opinion, this paper describes a deep neural network (DNN). A deep neural network is a machine learning model that is the foundation of deep learning and has a strong ability to mine potential information in data. By improving the loss function of the neural network, this paper reduces the influence of unbalanced data on the classification results and improves the classification effect of the model on a small number of categories. Aiming at the different lengths of Internet text, a more robust model of text sentiment classification is proposed, which makes the HCB-Att model better extract the local information and contextual information of the text. Finally, through comparative experiments, the optimization model used in this paper is proved to be effective for the analysis of network public opinion sentiment.

1. Introduction

Deep belief network prediction training algorithms are a robust and useful method for generating deep neural networks, optimizing, and reducing generalization errors. We use this algorithm for speech recognition in a large vocabulary context, and the model has made recent advances in telephone recognition using deep belief networks. It describes the key components of the model by applying CD-DNN-HMM to the LVSR acquisition process and analyzes the impact of various modeling options on performance [1]. This paper provides an entry point for interpreting deep neural network models and for interpreting their predictions. It is based on the tutorial given on the ICASSP 2017. There is a huge need for improved Reynolds mean turbulence models, informed by a richer set of turbulence physics, that can represent a richer turbulence physics. In this paper, we propose a method to learn a Reynolds stress anisotropy tensor model from simulation data with high fidelity using deep neural networks. We propose a new neural network architecture that beds Galileo invariance into the predicted anisotropic tensor using a multiplication layer with an invariant tensor basis. This Reynolds stress anisotropy prediction of the invariant neural network is propagated to the velocity field of the two test cases [2]. During the DNN learning process, large training sets ensure powerful modeling capabilities to estimate complex nonlinear mappings from observed noisy speech to desired clean signals. The acoustic context was found to improve the continuity of speech, successfully separating it from background noise without the annoying musical artifacts common to traditional speech enhancement algorithms. A series of pilot experiments were conducted under multicondition training and simulating voice data for more than 100 hours. In addition, in an assessment of subjective preferences of 10 listeners, it was found that 76.35% of subjects preferred DNN-based enhanced speech over voice obtained using other traditional techniques [3]. In view of the frequent occurrence of network public opinion on the Internet, this paper analyzes the requirements and business processes of the network public opinion monitoring and analysis system. This paper introduces the five-tier architecture of the information monitoring and analysis system, including automatic information collection, information preprocessing, information database, information analysis and service, information visualization, and reporting. Then, the implementation of key technologies in the system is studied, including public opinion information collection technology, information search analysis, information search and cluster analysis, speech analysis, and trend prediction analysis [4]. Compared with the original system, the accuracy of topic extraction and public opinion trend prediction of the redesigned system has been significantly improved, which further verifies the correctness of the scheme. A new model is built to describe public opinion events that change in thermal trends. The results show that the model can simulate the hot trend of public opinion events. It provides a quantitative basis for the government to control and guide public opinion and development plans. Audio messages, video messages, and facial expression messages are inversely correlated with new media information preferences. The volume is indexed by repeated questions in polls as well as variations of these questions. These questions came from the U.S. national sample conducted by the AIPO (Gallup), NORC, and Roper (mainly the general population) and have been conducted in the Roper Archived by the Center for Public Opinion Research. More than 90% of the basic data for the survey is only available from the Roper Center. The index was first surveyed in September 1936, the most recent in October 1973. The purpose of the index is to build a database for measuring and analyzing social change. 70 main topics include specific research questions from trends in agricultural farm incomes to U.S. attitudes toward Yugoslavia. Each item includes the wording of the question, the investigative body that raised the question, the number and date of the investigation, and the number of questions [5]. Social media, as an adjunct to researchers, have become an important communication channel for public opinion because of their popularity and rapid communication ability because they serve as the main virtual space to raise new public issues, discuss these issues, and put forward opinions on these issues. With the recognition of the importance of social media, many businesses and governments have tried to use social media as a marketing tool or a public opinion perception tool. In the aftermath of the Fukushima disaster, nuclear power has become one of the hot public issues in South Korea. Nuclear power is a double-edged sword because it is currently the most efficient way to generate electricity, and there are potential risks, such as radiation leakage. Governments and policymakers need to constantly monitor public perceptions of nuclear power, as they may change depending on the events [6]. In this study, we aim to propose a method to sense the public perception of nuclear power and analyze its trends based on public opinion mining techniques. We have also proposed a measure to track changes in the direction of public opinion on nuclear power. To demonstrate the utility of the proposed method, we collected tweets from South Korea Nuclear Power in 2009 and 2013. After categorizing tweets in 2009 and 2011, an emotional dictionary containing both positive and negative terms was constructed. Performance tests conducted using tweets in 2012 and 2013 showed that the method could be applied in practice. [7]. The social network Weibo's opinion research on Chinese medicine during the epidemic in novel coronavirus pneumonia (novel coronavirus pneumonia) comprehensively considers the research background of the people's fight against novel coronavirus pneumonia and the power of Chinese medicine during the epidemic to determine the research theme and research object. The timeline is divided into three stages based on the overall thermal change. In order to explore and compare people's emotions and concerns about TCM in different stages, deep learning analysis methods such as sentiment analysis and latent Dirichlet distribution analysis are used for large flow of people. The study found that the public’s positive “emotional component” towards TCM increased significantly on the timeline. At the same time, the public’s autonomy has increased, and overall public opinion has begun to show an upward trend [8]. The whole society pays great attention to it. With the continuous development of Internet technology, especially the advent of the self-media era, public opinion, as a public force, has brought a comprehensive impact and new challenges to the management of the college entrance examination. When disseminating public opinion on the college entrance examination, the organs of the college entrance examination shall understand and study the role and discipline of the media. Learn to contact the media and cooperate for win-win results. Build good relationships with media resources and networks. Adhere to the problem-oriented, and actively build a professional public opinion response mechanism. In particular, the focus is on the disposal in the context of new media public opinion. Facts have proved that timely, reasonable, and appropriate response is the crux of public opinion response and crisis management [9]. It can predict the heat of public opinion, proactively guide public opinion in time, and let the public accept and form a positive cognitive impression, which is conducive to the healthy development of public opinion. In this paper, the staging factor is introduced into the wheel selection method of the artificial swarm algorithm to improve the global search ability of ABC and to predict the hot trend of network public opinion based on the improved ABC-BP model. This model has a higher prediction accuracy when compared to other models. The above theoretical research can provide an effective model for the government and enterprises to timely grasp the hot trend of online public opinion and provide a reference for formulating feasible guidance measures and marketing strategies and added up to obtain the predicted values of raw data. Finally, to account for the accuracy and validity of the EEMD-NAR model [10], a new model is built to describe public opinion events that change in thermal trends. The results show that the model can simulate the hot trend of public opinion events. In the past era, with the explosive interaction and dissemination of Internet information, collecting a large amount of Internet information and mining hot topics of network public opinion have become a hot branch of research. In this paper, a neural network (NN) based network public opinion evaluation model and prototype are proposed. First of all, a new evaluation index system is proposed and designed.

Subsequently, we proposed a new poll analysis evaluation model and discussed the detailed steps. Finally, we conducted experiments to test the robustness and effectiveness of the method, and then we proposed a detailed explanation. As the number of Weibo users increases, the platform generates a large amount of data every day [11]. The rapid spread of data will lead to widespread changes in public opinion. Therefore, to correctly predict the public opinion on Weibo is an urgent challenge. To this end, this paper proposes a method to predict microblog public opinion using improved BP. First of all, according to the characteristics of the microblog public opinion, this paper constructs 9 public opinion indicators to analyze microblog public opinion. Third, since BP is susceptible to initial weight selection and shows a poor convergence rate, a genetic algorithm is introduced to optimize BP. However, genetic algorithms can easily fall into local optimal solutions [12]. Therefore, using the urban acceptance criterion to improve the local search capability of genetic algorithms, we propose an IGABP algorithm based on improved genetic algorithms. Finally, the effectiveness of the IGABP algorithm is verified by extracting and normalizing microblog data. Experimental results show that the IGABP algorithm is feasible in the prediction process [13]. The establishment of an effective network public opinion risk early warning index system and the evaluation method is an important basis for an emergency decision. Automated and rapid online poll assessments can greatly reduce human output. In order to establish an evaluation method suitable for the development level of network public opinion, a comprehensive evaluation algorithm combining the BP neural network and hierarchical analysis (AHP) is proposed. Besides, the possible optimization method is also discussed [14].

2. Key Technologies for Event Analysis

2.1. Event Recognition Technology

2.1.1. Hierarchical Clustering. Hierarchical clustering is a clustering algorithm that divides datasets into different levels based on the distance between clusters, resulting in a tree-shaped clustering structure. Depending on the division strategy, it can be divided into "top-down" and "bottom-up" [15].

Common calculation formulas for measuring distances between clusters are minimum distance, maximum distance, and average distance:

$$\begin{aligned} d_{\min}\left(C_i, C_j\right) &= \min_{x\in C_i, z\in C_j} \operatorname{dist}(x, z), \\ d_{\max}\left(C_i, C_j\right) &= \max_{x\in C_i, z\in C_j} \operatorname{dist}(x, z), \\ d_{avg}\left(C_i, C_j\right) &= \frac{1}{\left|C_i\right|\left|C_j\right|} \sum_{x\in C_i} \sum_{z\in C_j} \operatorname{dist}(x, z). \end{aligned} \tag{1}$$

The researchers optimized hierarchical clustering and proposed algorithms such as BIRCH and CURE. Each node of a cluster feature tree consists of three cluster features, including the number of samples, the vector of sample points, and the sum of squares of the sample point features [16]. The CURE algorithm reduces the effect of noise on clustering and optimizes the hierarchical clustering algorithm by choosing a fixed number of representative points to represent the class.

2.1.2. Nearby Propagation Clusters. In the AP algorithm, the representative points of the class are selected from the data points.

$$z = \sum_{i=1}^{n} s\left(x_i, x_k\right), \tag{2}$$

where x_i data point is the representative point of the data x_k and s(x_i, x_k) is the similarity between the two points. In the neighbor propagation clustering algorithm, select representative points from the data points:

$$s(i, j) = -\left\|x_i - x_j\right\|, \tag{3}$$

where x_i and x_j are data points.

The degree of attraction $r(i, k)$ indicates the accumulation of information to consider other potential representative points x_i x_k, indicating the x_k x_i appropriate degree of representation as a representative point.

The attractiveness $r(i, k)$ indicates the appropriate degree of accumulation of information as a representative point, taking into account other potential representative points.

$$r(i, k) = s(i, k) - \max_{k'\neq k}\left\{a\left(i, k'\right) + s\left(i, k'\right)\right\}. \tag{4}$$

The degree of attribution $a(i, k')$ represents the appropriate degree to select a representative point as a representative point, taking into account other potential representative points. The algorithm iteratively updates the values of $r(i, k)$ and $a(i, k')$ to find representative points for the data points [17]. The formula is as follows:

$$a(i, k) = \begin{cases} \min\left\{0, r(k, k) + \sum\limits_{i'\notin\{i,k\}} \max\left\{0, r\left(i', k\right)\right\}\right\}, & i \neq k, \\ \sum\limits_{i'\notin\{i,k\}} \max\left\{0, r\left(i', k\right)\right\}, & i = k. \end{cases} \tag{5}$$

The AP algorithm describes the probability that a data point becomes a representative point by the degree of preference (preference) $p = s(k, k)$. The larger the p value, the more likely the point is to become a representative point.

The attraction matrix r and attribution matrix a are initialized to a zero matrix and then selected in the order in which the attraction is updated first, and then the attribution is updated. After each round of updates, the choice of representative points is judged in a combination of attractiveness and attribution. The algorithm stops when the class boundary is updated or the generation number is reached. Another parameter in the neighbor propagation algorithm is the damping factor. In each generation, the updated results of the information are obtained by multiplying the results of the previous update by the sum of 1 of the values updated by this update. The damping coefficient avoids numerical oscillations and is used to control the convergence effect of the algorithm. The value range is [0, 1], usually set to 0.5.

$$\begin{aligned} r_{t+1}(i,k) &= \lambda * r_t(i,k) + (1-\lambda) * r_{t+1}(i,k), \\ a_{t+1}(i,k) &= \lambda * a_t(i,k) + (1-\lambda) * a_{t+1}(i,k). \end{aligned} \tag{6}$$

The algorithm has no requirements for the symmetry of the similarity matrix. The algorithm uses a wider range of application scenarios, clustering according to the degree of preference, and there is no need to know the number of clusters in advance. Useful when clustering cannot be scoped based on prior knowledge.

2.1.3. FastText Model. The model structure is shown in Figure 1. The structure of the FastText model is similar to CBOW. The model sums word vectors in the hidden layer by mapping words in the document to the vector input hidden layer. Unlike CBOW, the output layer changes from output intermediate to output classification labels [18].

N-gram features describe local word order and preserve morphological features within the vocabulary. The FastText model extracts the N-gram characteristics of words in sentences. Since storing N-gram features consumes a lot of storage space, the model uses a hash function to map feature vectors, record them as input layer inputs, and then map feature vectors to the n-dimensional hidden layer by linear transformation. The model uses hierarchical softmax to calculate the probability distribution of a predefined class, with the goal of minimizing the value of the log-likelihood function as follows:

$$L = -N \sum_{n=1}^{N} y_n \log\left(f\left(BAx_n\right)\right), \tag{7}$$

where x_n denote the standardized document feature vectors are y_n classification labels and A and B are the weight matrices.

If there are many text classification tags, the calculation cost of using softmax directly is relatively high. Therefore, the model uses hierarchical softmax to transform the global multiclassification problem into multiple binary classification problems by building a Huffman tree, thereby reducing the computational training complexity from $I(kh)$ to $O(kh)$. At the same time, the use of hierarchical softmax also improves the speed of the model in use.

2.2. Keyword Extraction Techniques. TextRank is a graph-based unsupervised text sorting model that is widely used in the field of text summary and keyword generation. The advantages of TextRank include the fact that there is no need to study documentation beforehand, and implementation is simple. The main idea is to divide the document with words or sentences and to use the divided words or sentences as nodes to construct the network diagram. Node points are connected to each other according to the similarity relationship, and the weight of the edge is the similarity between nodes. The weights of the nodes are updated iteratively, resulting in keywords or abstract sentences [19] depending on the size of node weights. The TexTank model builds a graph with a rightward figure of $G = (V, E)$, where V is the node set, E is the edge set, and E is a subset of x. The weight of the sum of any two points in the figure, which indicates the degree of similarity between the two points, $W_{ij} = W_{ji}$.

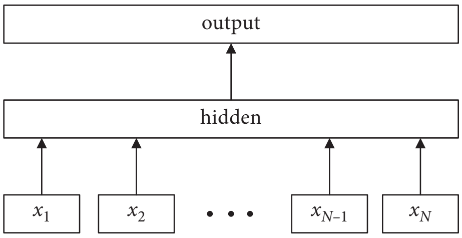

Figure 1: FastText structure diagram.

The V_i score for points is calculated as

$$S\left(V_i\right) = (1-\alpha) + \alpha \sum_{V_j \in \text{in}\left(V_i\right)} \frac{W_{ij}}{\sum_{V_k \in \text{out}\left(V_i\right)} W_{jk}} s\left(V_j\right), \tag{8}$$

where (V_i) is the node set that points to the point and $\text{out}(V_i)$ is the node set that points to the node. α is a damping factor, with a value range of [0, 1], which indicates the probability that a node points to another node, and a typical value of 0.85.

3. Neural Network Model Based on Adjusting Loss Function

3.1. Deep Neural Network Model Construction. The overall steps of vectorized data processing and model building are shown in Figure 2.

The feed-forward neural network model framework is shown in Figure 3.

(1) Text vector calculation $\hat{y}$: because the feed-forward neural network only receives the numerical tensor, requiring the continuity of vector values, in the data processing stage before the input classifier, it is necessary to distribute words, convert them into a continuous numerical vector, then numerically calculate the word vector corresponding to each word in the sentence, convert each comment text into a large vector, and enter it into the classifier for processing for the last time. The most commonly used method of text vector calculation is the direct averaging method of word vector, that is, after text data is segmented, the word vector training is performed on each word through the word2vec model, then the corresponding word vector of each word is added, and then the corresponding position is averaged to obtain a vectorized representation of each comment data [20]. This approach means that all words in a text are considered equally important, and it may not be possible to exclude the effects of words that have little impact on the classification result. Use the text parting tool Jeba to segment preprocessed text, while removing stop words, and train by the

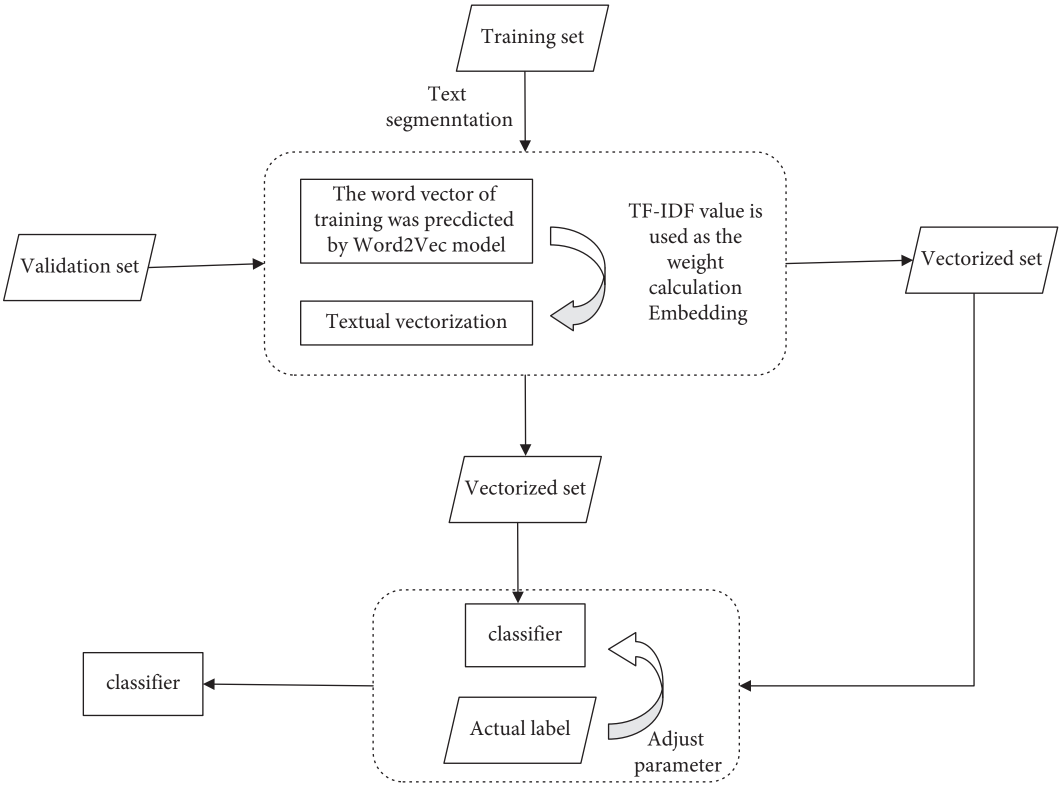

Figure 2: Schematic diagram of model building.

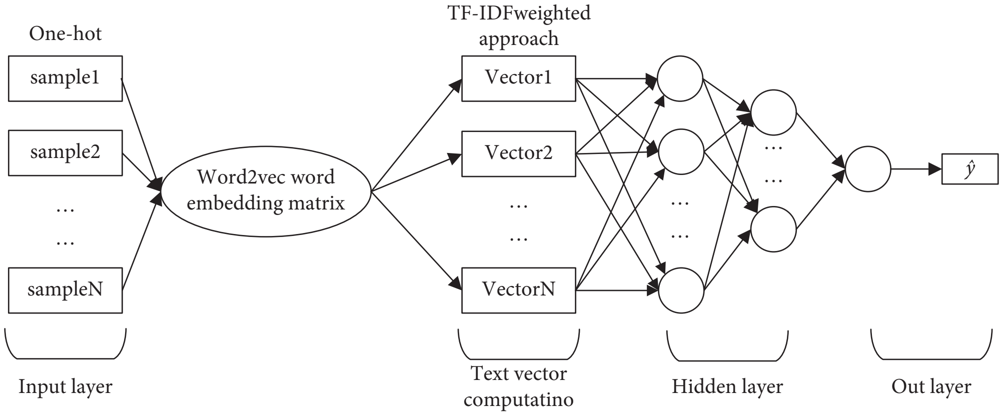

Figure 3: Feed-forward neural network construction.

word2vec model. Get the N-dimensional word vector for each word V_i (the total number of words is set to W), $V_i = (v_1, v_2, \ldots, v_N) \in (1, 2, \ldots, M)$, and for each piece of the text D_j $(j = 1, 2, \ldots, M)$, M represents the number of text data bars. Calculating the TF-IDF weight $T(t, D)$ of each word and the number of words n, the final vectorization of each text is expressed as

$$v_j = \frac{\sum_{t\in D_j} V_t \times T\left(t, D_j\right)}{n_j}. \tag{9}$$

(2) A hidden layer neural network maps an input (text vector) to a target (label) through a series of simple data transformations (layers). The learning process is to find a set of weight values for all layers of the neural network, so that the network can vectorize the text data corresponding to each input and its correct target output value one by one [21], and the distributed text vector is input to the hidden layer, the output of the feed-forward network of the first hidden layer has the following form:

$$NN_{MLP1}(v) = g(v \cdot W + b). \tag{10}$$

(3) where $v \in R^{d_{\text{in}}}$, $W \in R^{d_{\text{in}} d_{\text{out}}}$, and $b \in R^{d_{\text{out}}}$. W and b are the weight matrix and the bias term, respectively, and g is activation function.

(4) The disadvantage of the tanh function is that when the input value is very large or very small, the derivative will approach 0, making the gradient update very slow, causing the gradient to disappear. The Remu function is a piecewise linear function in the form:

$$a(z) = \max(0, z). \tag{11}$$

(5) The advantage of the Remu function is that the computation speed is fast, and it compensates for the disappearance of the gradient of the tanh function. The next calculation of the second layer also follows the transformation from above.

(6) Output layer: for the problem for multiclassification problems, the neural network is utilized. The softmax transform converts it to a probability distribution:

$$p_i = \mathrm{softmax}(a_i) = \frac{\exp(a_i)}{\sum_{i=1}^{c}\exp(a_i)}. \tag{12}$$

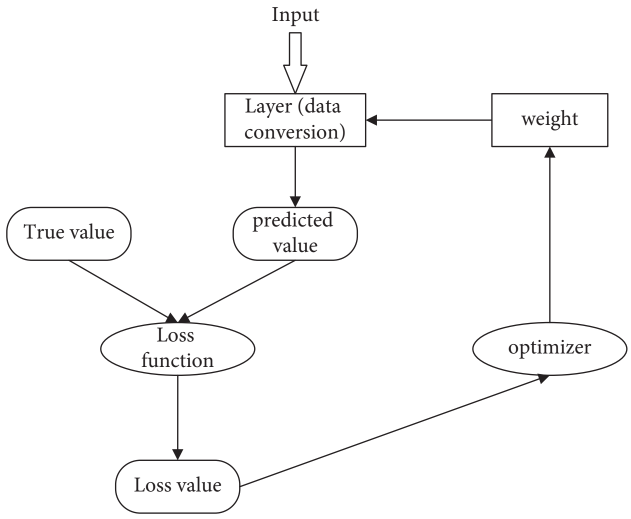

Figure 4: The process by which a neural network implements predictions.

3.2. Improved Loss Function. The data transformation operations performed by each layer of input data in the neural network are stored in the weights of that layer, and deep learning is actually looking for a set of weights for all layers of the neural network, so that the network can correspond input values of each sample to its true value one by one, which is the work of loss function of the neural network. The neural network uses this distance value to adjust weights to reduce the current corresponding loss value,as shown in Figure 4.

For classification problems, the magnitude of the error between the predicted and true values is measured using loss function in the neural network, and weight parameters in the network are updated with each training of the model, with the ultimate training goal of minimizing this error [22]. For different problems, it is necessary to define loss function according to actual situation, for example, some classification problems in the real scene belong to multiple classifications, and it is necessary to construct multiclassification loss function, and for example, some data have distribution imbalance problems, and the sample weight or classification difficulty problem in loss function needs to be considered. The cross-entropy loss function of the two classes are as follows:

$$L = \begin{cases} -\log p, \text{ if } y = 1, \\ -\log(1 - p), \text{ otherwise.} \end{cases} \tag{13}$$

The problem of categorical imbalance is also a manifestation of difference in classification difficulty. Thus, in the loss function of a neural network, by reducing the weights of the easier to classify samples, the model can focus more on the difficult to classify samples. Lin et al. proposed that focus loss aims to solve the problem of category imbalance in binary classification:

$$L_{\hbar} = \begin{cases} -(1 - p)^{\gamma}\log p, \text{ if } y = 1, \\ -p^{\gamma}\log(1 - p), \text{ otherwise,} \end{cases} \tag{14}$$

where p is the probability that the model estimates that the sample belongs to the true category. For modulation coefficients, when samples are correctly classified, modulation coefficients are relatively small and contribute less to the total loss, that is, modulation coefficient reduces loss contribution of easily divided samples, so that the model focuses on correcting misclassified samples. γ is a constant greater than or equal to zero that determines the magnitude of modulation coefficient's role in the loss, the greater the modulation coefficient's effect, and when $\gamma = 0$, the focal loss is equivalent to the cross-entropy loss [23].

In addition, the researchers found following weight adjustments for the focus loss, and the experimental results showed a slight improvement:

$$L_{\hbar} = \begin{cases} -\alpha \cdot (1 - p)^{\gamma}\log p, \text{ if } y = 1, \\ -(1 - \alpha) \cdot p^{\gamma}\log(1 - p), \text{ otherwise.} \end{cases} \tag{15}$$

Experimental data on Lin et al. showed that positive samples are a minority. The model works best when $\alpha = 0.25$ and $\gamma = 2$, with the effect of modifying modulation coefficients to avoid the relatively small weights assigned to easily classified samples, which should be γ slightly reduced when slightly increased α. Now, considering the multiclassification case, traditional loss function is defined as follows:

$$L = -\sum_{c=1}^{M} y_c \log(p_c). \tag{16}$$

Assuming M is the number of categories, it is the indicator variable, and the form of the category label y is one-hot. For sample x, if the real number category is class c, then the cth bit of the vector is 1, and the remaining bits are 0. The probability that the sample is predicted for each class is as follows: now, use the idea of focus loss to solve the multiclass problem, referring to multiclass cross-entropy loss function form, the multiclass focus loss is defined as follows:

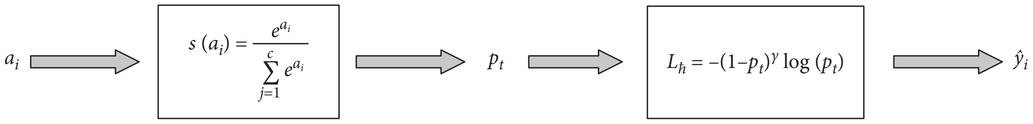
a_i
$s(a_i) = \frac{e^{a_i}}{\sum_{j=1}^{c} e^{a_i}}$
p_t
$L_\hbar = -(1-p_t)^\gamma \log(p_t)$
$\hat{y}_i$

Figure 5: Forward computation of the neural network under focus loss.

$$L_\hbar = -\sum_{i=1}^{c} y_i \cdot (1 - p_i)^\gamma \log(p_i). \tag{17}$$

It converts the category label y of data to the form one-hot, c is the number of categories, and p is the probability that the sample will be predicted as a category. To further balance the interference caused by the sample scale, a weighted factor $a1$ can also be introduced, which is used to represent the probability that the sample predicts correctly in p, and L can also be expressed as

$$L_\hbar = -\alpha_t (1 - p_t)^\gamma \log(p_t). \tag{18}$$

For weight adjustment parameter A, if it is applied to cross-entropy loss function, it can be set to inverse frequency of each category, or the optimal value can be obtained by cross-validation. There are two parameters in the focal damage, Y and A are one and the other relationship and A is more of a correction effect, which need to be through the cross-validation of Lin et al. to vigorously adjust its optimal value, after adjusting the parameters. The value of a has only a slight improvement in the classification effect [24] and can be thought as 1, and loss function used by the model is shown in equation (17). In the case of using focus loss function, the forward computation process of the neural network is shown in Figure 5:

The fully connected layer a_i for the ith output, the softmax layer is later linked, the predicted probability values of each category are calculated, and the focus loss of the calculation process is updated backwards as follows:

$$\frac{\partial L_\hbar}{\partial a_i} = \begin{cases} -(1 - p_t)^\gamma \cdot (1 - p_t - \gamma p_t \log\, p_t),\ i = t, \\ p_i \cdot (1 - p_t)^{\gamma - 1} \cdot (1 - p_t - \gamma p_t \log\, p_t),\ i \neq t. \end{cases} \tag{19}$$

4. Experimental and Algorithmic Evaluation Depth Prices

4.1. Experimental Evaluation Indicators. Overall, the model is more accurate in its evaluation metrics in the field of text sentiment analysis (accuracy) and recall. They all represent different performances, but the same is that the higher the value, the better the performance of the classifier.

The above evaluation criteria are generally used for binary classification problems, and data sets used in the experimental section of this chapter clearly indicate that multiple categories will be identified; so based on this consideration, the evaluation criteria under this type will be described as follows. Considering the multicategory problem as a number of binary classification problems, assuming that a certain text data can be clearly distinguished into three categories A, B, and C, then for the research object, a class, there is an elephant Table 1. This is a kind of confusion matrix.

Therefore, the calculation formula for each evaluation criterion is as follows: accuracy, Acc:

$$\text{Acc} = \frac{\text{TA} + \text{TNA}}{\text{TA} + \text{FA} + \text{FNA} + \text{TNA}}. \tag{20}$$

Precision, P:

$$P = \frac{\text{TA}}{\text{TA} + \text{FA}}. \tag{21}$$

Recall, R:

$$R = \frac{\text{TA}}{\text{TA} + \text{FNA}}. \tag{22}$$

$F1$ value:

$$F1 = \frac{2 \times P \times R}{P + R}. \tag{23}$$

We assume that n categories need to be evaluated, a total of N samples, and exactly match the number of categories, so this article will calculate the mean of the above evaluation criteria as an evaluation criterion, as follows:

$$\text{Acc}_{\text{avg}} = \frac{1}{N} \sum (\text{Acc} * m_i). \tag{24}$$

Similarly,

$$\begin{aligned} P_{\text{avg}} &= \frac{1}{N} \sum (P_i * m_i), \\ R_{\text{avg}} &= \frac{1}{N} \sum (R_i * m_i), \\ F1_{\text{avg}} &= \frac{1}{N} \sum (F1_i * m_i). \end{aligned} \tag{25}$$

4.2. Model Training Process. When training the HCB model constructed in this paper, it is clear that some params settings affect the classification results, and in this section, the experiments in question eventually identify the params of some nodes. In this article, we need to use the relevant evaluation indicators to evaluate the performance of the model, and the following will build the model for some experiments and training. Among them, the training takes the form of setting parameters, as shown in Tables 1–3, to obtain parameters that contribute to the excellent performance of the model.

From Figures 6 and 7, it can be seen that the HCB model constructed in this paper is relatively stable, and final loss function tends to a value, indicating convergence, and the

Table 1: Convolutional module parameter setting table.

The parameter name	Convolutional kernel size	Number of convolutional kernels	*K* value of pooling
Value	3, 4, 5	128	3

Table 2: Cyclic module parameter setting table.

The parameter name	The number of nodes hidden by BiLSTM	Attention dimension
Value	256	512

Table 3: HCB model parameter settings.

The parameter name	The parameter value
Embed layer dropout values	0.2
Full-connection layer dropout value	0.2
Optimizer	AdaGrad
Number of trainings	50
Learning rate	0.01
Batch_size	64
The number of nodes in the fully connected tier	256

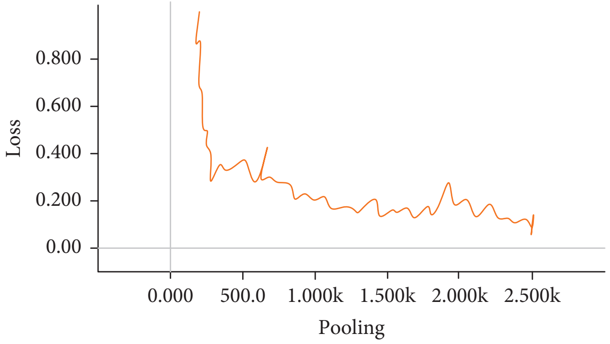

Figure 6: Loss change during training.

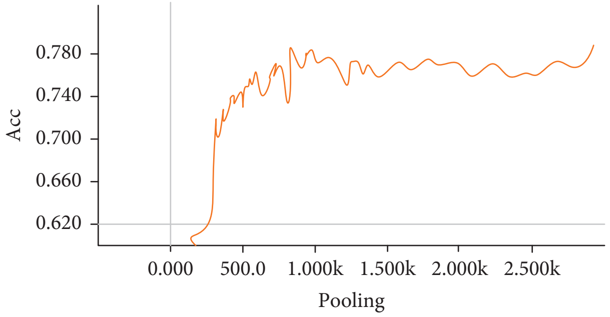

Figure 7: Acc change graph during training.

model can be used for classification. In addition, changes in Acc indicate that the model performs well.

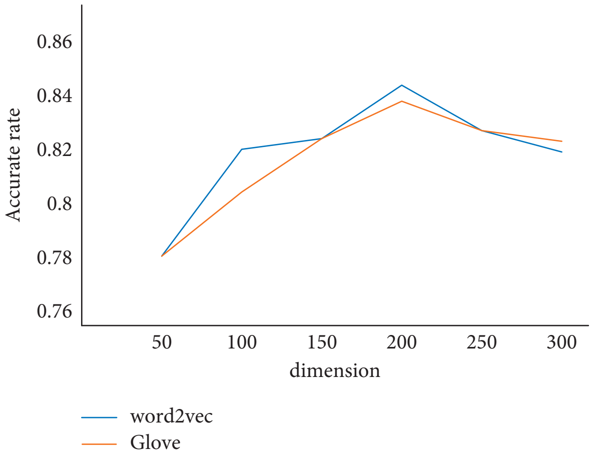

Figure 8: Word vector dimension comparison chart.

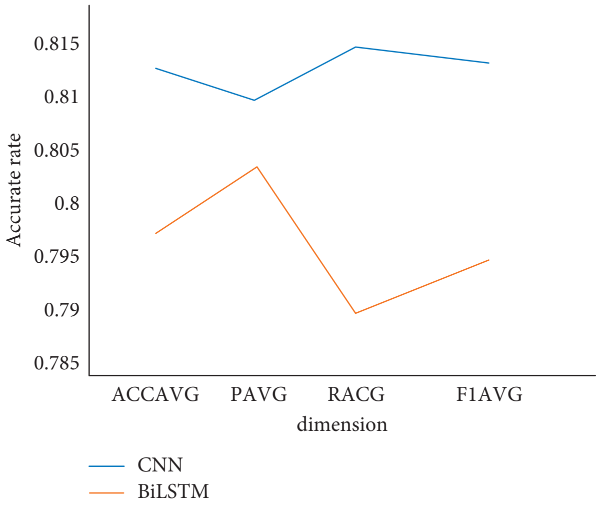

Figure 9: Comparison of CNN and BiLSTM models' training results.

In the previous settings of params, the selection of word vector models and the setting of word vector dimensions were ignored. These two cannot simply be subjective and need to be clarified by experimentation. Therefore, the following experiment is to obtain the orientation of two models for the built model HCB at different sizes. Use only Acc as an evaluation criterion to obtain the experimental results plot as shown in Figure 8.

As can be seen from this comparison result, for the HC model constructed in this paper, the word vector dimension obtained by the 200 word2vec and Glore models performed better, and the word vector obtained by the word2vec model was better than others. Thus, fix the word vector dimensions and model of the model in the following experimental section.

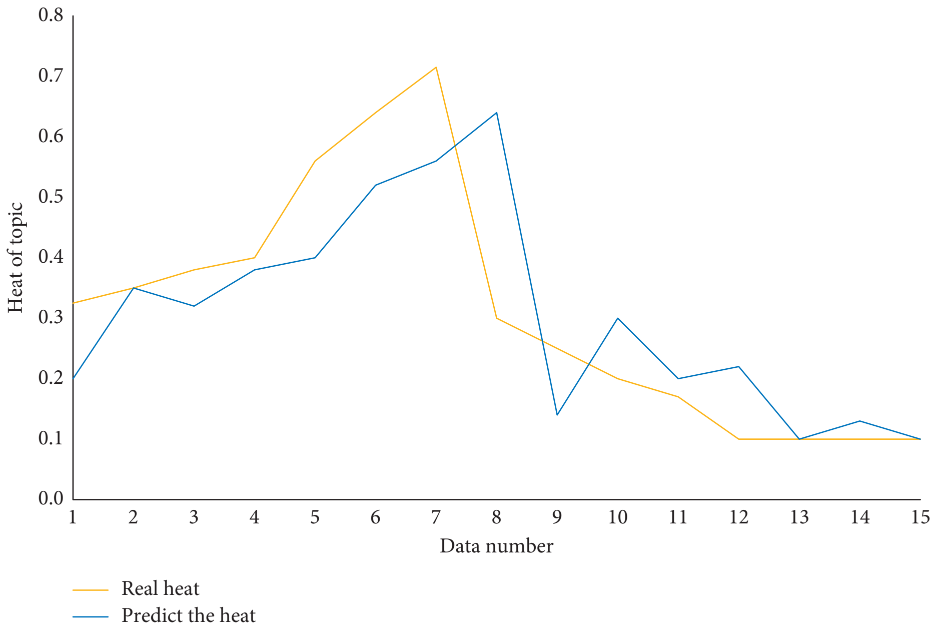

Figure 10: Comparison of the predicted value and the true value of the topic heat trend prediction model based on LSTM.

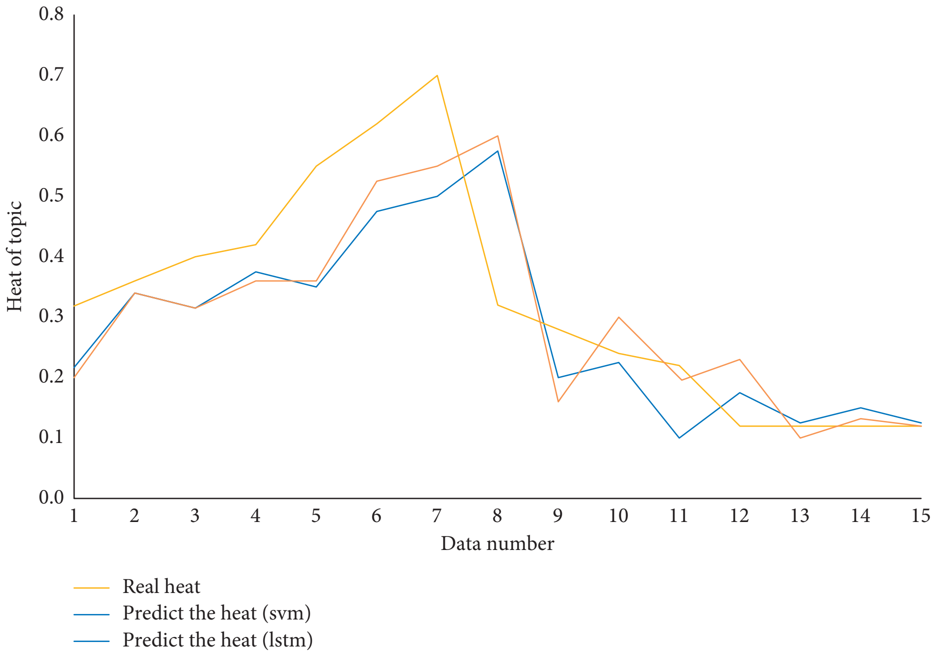

Figure 11: Comparison of prediction results.

4.3. Experimental Results. The CNN and BiLSTM models involved in the literature were selected to classify the first dataset mentioned in this paper, and the final results are shown in Figure 9.

According to the experimental comparison results, the results of two typical classification models are not very poor. Obviously, the classification effect of CNN is better than that of BiLSTM, reflecting the effectiveness of the HCB model.

In this paper, the comparison experiment is based on the real value. The experimental results shown in Figure 10 show that the LSTM model is highly accurate.

By using a prediction method based on support vector machine, this paper verifies whether the topic heat trend prediction model based on LSTM has a high degree of reduction over time, and the experimental results are shown in Figure 11.

From the above figure, it can be seen that the average error on the test set is 0.08342, and the topic heat trend prediction model based on LSTM is more in line with the real situation, reflecting the rationality and superiority of the proposed model.

5. Conclusion

Internet information exchange has become the mainstream of modern social networking, the real-time, accuracy, traceability, and reliability of Internet communication, so that the public relies on it to disseminate and obtain information. The Internet communication platform has gradually become a distribution center of social opinion, through the deep excavation of the topic, you can find the development law of the topic, convenient for users to track interest and personalized experience. In this paper, the embedded matrix is constructed as the input of the CNN-GRU model through data preprocessing, word frequency statistical analysis, and text vectorization, and then the loss function is improved to improve the classification effect of the model on the small category data to alleviate the data imbalance; finally, combined with the deep neural network model proposed in this paper, several sets of comparative experiments are designed to prove the improvement of classification performance of the model and the effectiveness of the optimized model.

At present, it is based on the real user data of Xinlang Weibo, which has strong authority. Follow-up research work will focus on the following points:

(1) Increase the extraction rate of seat features to improve the accuracy of hotspot recognition

(2) The classification of hot topics is complex, and the classification of hot topics and the prediction of hot development trends by category can improve the prediction efficiency

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares there are no conflicts of interest regarding this work.

References

[1] G. Hinton, L. Deng, D. Yu et al., “Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups,” *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[2] G. E. Dahl, D. Yu, and L. Deng, “Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition[J],” *IEEE Transactions on Audio Speech and Language Processing*, vol. 20, no. 1, pp. 30–42, 2011.

[3] S. Han, H. Mao, and W. J. Dally, “Deep compression: compressing deep neural networks with pruning, trained quantization and Huffman coding,” *Fibers*, vol. 56, no. 4, pp. 3–7, 2015.

[4] G. Montavon, W. Samek, and K.-R. Müller, “Methods for interpreting and understanding deep neural networks,” *Digital Signal Processing*, vol. 73, pp. 1–15, 2018.

[5] J. Ling, A. Kurzawski, and J. Templeton, “Reynolds averaged turbulence modelling using deep neural networks with embedded invariance,” *Journal of Fluid Mechanics*, vol. 807, pp. 155–166, 2016.

[6] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, “An experimental study on speech enhancement based on deep neural networks,” *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, 2014.

[7] D. S. Kim and J. W. Kim, “Public opinion sensing and trend analysis on social media: a study on nuclear power on twitter,” *International Journal of Multimedia and Ubiquitous Engineering*, vol. 9, no. 11, pp. 373–384, 2014.

[8] S. Li, X. Cai, C. Chen et al., “Analysis of microblog public opinion characteristics on traditional Chinese medicine against COVID-19 based on deep learning,” *History and Philosophy of Medicine*, vol. 3, no. 2, p. 9, 2021.

[9] H. He and S. Patnaik, “Research on prediction of internet public opinion based on grey system theory and fuzzy neural network,” *Journal of Intelligent and Fuzzy Systems*, vol. 35, no. 1, pp. 325–332, 2018.

[10] F. M. Yin, H. H. Xu, and H. H. Gao, “Research on Weibo public opinion prediction using improved genetic algorithm based BP neural networks,” *Computer Journal*, vol. 30, no. 3, pp. 82–101, 2019.

[11] C. Monterola, “Feasibility of a neural network as classifier of undecided respondents in a public opinion survey,” *International Journal of Public Opinion Research*, vol. 14, no. 2, pp. 222–229, 2002.

[12] L. Sun, J. Zhou, and W. Lin, “On network public opinion crisis early WarningBased on the BP neural network and genetic algorithm,” *Intelligence magazine*, vol. 33, no. 11, pp. 18–24, 2014.

[13] J. W. Xu, J. F. Zhao, W. C. Zhang, and X. X. Xu, “A novel soil moisture predicting method based on artificial neural network and Xinanjiang model,” *Advanced Materials Research*, vol. 121-122, pp. 1028–1032, 2010.

[14] D. Z. Wei, F. J. Chen, and X. X. Zheng, “Internet public opinion chaotic prediction based on chaos theory and the improved radial basis function in neural networks,” *Acta Physica Sinica*, vol. 64, no. 11, 2015.

[15] F. Atefeh and W. Khreich, “A survey of techniques for event detection in twitter,” *Computational Intelligence*, vol. 31, no. 1, pp. 132–164, 2015.

[16] J. Weng and B. S. Lee, “Event detection in twitter,” in *Proceedings of the International AAAI Conference on Web and Social Media*, pp. 401–408, Catalonia, Spain, July 2011.

[17] Y.-T. Chang, H. Yu, and H.-P. Lu, “Persuasive messages, popularity cohesion, and message diffusion in social media marketing,” *Journal of Business Research*, vol. 68, no. 4, pp. 777–782, 2015.

[18] M. Jiang, P. Cui, and N. J. Yuan, “Little is much:Bridging cross-platform behaviors through overlapped crowds,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 13–19, Arizona, USA, February 2016.

[19] G. Chen, Q. Kong, N. Xu, and W. Mao, “NPP: a neural popularity prediction model for social media content,” *Neurocomputing*, vol. 333, pp. 221–230, 2019.

[20] J. Huang, Y. Tang, Y. Hu, J. Li, and C. Hu, “Predicting the active period of popularity evolution: a case study on Twitter hashtags,” *Information Sciences*, vol. 512, pp. 315–326, 2020.

[21] P. Usharani, “Fast algorithms for mining association rules,” *International Journal of Scientific & Technology Research*, vol. 23, no. 3, pp. 21–30, 2013.

[22] T. Zhang, R. Ramakrishnan, and M. Livny, “Birch,” *ACM Sigmod Record*, vol. 25, no. 2, pp. 103–114, 1996.

[23] B. J. Frey and D. Dueck, “Clustering by passing messages between data points,” *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[24] S. Guha, R. Rastogi, and K. Shim, “Cure:an efficient clustering data clustering method for very large databases,” *ACM Sigmod Record*, vol. 25, no. 2, pp. 103–114, 1996.

[25] T. Rathod and M. Barot, “Trend analysis on twitter for predicting public opinion on ongoing events,” *International Journal of Computer Application*, vol. 180, no. 26, pp. 13–17, 2018.