

## Article

# Social Media Opinion Analysis Model Based on Fusion of Text and Structural Features

Jie Long<sup>1</sup>, Zihan Li<sup>2,3</sup>, Qi Xuan<sup>2,3</sup>, Chenbo Fu<sup>2,3</sup>, Songtao Peng<sup>2,3</sup> and Yong Min<sup>4,5,\*</sup> <sup>1</sup> College of Computer Science, Zhejiang University of Technology, Hangzhou 310023, China<sup>2</sup> Institute of Cyberspace Security, Zhejiang University of Technology, Hangzhou 310023, China<sup>3</sup> College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China<sup>4</sup> Center for Computational Communication Research, Beijing Normal University, Zhuhai 519087, China<sup>5</sup> School of Journalism and Communication, Beijing Normal University, Beijing 100875, China

\* Correspondence: myong@bnu.edu.cn

**Abstract:** The opinion recognition for comments in Internet media is a new task in text analysis. It takes comment statements as the research object, by learning the opinion tendency in the original text with annotation, and then performing opinion tendency recognition on the unannotated statements. However, due to the uncertainty of NLP (natural language processing) in short scenes and the complexity of Chinese text, existing methods have some limitations in accuracy and application scenarios. In this paper, we propose an opinion tendency recognition model HGAT (heterogeneous graph attention network) that integrates text vector and context structure methods to address the above problems. This method first trains a text vectorization model based on annotation text content, then constructs an isomorphic graph with annotation, news, and theme as its apex, and then optimizes the feature vectors of all nodes using an isomorphic graph neural network model with attention mechanism. In addition, this article collected 1,684,318 news items and 57,845,091 comments based on Toutiao, sifted through 511 of those stories and their corresponding 103,787 comments, and tested the impact of HGAT on this dataset. Experiments show that this method has stable improvement effect on different NLP methods, increasing accuracy by 2–10%, and provides a new perspective for opinion tendency recognition.



**Citation:** Long, J.; Li, Z.; Xuan, Q.; Fu, C.; Peng, S.; Min, Y. Social Media Opinion Analysis Model Based on Fusion of Text and Structural Features. *Appl. Sci.* **2023**, *13*, 7221. <https://doi.org/10.3390/app13127221>

Academic Editors: Rafal Rzepka, Pawel Dybala and Michal Ptaszynski

Received: 4 April 2023

Revised: 7 June 2023

Accepted: 14 June 2023

Published: 16 June 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** social network; natural language process; opinion tendency recognition; graph embedding; graph neural network

## 1. Introduction

With the rapid development of the Internet, the public can participate in and discuss a wide range of topics on the Internet according to their background, positions and viewpoints. The development of online media has gone through three generations, including a one-way e-newspaper model represented by a portal, a social network and media model represented by Weibo and WeChat, and an intelligent model based on personalized recommendation information represented by Toutiao and Douyin. In the current internet ecology, different types of internet media complement each other, leading to the diversification of motivations, paths and modes of interaction in internet information dissemination and topic discussion. People can express their views on major social media platforms at any time, leading to a flood of content with a personal bias. This phenomenon of open comment, on the one hand, increases the motivation and initiative of public to participate in the management of society, on the other hand, it raises problems of the proliferation of fake news, the polarization of perceptions and the intensification of public opinion conflicts [1,2].

The study of sentiment analysis has received extensive attention and research from interdisciplinary researchers, especially in the fields of fake news detection brought about by online media, social media bot/dong-army recognition, filter bubble identification and countermeasures [3,4]. The most direct way to reflect public's attitude to these real

news events is to analyze the content of their published comments. The comment text on social media represents the views of each individual on the current issue, and accurate identification of the sentiment orientation in these comments can help observe and judge the value trend of the public towards hot events. However, comments in the online media were mainly short stories, which made the study difficult. Compared with ordinary texts, short Chinese web texts tend to have the following characteristics: Comments tend to be short in length, and most people do not form complex sentences when commenting. There is a lot of noisy data, and short text posted on the internet often contains misspellings due to the lack of strict writing requirements. There are many “new words” with special meanings, and social media platforms often have original words with special meanings. Traditional text recognition methods struggle to understand the sentiment orientation these words represent, and Chinese participles are ambiguous and polysemous. Because of the above characteristics, traditional text recognition models cannot accurately recognize the sentiment orientation of network short text.

Research on recognizing sentiment orientation has accumulated to some extent, and most current methods rely on text content for identifying and judging sentiment orientation. These methods are based on natural language processing models to quantify Chinese words or characters [5,6], and then map the entire sentence or paragraph to a vector space followed by the recognition of sentiment orientation and judgement via classification methods. While NLP methods are able to identify sentiment orientation information in text to some degree, existing models are not effective in these problems due to the semantic diversity of Chinese texts and the emergence of new words on the Web. In recent years, the rise of graph convolutional neural networks and the significant advantages of graph embedding algorithms have been accompanied by: Ability to handle large-scale graph data. Graph embedding algorithms typically have efficient computational complexity and can handle large-scale graph data with millions or billions of nodes. Ability to capture similarity and association between nodes. Graph embedding algorithms can map adjacent nodes to similar low-dimensional vector spaces, thereby preserving the similarity and association between nodes. Ability to support various application scenarios. Graph embedding algorithms can be used in social network analysis, recommendation systems, bioinformatics, and other fields. Graph embedding algorithms have gradually been applied in the field of sentiment orientation classification [6,7]. This paper thus design a model based on word vectors and methods for embedding heterogeneous network graphs for comment response structures and textual representations in news stories. Using NLP methods, the model converts short text comments to word vectors and combines the feature vectors of neighbouring comments in the current comment domain (the neighboring comments include the first-order neighborhood and higher-order neighborhood range of neighboring nodes). Attention mechanisms can weight different parts of input, allowing the model to pay closer attention to important information and better understand the context of the input. Therefore, our model incorporates an attention mechanism for extracting important features and ultimately uses a classifier to classify the sentiment orientation of comments.

Given the paucity of datasets containing feedback-reply relationships, 511 public news items about Huawei and corresponding comments were obtained from Today’s Headline media platform as the dataset, covering a range of hot topics such as finance and technology. All comments in this article have been manually annotated for sentiment orientation and performed some exploratory research on sentiment analysis of user comments on this dataset, with the following key contributions:

1. Given the special nature of Chinese short texts, a method of extracting features from the comment structure to modify text vectors was adopted in order to achieve more accurate vector mappings. To improve the classification accuracy of sentiment orientation for Chinese short texts, we combined the dynamic comment representation vectors generated by the text vectorization model with features of the comment network structure.

2. In response to the heterogeneity of the comment structure network (the sentiment orientation expressed by comments with reply relationships is often opposite), in this paper, we proposed the HGAT model for efficient feature fusion, and we also incorporated an attention mechanism to aid in feature embedding. The proposed model can further improve text features, and even with a relatively simple classifier, it can obtain good prediction results for the sentiment orientation of the comments.
3. For the proposed HGAT model, a Toutiao dataset is proposed in this paper, and it is verified on the Toutiao dataset that the model in this paper performs better on the dataset of Toutiao compared to the text-only classification method and the graph embedding method based on isomorphism network.

The remainder of this paper is organized as follows. Section 2 is devoted to related work, including opinion tendency recognition, opinion conflict detection, and graph embedding algorithms. Section 3 details the proposed opinion tendency recognition method based on heterogeneous network graph embedding fused attention mechanism in this paper, where the related designs of word embedding layer, input layer, coding layer and output layer are explicitly given. Section 4 conducts experiments on the Toutiao dataset based on the method proposed in this paper, and shows the comparison experimental results between this method and NLP-based, structure-based methods and combined NLP and structure-based algorithms, through which it is verified that the model in this paper has better classification effect and better stability. Section 5 summarizes the various advantages of our proposed model and points out some directions for future work.

## 2. Related Work

In this section, the paper reviews the work related to this study, including opinion tendency recognition, opinion conflict detection, and graph embedding algorithms.

### 2.1. Opinion Tendency Recognition

Opinion tendency recognition is a subtask of sentiment classification, which is primarily concerned with the problem of text classification. The purpose of this task is to judge whether a statement about a target expresses a positive, negative, or neutral attitude toward the target. Early traditional methods relied primarily on dictionary models to perform feature counting on textual phrases, such as computing feature values via text decomposition, keyword extraction, and so on, and then determining the opinion-orientation of the text based on the feature values. Early research not only used dictionaries to count feature information in text, but also applied machine learning methods like Support Vector Machines (SVM) [8], decision trees [9,10] and other methods, which are primarily statistical methods for the analysis of opinion guidance. Most of these methods first compute sentence features based on text features and then combine all of the features into a vector. Lastly, they train classifiers such as SVMs and decision trees to achieve opinion-driven text classification.

The aforementioned methods, however, require manual design of feature computation methods and statistical analysis, which is time-consuming and labor-intensive. In contrast, neural network methods compensate for the shortcomings of the above methods. The success of neural networks in image and speech recognition has led to the gradual application of some related models to position detection. For example, in recent years, a variety of methods have been proposed that incorporate textual information into CNN and RNN models for sentiment analysis [11–14]. Apart from CNN and RNN models, LSTM has also been shown to have significant effects in opinion tendency recognition. Siddiqua et al. [15] proposed a nested BiLSTM and LSTM model structure in order to learn information from a larger set of contextual texts. Furthermore, they used an attention mechanism to magnify the impact of salient information content, which further improved the accuracy of the results. Mohtarami et al. [16] augmented the model structure of memristive neural networks using LSTM and CNNs, and introduced a similarity matrix for comparing content similarity across context, improving the prediction accuracy of the model.

With the introduction of the BERT model in recent years [17], a large number of Transformer based models have demonstrated significant performance on NLP tasks. Some recent work has also investigated the effectiveness of the BERT model for opinion tendency recognition. For example, Ghosh et al. [18] compared the performance of the BERT model with other methods on the SemEval2016 dataset and demonstrated that the BERT model has the best performance. Li et al. [19] also studied the effectiveness of the BERT model for data augmentation, and found that the BERT model performs exceptionally well in opinion tendency recognition.

For sentiment classification in the Chinese domain, scholars have optimized different aspects based on the mainstream methods. Chinese text opinion mining research started relatively late, and the sentiment lexicon and deactivation lexicon are not as rich as English. To solve this problem, Xu et al. [20] constructed an extended sentiment dictionary. The extended sentiment dictionary contains basic sentiment words, domain sentiment words and multisense sentiment words, which improves the accuracy of sentiment analysis. A plain Bayesian classifier is used to determine the text domain in which the multisense sentiment words are located. To address the problem that neural network models cannot accurately capture sentiment information in sentiment analysis tasks, Li et al. [21] proposed a sentiment information-based network model (SINM) that uses converter encoders and LSTM as model components to automatically find sentiment knowledge in Chinese texts with the help of Chinese sentiment dictionaries. Sheng et al. [22] to better solve the problem of sentiment analysis of long Chinese texts, proposed a bert-based fusion model and further used the attention mechanism to obtain the effective core sentiment of long Chinese texts.

Opinion tendency recognition by NLP is relatively intuitive and straightforward, but for social networks, NLP methods tend to ignore information related to the content of contextual comment texts, and there may be some correlation between the opinion orientation of contextual comments and the opinion orientation of current comment texts. There is a lack of effective methods in the field of Chinese opinion tendency detection to fuse different granularity information in models. Therefore, we consider integrating text information with other structural information to explore whether this method will improve the effectiveness of opinion tendency detection.

## 2.2. *Opinion Conflict Detection*

One aspect of opinion tendency recognition is the prediction of conflicts and controversies, and relevant research on detecting controversies on web pages and social media platforms has been going on for a long time. Recent work by Garimella et al. [23] explained a series of graph structure characteristics of feedback responses under a variety of topics. In addition, they proposed a feature-based graph structure algorithm for measuring the level of controversy on a topic. Research on micro-level conflict at the post or comment level, however, is still not well developed because there are many typos or special terms in the text of posts or comments. The focus of research in this area is on the use of linguistic features of comments (such as the number of appearances of statistically opinionated and topic-related phrases, and some Twitter-specific feature statistics) for controversy detection. Coletto et al. [24], for example, devised a method based on pattern feature extraction to extract features from a Twitter data set and to determine which of the responses in the data set are controversial. Coletto et al. primarily constructed graphs based on responses between comments and relationships between the users' friends in the data set, and then extracted a set of structural features using the motifs algorithm. Lastly, we applied a classifier to classify the features obtained. Zhong et al. [25] also designed TPC-GCN and DTPC-GCN based on the GCN model in order to distinguish whether the post content is controversial or not. Both the The TPC-GCN and DTPC-GCN methods efficiently incorporate structural information from the heterogeneous network and introduce attention mechanisms, achieving higher recognition accuracies than the NLP methods.

### 2.3. Graph Embedding

Graph embedding methods are now a common approach to network analysis and research. Graph embedding methods map nodes, edges or the entire network of graphs into a low dimensional vector space for representation, and then use machine learning, deep learning, and other methods for downstream tasks such as classification of nodes, prediction of links, and classification of graphs.

DeepWalk [26] is considered to be the first work in this area in graph embedding problems. DeepWalk collects a series of sequences of nodes from the graph via random walks and then vectorizes each representation of the nodes using the Skip-Gram [27] model. The node2vec [28] method expands the search space of DeepWalk's sampling strategy by combining breadth-first search and depth-first search in order to obtain more global and local structural features in a better way, which is referred to as biased random walks, improving the representational learning capability of the network. The development of this typical research field has been accelerated by the emergence of deep learning methods. Variants of Graph Neural Networks (GNN) include GCN (Graph Convolution Network) [29], which provides a simplified method for computing graph embeddings. Based on the structure of the CNN model, the GCN model captures the graph structure and corresponding feature information of each node through graph convolution calculation, and converts the structural information between nodes into vector representations.

Finally, the emergence of attention mechanisms provides a novel method for making accurate predictions based on the weighted combination of all encoded input feature vectors. Li et al. [30] proposed an alternative method to utilize attention mechanisms in dynamic heterogeneous networks. The system employs three types of attention: structural, semantic and temporal, and obtains a better performance.

All of the aforementioned graph embedding methods based on message passing and attention mechanisms are based on the assumption of network isomorphism, i.e., nodes in the network are primarily connected with nodes belonging to the same attributes or classes. Real world networks, however, do not always satisfy the isomorphism hypothesis. To address highly heterogeneous networks, Zhu et al. [31] proposed the H<sub>2</sub>GCN model, which aggregates neighborhood node characteristics and uses node degree values for normalization to update node characteristics, thus improving the efficiency of graph embedding algorithms in heterogeneous networks. Fu et al. [32] proposed a method to improve the performance of cross domain classification tasks using network embedding similarity metrics.

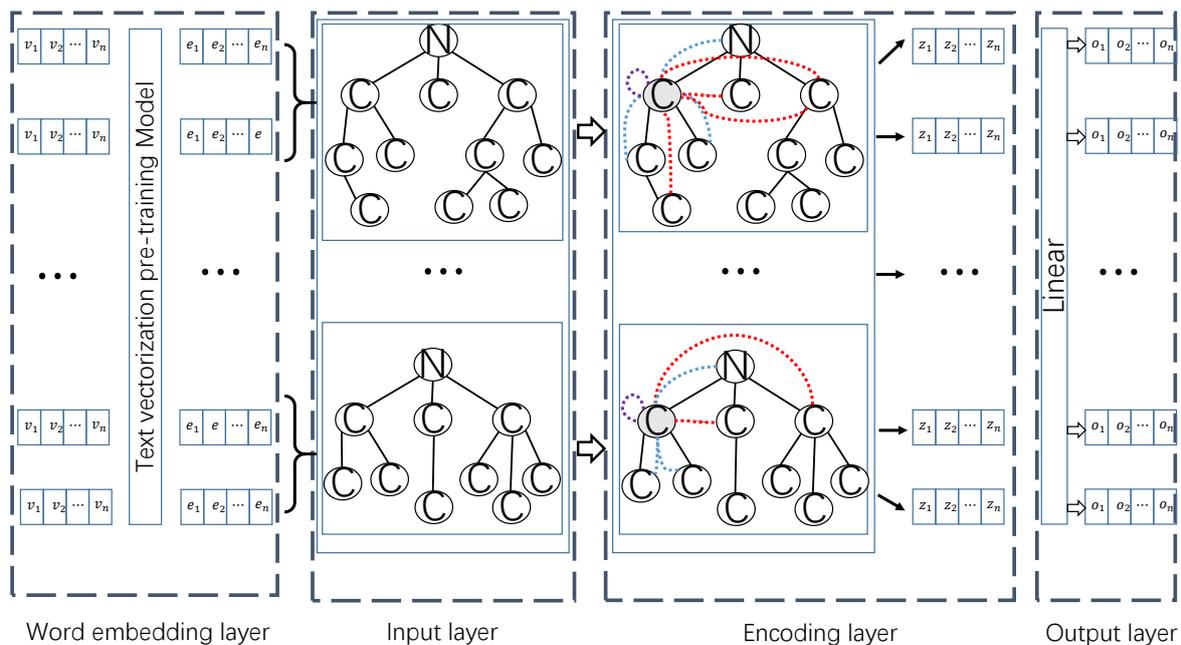
With respect to the application of graph embedding algorithms in sentiment analysis of opinions, Zhang et al. [7] used a dependency tree to build simple syntactic dependency relations and used a graph convolutional neural network to fuse the syntactic information, learn text-to-word vector representations, and derive the final vector representation weighted by the importance of the context content. The algorithm introduces graph embedding methods to sentiment classification, but there are few methods that combine feedback response relationships with feedback text for exploration and search.

To summarize, feature-based algorithms of text or network structure have achieved some effects in sentiment classification problems. For text vectorization methods, on the other hand, they ignore the structural information of the feedback response context in social media, although pure network methods fall short for information mining of textual information. Motivated by the above situation, this paper proposes a model that integrates textual information with structural response information from comments for the opinion tendency recognition task.

### 3. Proposed HGAT Model

This article mainly performs the task of identifying the opinion tendency of comments on social media. The sentence that needs to be predicted can be visually described as a triplet  $T = \langle S, P, C \rangle$  composed of text content, network attributes, and emotional polarity,  $S = w_1, w_2, w_3, \dots, w_n$  represents a sentence consisting of  $n$  words  $w_i$ , where  $w_i, 0 < i < n$ ,

$P = T_1, T_2, T_3, \dots, T_n$  represents  $n$  triplet entities that have a direct reply relationship with the current sentence.  $C = -1, 0, 1$  represents the opinion polarity of the sentence, where  $-1$  indicates that the comment statement has a negative opinion on the topic of the current news,  $0$  indicates a neutral opinion, and  $1$  indicates a positive opinion. The main task of this article is to predict the opinion tendency of a specified comment based on the text information content of the comment and the reply relationships between comments. The overall architecture of the HGAT model consists of four layers, as shown in Figure 1.



**Figure 1.** Logical structure of HGAT model.  $N$  is the news node, and  $C$  is the comment node. The red and blue lines represent the process of attention coefficient calculation between the target node and each node in its different neighborhood, and feature fusion is performed based on the attention coefficient. The red line represents the second-order domain, while the blue line represents the first-order domain. The purple line indicates that the encoding layer concatenates the input features of nodes into the output vector.

1. **Word Embedding Layer:** The text of the comment is tokenized and fed into a pre-trained text vectorization model using the corresponding phrase table. In this case, The input text is the text from the training set news, and the initial vector representations of all comment phrases and news content are obtained based on the output of the model after secondary training;
2. **Inputting Layer:** The feature vectors of the text obtained from the previous layer are combined with the social media commentary network, and text feature vectors are used as input to node features;
3. **Encoding Layer:** Training is carried out on the structure of each news story. The attention mechanism is used to merge the features of each neighbor node of the comment node at various levels with the node’s own features, and the feature vector of the comment node is changed accordingly;
4. **Output Layer:** Given the output vector of the previous layer, the polarity category of the predicted opinion is obtained via the softmax function.

### 3.1. Word Embedding Layer

The HGAT model first divides and filters the raw data at the word embedding layer, and the division tool used in this paper is the tokenizer package under transformer, and the deactivation table uses the data from the deactivation table published by the Harbin Insti-

tute of Technology [33]. First of all, based on the stopwords in the stopword list, words that are not highly relevant to opinion orientation information or expression of position information in the corpus are filtered out for subsequent opinion orientation analysis. In terms of representation, the tokenize method splits the text into tokens by word. Assuming a statement  $S = w_1, w_2, w_3, \dots, w_n$  represents a statement composed of  $n$  words  $w_i$ , each word is converted into its corresponding numerical value through a tokenizer-predefined dictionary to form the corresponding vector of the sentence:  $V = v_1, v_2, v_3, \dots, v_n$ , where  $v_i, i \in [1, n]$  is the numerical value after each word is converted. Then, after the training set is processed and numbered, all the numbered comment texts of the training set are input into the pre-trained text vectorization model, and the dynamic word vector  $E$  is obtained through calculation:  $E = [e_1, e_2, e_3, \dots, e_n]^T$ . Where  $e$  represents the vectorized output of each comment utterance after being trained by the text vectorization model. This paper added a fully connected layer to the original text vector model after the output of the model to reduce the dimension of the output vector to 128 dimensions, i.e.,  $n = 128$ . Furthermore, based on the trained model described above, this paper also outputs a corresponding 128 dimensional vector for the textual content of all news stories, representing news content.

### 3.2. Input Layer

After obtaining the comment text vectors, the HGAT model fuses the text vector features with the network structure features in the input layer and passes them to the subsequent coding layer. In the case of the news commentary network constituted by the dataset in this paper: it consists primarily of two types of nodes: news nodes and comment nodes. News nodes represent the specific content of news releases, and comment nodes are the content of comments posted by users. The composition presented in this paper is based on the response relationship between comments, and the results are presented in Figure 2. Since the news nodes and the comment nodes in this paper's method are both represented by textual feature vectors, and the dimensions of both of these vector types are 128, the network containing the two types of nodes is treated as a homogeneous network when the network is input to the input layer. In this paper, the network is constructed based on the comment response structure: where  $N$  represents a news node,  $C_1, C_2, C_3$  represent three independent first level comments under  $N$  news stories, this means that all three of these comments are directly commented on under the new. Similarly,  $C_{(1-1)}, C_{(1-2)}$  represent two comments replying to  $C_1$ , and  $C_{(3-1)}, C_{(3-2)}$  represent two comments replying to  $C_3$ . This paper uses the above method of graph construction to convert the set of response relations in 511 news content into network structures for the purpose of representation:  $G = g_1, g_2, \dots, g_n$ , where  $g_i, i \in [1, n]$  represents the network consisting of all comments below a news item and the news node itself.

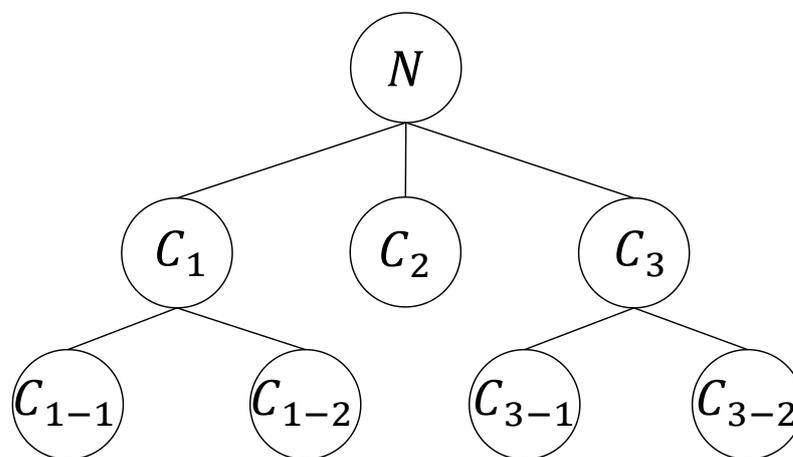


Figure 2. News-Comment Network Architecture.

### 3.3. Encoding Layer

At this layer, a graph embedding method based on heterogeneous networks is used to extract and fuse contextual feature information from the feedback. In the dataset of this paper, the polarity of sentiment expressed in contextual feedback is often opposite, i.e., responses to feedback under the “Huawei” topic news often have conflicting relationships with one another. Unfortunately, most graph embedding algorithms, such as GCN and GAT, often merge neighboring node features based on the assumption of node homogeneity, which is contrary to the purpose of the present paper. Social media opinion prediction requires an algorithm that can perform graph embeddings on strongly heterogeneous graphs and can aggregate features from neighboring nodes with strong heterogeneity in order to update the node’s own feature vector.

The H<sub>2</sub>GCN model is a graph embedding vector representation method for heterogeneous networks, which re-designs the feature fusion strategy for highly heterogeneous networks. The H<sub>2</sub>GCN model first aggregates the features of the neighboring nodes of the target node and combines the calculation results with the target node’s own feature vector to update the target node’s feature. In this paper, the HGAT model is modified based on the H<sub>2</sub>GCN model, and the node representation vector obtained by the input layer is multiplied by an initial trainable weight matrix  $W_{emb} \in \mathbb{R}^{(128 \times \rho)}$  as the initial vector feature  $r_v^0$  of the node.

$$r_v^0 = \text{relu}(F_{fc}(X) * W_{emb}) \tag{1}$$

*relu* represents the Rectified Linear Unit activation function,  $F_{fc}$  represents the fully connected function,  $\rho$  represents the specified output vector dimension of the hidden layer in HGAT, and  $X$  represents the matrix of input vector representations in the encoding layer. In addition, in this paper, an attention mechanism is introduced to aggregate the characteristics of the neighboring nodes for the representation vector  $r_v^{(k)}$  of the node  $v$ . The attention mechanism calculates the attention coefficients of the neighboring nodes of the target node and updates the feature vector of the target node based on the feature vectors of the neighboring nodes. A key point of the attention mechanism is to determine the influence of the surrounding nodes on the attention coefficients of the current node. To calculate the attention coefficients, the correlation coefficient between the target node and its neighboring nodes needs to be calculated. The input feature matrix  $h = \{\vec{h}_0, \vec{h}_1, \vec{h}_2, \dots, \vec{h}_n\}$ ,  $\vec{h}_i \in \mathbb{R}^F$ , then the calculation formula of the correlation coefficient  $e_{i,j}$  between node  $i$  and node  $j$  is as follows:

$$e_{i,j} = \text{LeakyReLU}(\vec{a}(W\vec{h}_i || W\vec{h}_j)) \tag{2}$$

in the formula,  $W \in \mathbb{R}^{(F' \times F)}$  represents the weight matrix, where  $F'$  represents the dimension of the specified output features.  $W$  is applied to each node to ensure that each node can perform a self-attention operation to obtain an attention coefficient. The  $||$  symbol represents the defined attention operation function. The vector  $\vec{a}$  in the formula represents a feedforward neural network with a dimension of  $\vec{a} \in \mathbb{R}^{(2 \times F')}$ , and the result calculated by the neural network is non-linearized by the *LeakyReLU* function to obtain the correlation coefficient  $e_{i,j}$  between node  $i$  and node  $j$ .

Based on Equation (3), the attention coefficient between node  $i$  and node  $j$  can be obtained by normalization, denoted as  $\alpha_{(i,j)}$ , which is given by:

$$\alpha_{i,j} = \frac{\exp(e_{i,j})}{\sum_{k \in N_i} \exp(e_{i,k})} \tag{3}$$

where  $N_i$  represents all neighboring nodes of the target node  $i$ . After obtaining the node attention coefficient, the updated feature vector of node  $i$  can be obtained by summing up the weighted features of all its neighboring nodes.

$$h_i = \sigma\left(\sum_{j \in N_i} \alpha_{i,j} W \vec{h}_j\right) \tag{4}$$

where,  $\sigma$  represents a non-linear activation function. Since this paper needs to aggregate the features of the second-order neighboring nodes of node  $i$ , during the process of updating the original network node features using the attention mechanism, feature updates need to be performed through different adjacency matrices. Then, the feature vectors updated by the second-order neighborhood are concatenated to obtain the  $n$ th-order vector feature.

$$r_v^n = [r1, r2] \tag{5}$$

$$r1 = att(r_v^{n-1}, A - E), r2 = att(r_v^{n-1}, A * A - A - E) \tag{6}$$

where  $att$  denotes the process of obtaining the node update vector based on the attention mechanism. After stacking  $n$  layers through the above process, the vector  $r_v^n$  of the output vertex  $v$  will be used as the node vector input of the next output layer. In this paper,  $n$  is set to 2 in the experiments.

The output vector after the feature vector update can be represented as:

$$r_v^{(final)} = [r_v^{(1)}, r_v^{(2)}, \dots, r_v^{(K)}, F_{fc}(X)] \tag{7}$$

$r_v^{(final)}$  is the output vector of the embedding layer, which is the node embedding vector output by the graph embedding model. The benefit of doing this is to separate the target node's own features from the features of its neighboring nodes, and calculate them separately, emphasizing the heterogeneity between the node and its surrounding nodes. Therefore, when dealing with heterogeneous networks, H<sub>2</sub>GCN often has better performance than graph embedding algorithms based on the homogeneity assumption, such as GCN.

### 3.4. Output Layer

This layer mainly obtains the node embedding vector representation of the previous layer's encoding layer, smoothly calculates the result through the *softmax* function, and predicts the sentiment polarity of the review text, as shown in the equation:

$$Y = softmax(r_v^{(final)} W_c) \tag{8}$$

where  $W_c \in \mathbb{R}^{((2^{K+1}-1)*\rho \times \lceil Y \rceil)}$  is a weight matrix,  $\lceil Y \rceil$  is the number of sentiment polarities that need to be classified, and  $\lceil Y \rceil$  is set to 3 in this paper. The model in this paper is trained by minimizing the cross-entropy loss value between the predicted value and the true value, as shown below:

$$l_n = -\omega_{Y_n} X_{n,Y_n} \tag{9}$$

$$l(X, Y) = \begin{cases} \sum_{n=1}^N \frac{1}{\sum_{n=1}^N \omega_{Y_n}} l_n, & \text{if reduction} = \text{'mean'} \\ \sum_{n=1}^N l_n, & \text{if reduction} = \text{'sum'} \end{cases} \tag{10}$$

where  $N$  represents the batch data  $D(X, Y)$  containing  $N$  samples, where  $X$  is the output of the neural network and has been normalized and logarithmically processed, and  $Y$  is the category label corresponding to the sample.  $l_n$  is the loss corresponding to the  $n$ th sample, which can be obtained from Formula (9). The constant  $\omega$  is used to deal with the problem of sample imbalance among multiple categories. Formula (10) represents the loss result calculated for the batch data containing  $N$  samples, where *reduction* = 'mean'

and *reduction = 'sum'* represent two ways of calculating the loss by taking the mean and directly accumulating, respectively.

#### 4. Experimental Section

##### 4.1. Datasets and Evaluation Metric

Given the paucity of Chinese datasets with comment-reply structures and sentiment labels at the comment level, data for this article were obtained from the Toutiao media platform. The site includes news articles, reviews, and information on the corresponding users of reviews. The news articles focus on multiple areas such as technology, finance and entertainment, with a particular focus on Huawei-related news published between March and December 2019. Multiple comments are included in each news article, and these comments reflect users' attitudes toward the content of the news article. For this reason, this paper categorizes the sentiment of comments into positive, neutral, and negative. An example of a conflict occurring in a news article is shown in Figure 3. In this example, the news article *N* belongs to the topic *T*, and multiple comments in the news article express different views of the users. Figure 3 shows that comments are labelled as positive, neutral or negative depending on the user's attitude towards the information. There is a case where the comment  $C_{3-1}$  expresses a negative sentiment, but in fact supports the content of the news article *N*. This is because in the structure of the comment tree,  $C_{3-1}$  refutes the viewpoint of  $C_3$  towards news article *N*, while  $C_3$  expresses an opposing opinion towards news article *N*. Therefore, we consider comments like  $C_{3-1}$  to have a positive sentiment towards news article *N*. In data preprocessing, we manually confirmed each comment's label based on its content and contextual feedback. If one of the two comments expresses a positive attitude toward the news while the other one expresses a negative attitude, we consider this to be a conflict of interest. Edges between comments with no conflict indicate that both comments have the same sentiment. Furthermore, this paper does not consider edges between neutral comments and other types of comments in the news. As shown in Table 1, we collected a total of 511 news articles and 103,787 reviews for the entire dataset. In some special cases, there are comments without content or publication time. We conjecture that these comments have been removed by users. In the case of such comments, their labels can only be determined on the basis of the labels of their child comments.



**Figure 3.** News and related Comments.

Depending on the needs of the task, the focus of this paper is on the conflicting content in hot news among all news stories. For this purpose, we extracted three data subsets for experimentation. In particular, we first found the two most active users who made the most comments under different news stories, denoted by  $u_1$  and  $u_2$ , respectively. To simplify the notation, we represent the actuality and corresponding comments commented by  $u_1$  as

Toutiao#1, and the other subset consisting of all the news items commented by  $u_2$  such as Toutiao#2. Toutiao#1 is connected to each news item in Toutiao#2 in this paper to form a larger news commentary dataset. The text-feature vector of the Topic node is the vectorised representation of the word “Huawei”. Table 1 shows the statistics for the two subsets and the entire dataset.

**Table 1.** Toutiao and three subsets Dataset Statics.

	Toutiao	Toutiao#1	Toutiao#2
Number of news	511	11	11
Number of users	71,579	3496	5940
Number of comments	103,787	5570	10,580
Positive comments	54,994	2224	3622
Neutral comments	23,236	1647	4095
Negative comments	25,557	1699	2863

The evaluation criteria used in this article are *accuracy*, *macro-precision*, *macro-recall*, and *macro-F1*. The calculation formulas are shown below:

$$accuracy = \frac{TP + FN}{TP + TN + FP + FN} \quad (11)$$

$$recall_i = \frac{TP_i}{TP_i + FN_i} \quad (12)$$

$$precision_i = \frac{TP_i}{TP_i + FP_i} \quad (13)$$

$$F1_i = 2 \frac{recall_i precision_i}{recall_i + precision_i} \quad (14)$$

$$macro-recall = \frac{\sum_{i=1}^N recall_i}{N} \quad (15)$$

$$macro-precision = \frac{\sum_{i=1}^N precision_i}{N} \quad (16)$$

$$macro-F1 = \frac{\sum_{i=1}^N F1_i}{N} \quad (17)$$

where,  $TP_i$  represents the number of samples that were predicted as positive and actually are positive for the  $i$ -th opinion polarity category,  $FP_i$  represents the number of samples that were predicted as positive but actually are negative for the  $i$ -th opinion polarity category,  $FN_i$  represents the number of samples that were predicted as negative but actually are positive for the  $i$ -th opinion polarity category, and  $TN_i$  represents the number of samples that were predicted as negative and actually are negative for the  $i$ -th opinion polarity category.  $N$  represents the total number of opinion polarity categories.

#### 4.2. Baseline Method

A comparison of our proposed model with other text vectorization methods including RoBERTa, Ernie, BERT, CPT, GPT2, and so on. The description of these models is as follows.

RoBERTa [34]: RoBERTa is an enhanced version of BERT, achieving better performance by improving training tasks as well as data generation methods, by training for a longer period of time, using larger batch sizes, and by using more data.

Ernie [35]: ERNIE is a large scale knowledge enhanced model published by Baidu, covering NLP and cross-modal models. ERNIE performed further optimisations based on the BERT model and performed better on Chinese NLP tasks. The main improvement of this technique is the masking mechanism.

BERT [17]: BERT is a pre-trained language representation model that emphasizes not using traditional unidirectional language models or shallowly concatenating two unidirectional language models for pre-training, but by using a novel masked language model (MLM) to generate a bidirectional deep language representation.

CPT [36]: The CPT model primarily makes changes to the structure of the encoding and decoding portions of the Transformer and adds three portions: the shared encoder (S-Enc), the decoder comprehension (U-Dec), and the decoder generation (G-Dec). It improves language comprehension and generation skills by using multi-task articulatory training.

GPT2 [37]: The GPT2 model is a new pre-trained model published by the OpenAI organisation on the basis of the GPT model in 2018. Unlike the BERT model, GPT2 is built using the transformer decoder module, while BERT is built using the transformer encoder module. The auto-regression mechanism allows GPT2 to better capture information about the context content.

GCN [29]: GCN was first proposed by Bruna et al. Each node in the network structure updates its own node state by exchanging information with one another, and uses convolutional computations to extract spatial features to learn the node representation, thus performing classification tasks.

#### 4.3. Model Parameter Setting

The experiments in this article were conducted using the PyTorch framework and the A-100 GPU for training.

The number of epoch iterations was set to 50, and the best model was saved based on the accuracy score.

#### 4.4. Comparison of Experimental Results

In addition to the aforementioned text vectorization models, several comparative models were added for ablation experiments in this article: the HGAT + ones method represents the use of an identity matrix instead of node feature vectors as input to the HGAT model in this article. We use this method to test the effect of pure network methods on the true opinion classification after ablation of the feature vectors of the text vectors. The GCN + BERT method represents the use of the BERT model to extract comment node text vector representations and updating comment node feature vectors through the GCN model. We use this method to test the effectiveness of the heterogeneity graph embedding model in HGAT for feature extraction of network structure. In this paper, we build on the above models by comparing several text classification methods, network structure-based methods, and two methods that combine text features with structural features.

The same set of training and test samples was used during implementation so that the experimental results from different algorithms could be compared. Likewise, the training and test sets used for training with the text methods and updating the node feature vectors using the GCN and HGAT models were the same set of nodes. To balance the training and test samples, this paper divided the three comment types equally, and the split results are presented in Table 2. In this paper, the ratio of training set to test set is 90/10, we also did the experiment under 80/20 ratio, which has little effect on the result, here we choose the result under 90/10 ratio.

**Table 2.** The division result of datasets.

Data Set	Toutiao#1			Toutiao#2		
	Positive	Neutral	Negative	Positive	Neutral	Negative
Training set	904	899	899	1689	1689	1692
Test set	97	102	102	188	191	185

Table 3 shows the experimental results based on the training and test sets above, where Acc. is the precision, Pre. represents the macro-precision, Rec. represents the macro-recall, and F1 represents the macro-F1. The experimental results show that:

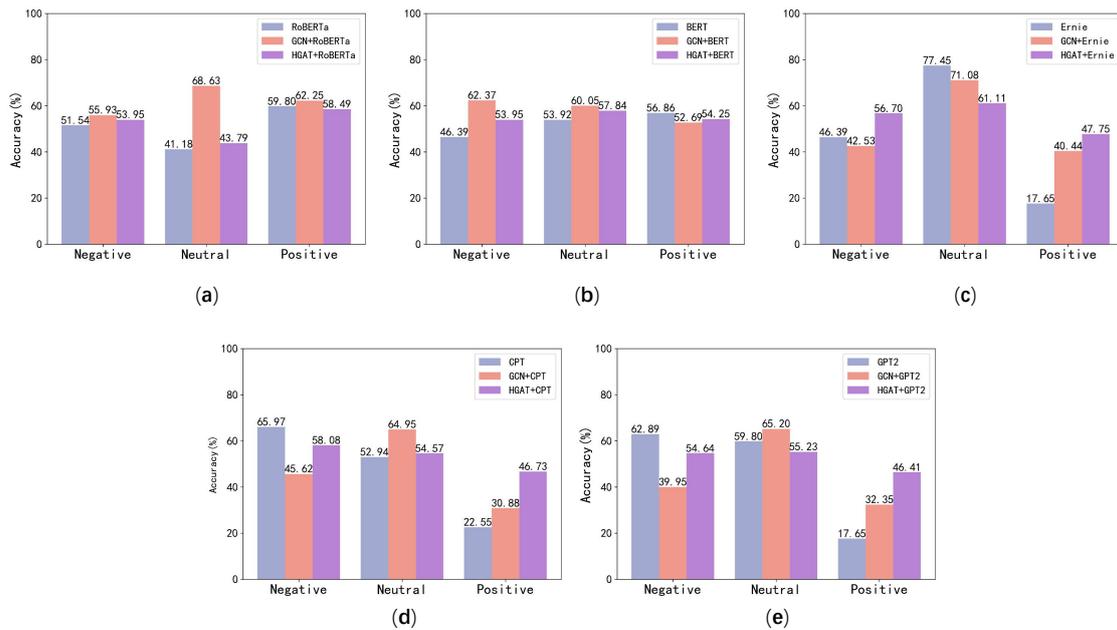
1. On both the Toutiao#1 and Toutiao#2 datasets, the text and structural feature combination method proposed in this paper performs better than the original text vectorisation algorithm, which states that the algorithm proposed in this paper, which optimizes and updates the text vectors based on the structural characteristics of the network, has some effect.
2. The HGAT + BERT model proposed in this paper outperforms all text vectorization methods as well as all structure-based algorithms on both datasets.
3. On the Toutiao#1 dataset, the HGAT proposed in this paper has a slightly weaker effect than the GCN, which could be because there are more neutral comments in the structure of the network. This paper's model uses graph embeddings of heterogeneous networks to update node vectors, as shown in Figures 4 and 5, but responses to neutral feedback are often still neutral feedback, making prediction of neutral feedback vectors slightly worse for HGAT than for GCN.

In order to test whether the proposed HGAT model is universally applicable to a variety of text vectorization algorithms, in this paper, five text vectorization algorithms have been combined and compared with two graph embedding algorithms, GCN and HGAT, and the experimental results can be seen in Table 4. As can be seen in Table 4, the algorithm that updates the node based on the word embedding vectors Ernie, CPT, and GPT2 performs better than GCN, although it is weaker than GCN in terms of updating RoBERTa word embedding vectors but still has better performance than the original RoBERTa algorithm. Furthermore, while the RoBERTa text vectorization method does not perform as well as BERT in the pure text vectorization representation, the improvement is largest after updating the nodes by the network structure method. In Toutiao#2 dataset, the value of ACC. was increased from the initial value of 54.96% to 61.82% by HGAT algorithm, and to 68.66% by the GCN algorithm. Figures 4 and 5 show that while the RoBERTa algorithm has the lowest accuracy for predicting neutral feedback, it performs best overall in predicting both negative and positive feedback. As a result, the overall result obtained after updating the node feature vector by the HGAT and GCN graph neural network models is better than that obtained by the BERT, Ernie, CPT and GPT2 algorithms. Furthermore, when compared to all text vectorization algorithms, the algorithm combined with HGAT and text vectorization algorithms has superior performance, who show that the heterogeneous network structure-based method for graph embeddings can efficiently optimize the original text vector and obtain better results with better overall performance. The main reason for the improvement, in our view, is that the HGAT model can effectively capture the correlation between the structural features of sentences with different views and their context, which makes the identification of the opinion tendency more accurate.

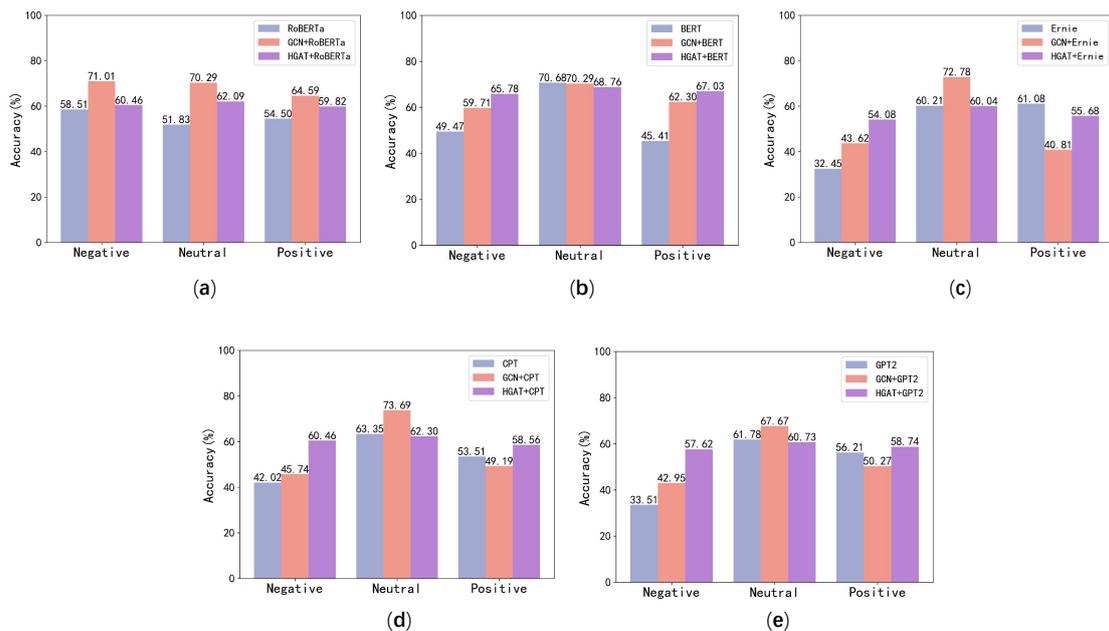
In order to investigate the accuracy of the text vectorization methods, as well as the combination of structure and text vectorization methods for the identification of three types of comment nodes, in this paper, we have compared five text vectorization models and models that combine textual and structural features. Figures 4 and 5 show the results. It can be seen from the results that.

1. For the most part, the model that combines HGAT and text vectorization has better accuracy in identifying both positive and negative reviews than the model that combines GCN and text vectorization, and also has a larger improvement over the text vectorization methods.
2. The combination model of GCN and text vectorisation has an advantage in terms of identifying neutral comments. We believe this is due to the fact that the context of neutral comments often involves neutral comments in response comments, thus, the node homogeneity-based GCN method has an advantage for identifying neutral comments.
3. The GCN and text vectorization combination models are relatively unstable, and their accuracy is greatly impacted by the text vectorization. Of these, in the Toutiao#1

dataset, based on the CPT vectorisation method, the GCN + CPT model even had an accuracy of around 20% lower in the identification of negative comments when compared to the original method. The proposed HGAT model in this paper has better stability than the GCN model. The reason for the improvement, in our view, is that the HGAT model introduces attention mechanisms, which may better quantify the influence of neighbouring nodes on the current one. Furthermore, the network heterogeneity-based model is best suited to environments with extreme opinion polarization in social networks.



**Figure 4.** Comparison of the recognition results of comments with different opinions by different methods on the Toutiao#1 dataset. (a) Based on the RoBERTa model; (b) Based on the BERT model; (c) Based on the Ernie model; (d) Based on the CPT model; (e) Based on the GPT2 model.



**Figure 5.** Comparison of the recognition results of comments with different opinions by different methods on the Toutiao#2 dataset. (a) Based on the RoBERTa model; (b) Based on the BERT model; (c) Based on the Ernie model; (d) Based on the CPT model; (e) Based on the GPT2 model.

**Table 3.** Comparison of evaluation for different models.

Model		Toutiao#1				Toutiao#2			
		Acc. (%)	Pre. (%)	Rec. (%)	F1 (%)	Acc. (%)	Pre. (%)	Rec. (%)	F1 (%)
NLP-based	Roberta	50.83 ± 0.06	51.15 ± 0.60	50.84 ± 0.07	50.55 ± 0.05	54.96 ± 0.51	54.99 ± 0.44	54.98 ± 0.50	54.94 ± 0.69
	Bert	52.49 ± 0.20	52.77 ± 2.07	52.39 ± 0.22	52.38 ± 0.99	55.32 ± 0.70	55.27 ± 1.13	55.18 ± 0.71	54.72 ± 0.81
	Ernie	47.17 ± 0.66	48.70 ± 0.33	47.16 ± 0.60	43.83 ± 0.82	51.24 ± 0.81	52.15 ± 1.07	51.25 ± 0.82	50.34 ± 1.28
	CPT	46.51 ± 0.66	46.73 ± 1.15	46.78 ± 0.62	43.80 ± 0.45	50.53 ± 0.20	50.38 ± 1.63	50.50 ± 0.35	49.70 ± 1.60
	GPT2	46.84 ± 0.60	47.77 ± 0.49	47.16 ± 0.65	44.95 ± 0.79	53.01 ± 1.78	53.08 ± 1.86	52.96 ± 1.14	52.63 ± 0.94
Structure-based	HGAT + ones	44.43 ± 3.58	48.03 ± 7.56	44.68 ± 3.65	42.07 ± 5.16	48.23 ± 1.93	48.36 ± 2.00	48.28 ± 1.93	48.16 ± 1.94
	GCN + ones	53.15 ± 1.16	55.18 ± 1.48	53.32 ± 1.21	52.61 ± 1.28	66.49 ± 1.00	67.34 ± 0.32	66.46 ± 0.91	66.32 ± 0.91
Combined	GCN + Bert	<b>58.31 ± 1.17</b> <sup>1</sup>	<b>59.39 ± 1.48</b> <sup>1</sup>	<b>58.37 ± 1.39</b> <sup>1</sup>	<b>58.38 ± 1.28</b> <sup>1</sup>	64.14 ± 0.84	64.82 ± 0.32	64.09 ± 0.90	64.00 ± 0.91
	HGAT + Bert	55.82 ± 1.48	56.04 ± 1.45	55.79 ± 1.46	55.84 ± 1.46	<b>67.20 ± 0.82</b> <sup>1</sup>	<b>67.38 ± 0.88</b> <sup>1</sup>	<b>67.19 ± 0.82</b> <sup>1</sup>	<b>67.22 ± 0.83</b> <sup>1</sup>

<sup>1</sup> The bold numbers represent the best experimental results.**Table 4.** Experimental results comparison of different NLP methods and structural combination.

Model	Toutiao#1				Toutiao#2			
	Acc. (%)	Pre. (%)	Rec. (%)	F1 (%)	Acc. (%)	Pre. (%)	Rec. (%)	F1 (%)
Bert	52.49 ± 0.20	52.77 ± 2.07	52.39 ± 0.22	52.38 ± 0.99	55.32 ± 0.70	55.27 ± 1.13	55.18 ± 0.71	54.72 ± 0.81
GCN + BERT	<b>58.31 ± 1.32</b> <sup>1</sup>	<b>59.39 ± 1.12</b> <sup>1</sup>	<b>58.37 ± 1.36</b> <sup>1</sup>	58.38 ± 1.34 <sup>1</sup>	64.14 ± 0.64	64.82 ± 0.36	64.09 ± 0.68	64.00 ± 0.73
HGAT + BERT	55.82 ± 1.48	56.04 ± 1.45	55.79 ± 1.46	55.84 ± 1.46	<b>67.20 ± 0.82</b> <sup>1</sup>	<b>67.38 ± 0.88</b> <sup>1</sup>	<b>67.19 ± 0.82</b> <sup>1</sup>	<b>67.22 ± 0.83</b> <sup>1</sup>
Roberta	50.83 ± 0.06	51.15 ± 0.60	50.84 ± 0.07	50.55 ± 0.05	54.96 ± 0.51	54.99 ± 0.44	54.98 ± 0.50	54.94 ± 0.69
GCN + Roberta	<b>62.37 ± 3.10</b> <sup>1</sup>	<b>64.67 ± 2.04</b> <sup>1</sup>	<b>62.27 ± 3.07</b> <sup>1</sup>	<b>62.44 ± 3.23</b> <sup>1</sup>	<b>68.66 ± 0.30</b> <sup>1</sup>	<b>69.23 ± 0.48</b> <sup>1</sup>	<b>68.63 ± 0.36</b> <sup>1</sup>	<b>68.58 ± 0.48</b> <sup>1</sup>
HGAT + Roberta	52.41 ± 1.17	52.65 ± 1.24	52.44 ± 1.19	52.24 ± 1.13	61.82 ± 0.48	61.89 ± 0.56	61.79 ± 0.49	61.79 ± 0.52
Ernie	47.17 ± 0.66	48.70 ± 0.33	47.16 ± 0.60	43.83 ± 0.82	51.24 ± 0.81	52.15 ± 1.07	51.25 ± 0.82	50.34 ± 1.28
GCN + Ernie	51.49 ± 0.70	52.44 ± 0.93	51.35 ± 0.74	50.54 ± 0.97	52.57 ± 0.54	52.84 ± 0.52	52.41 ± 0.51	51.49 ± 0.51
HGAT + Ernie	<b>54.41 ± 3.33</b> <sup>1</sup>	<b>54.34 ± 3.36</b> <sup>1</sup>	<b>54.43 ± 3.35</b> <sup>1</sup>	<b>54.23 ± 3.39</b> <sup>1</sup>	<b>56.62 ± 0.47</b> <sup>1</sup>	<b>56.60 ± 0.46</b> <sup>1</sup>	<b>56.60 ± 0.46</b> <sup>1</sup>	<b>56.59 ± 0.46</b> <sup>1</sup>
CPT	46.51 ± 0.66	46.73 ± 1.15	46.78 ± 0.62	43.80 ± 0.45	50.53 ± 0.20	50.38 ± 1.63	50.50 ± 0.35	49.70 ± 1.60
GCN + CPT	47.17 ± 2.68	52.84 ± 2.92	47.15 ± 2.81	43.56 ± 4.09	56.34 ± 0.57	56.91 ± 0.55	56.21 ± 0.61	55.39 ± 0.45
HGAT + CPT	<b>52.99 ± 1.20</b> <sup>1</sup>	<b>53.01 ± 1.05</b> <sup>1</sup>	<b>53.06 ± 1.18</b> <sup>1</sup>	<b>52.95 ± 1.22</b> <sup>1</sup>	<b>60.46 ± 0.95</b> <sup>1</sup>	<b>60.64 ± 0.92</b> <sup>1</sup>	<b>60.44 ± 0.94</b> <sup>1</sup>	<b>60.47 ± 0.92</b> <sup>1</sup>
GPT2	46.84 ± 0.60	47.77 ± 0.49	47.16 ± 0.65	44.95 ± 0.79	53.01 ± 1.78	53.08 ± 1.86	52.96 ± 1.14	52.63 ± 0.94
GCN + GPT2	45.93 ± 1.23	46.39 ± 1.05	45.83 ± 1.25	44.53 ± 1.47	53.72 ± 0.43	53.66 ± 0.46	53.63 ± 0.42	53.16 ± 0.28
HGAT + GPT2	<b>51.24 ± 2.21</b> <sup>1</sup>	<b>51.28 ± 2.20</b> <sup>1</sup>	<b>51.30 ± 2.19</b> <sup>1</sup>	<b>51.16 ± 2.25</b> <sup>1</sup>	<b>59.04 ± 0.78</b> <sup>1</sup>	<b>59.09 ± 0.80</b> <sup>1</sup>	<b>59.03 ± 0.79</b> <sup>1</sup>	<b>59.05 ± 0.79</b> <sup>1</sup>

<sup>1</sup> The bold numbers represent the best experimental results.

## 5. Conclusions

The HGAT model proposed in this paper, which combines semantic and structural features, first extracts the semantic feature vector from the text of the comment by means of NLP, obtaining the text feature vector of the comment. Based on the graph embedding method of heterogeneous networks, the text vector is then further modified according to the context response relation of the comment, effectively ameliorating the problem caused by the short length of the Chinese comments and expressing the comment vector accurately. The attention mechanism is simultaneously combined with the graph embedding method of heterogeneous networks to assign attention probabilities to the salient features during learning of the input features, improving the final classification accuracy. Overall, the experimental results demonstrate that the proposed model has better performance on the Toutiao News dataset when compared to both pure text classification and graph embedding methods based on homogenous networks. Future analysis will focus on a more in-depth exploration of the textual representation and properties of the comment-news network, by combining deeper semantic recognition models with heterogeneous network graph embedding models to improve the effectiveness of opinion trend classification. In summary, the proposed method is easy to transfer, and the trained model can be used extensively in practical social media platforms.

**Author Contributions:** Conceptualization, J.L., Z.L. and Y.M.; methodology, J.L. and Z.L.; data curation and formal analysis J.L. and Z.L.; writing—original draft preparation, J.L. and Z.L.; writing—review and editing, Q.X., C.F. and S.P.; funding acquisition, Q.X. and Y.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was partially supported by the National Natural Science Foundation of China under Grant 61973273 and by the Zhejiang Provincial Natural Science Foundation of China under Grant LGF21G010003.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Informed consent was obtained from all subjects involved in the study.

**Data Availability Statement:** Data will be made available upon reasonable request to the corresponding author.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Conover, M.; Ratkiewicz, J.; Francisco, M.; Goncalves, B.; Menczer, F. Political Polarization on Twitter. In Proceedings of the International AAAI Conference on Web and Social Media, Barcelona, Spain, 17–21 July 2011.
2. Linden, S.; Roozenbeek, J.; Compton, J. Inoculating against fake news about COVID-19. *Front. Psychol.* **2020**, *11*, 566790. [[CrossRef](#)] [[PubMed](#)]
3. Sharma, K.; Feng, Q.; He, J.; Ruchansky, N.; Liu, Y. Combating Fake News: A Survey on Identification and Mitigation Techniques. *ACM Trans. Intell. Syst. Technol.* **2019**, *10*, 1–42. [[CrossRef](#)]
4. Vicario, M.; Vivaldo, G.; Bessi, A.; Zollo, F.; Quattrociocchi, W. Echo Chambers: Emotional Contagion and Group Polarization on Facebook. *Sci. Rep.* **2016**, *6*, 37825. [[CrossRef](#)] [[PubMed](#)]
5. Zhang, D.; Xu, H.; Su, Z.; Xu, Y. Chinese comments sentiment classification based on word2vec and SVMperf. *Expert Syst. Appl.* **2015**, *42*, 1857–1863. [[CrossRef](#)]
6. Qiao, Q.; Bo, T.; Huang, M.; Yang, L.; Zhu, X. Learning Tag Embeddings and Tag-specific Composition Functions in Recursive Neural Network. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 26–31 July 2015; Volume 1: Long Papers.
7. Zhang, Y.; Qi, P.; Manning, C.D. Graph Convolution over Pruned Dependency Trees Improves Relation Extraction. *arXiv* **2018**, arXiv:1809.10185.
8. Bouchlaghem, R.; Elkhelifi, A.; Faiz, R. SVM based approach for opinion classification in Arabic written tweets. In Proceedings of the 2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA), Marrakech, Morocco, 17–20 November 2015.
9. Es-Sabery, F.; Es-Sabery, K.; Qadir, J.; Sainz-De-Abajo, B.; Torre-Diez, I. A MapReduce Opinion Mining for COVID-19-Related Tweets Classification Using Enhanced ID3 Decision Tree Classifier. *IEEE Access* **2021**, *9*, 58706–58739. [[CrossRef](#)]

10. Sanjay, K.S.; Danti, A. Detection of fake opinions on online products using Decision Tree and Information Gain. In Proceedings of the 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 27–29 March 2019.
11. Wan, W.; Xiao, Z.; Liu, X.; Wei, C.; Wang, T. pkudblab at SemEval-2016 Task 6: A Specific Convolutional Neural Network System for Effective Stance Detection. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), San Diego, CA, USA, 16–17 June 2016.
12. Du, J.; Xu, R.; He, Y.; Lin, G. Stance Classification with Target-specific Neural Attention. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, 19–25 August 2017.
13. Zhou, Y.; Cristea, A.I.; Shi, L. Connecting Targets to Tweets: Semantic Attention-Based Model for Target-Specific Stance Detection. In *Lecture Notes in Computer Science, Proceedings of the Web Information Systems Engineering—WISE 2017—18th International Conference, Puschino, Russia, October 7–11, 2017, Proceedings, Part I*; Bouguettaya, A., Gao, Y., Klimenko, A., Chen, L., Zhang, X., Dzerzhinskiy, F., Jia, W., Klimenko, S.V., Li, Q., Eds.; Springer: Berlin/Heidelberg, Germany, 2017; Volume 10569, pp. 18–32.
14. Augenstein, I.; Rocktäschel, T.; Vlachos, A.; Bontcheva, K. Stance Detection with Bidirectional Conditional Encoding. *arXiv* **2016**, arXiv:1606.05464.
15. Siddiqua, U.A.; Chy, A.N.; Aono, M. Tweet Stance Detection Using an Attention based Neural Ensemble Model. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019), Minneapolis, MO, USA, 2–7 June 2019; Long and Short Papers; Association for Computational Linguistics: Minneapolis, MO, USA, 2019; Volume 1.
16. Mohtarami, M.; Baly, R.; Glass, J.R.; Nakov, P.; Moschitti, A. Automatic Stance Detection Using End-to-End Memory Networks. *arXiv* **2018**, arXiv:1804.07581.
17. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
18. Ghosh, S.; Singhanian, P.; Singh, S.; Rudra, K.; Ghosh, S. Stance Detection in Web and Social Media: A Comparative Study. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: Proceedings of the 10th International Conference of the CLEF Association, CLEF 2019, Lugano, Switzerland, 9–12 September 2019, Proceedings 10*; Springer International Publishing: Berlin/Heidelberg, Germany, 2020.
19. Li, Y.; Caragea, C. Target-Aware Data Augmentation for Stance Detection. In Proceedings of the Association for Computational Linguistics 2021, Virtual, 1–6 August 2021; pp. 1850–1860.
20. Xu, G.; Yu, Z.; Yao, H.; Li, F.; Meng, Y.; Wu, X. Chinese Text Sentiment Analysis Based on Extended Sentiment Dictionary. *IEEE Access* **2019**, *7*, 43749–43762. [[CrossRef](#)]
21. Li, G.; Zheng, Q.; Zhang, L.; Guo, S.; Niu, L. Sentiment Information based Model For Chinese text Sentiment Analysis. In Proceedings of the 2020 IEEE 3rd International Conference on Automation, Electronics and Electrical Engineering (AUTEEE), Shenyang, China, 20–22 November 2020; pp. 366–371. [[CrossRef](#)]
22. Sheng, D.; Yuan, J. An Efficient Long Chinese Text Sentiment Analysis Method Using BERT-Based Models with BiGRU. In Proceedings of the 2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Dalian, China, 5–7 May 2021; pp. 192–197. [[CrossRef](#)]
23. Garimella, K.; Morales, G.D.; Gionis, A.; Mathioudakis, M. Quantifying Controversy on Social Media. *ACM Trans. Soc. Comput.* **2018**, *1*, 1–27. [[CrossRef](#)]
24. Coletto, M.; Garimella, K.; Gionis, A.; Lucchese, C. Automatic controversy detection in social media: A content-independent motif-based approach. *Online Soc. Netw. Media* **2017**, *3–4*, 22–31. [[CrossRef](#)]
25. Zhong, L.; Cao, J.; Sheng, Q.; Guo, J.; Wang, Z. Integrating Semantic and Structural Information with Graph Convolutional Network for Controversy Detection. *arXiv* **2020**, arXiv:2005.07886.
26. Perozzi, B.; Al-Rfou, R.; Skiena, S. DeepWalk: Online Learning of Social Representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014.
27. Mikolov, T.; Sutskever, I.; Kai, C.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2013.
28. Grover, A.; Leskovec, J. node2vec: Scalable Feature Learning for Networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 13–17 August 2016.
29. Kipf, T.N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv* **2016**, arXiv:1609.02907.
30. Li, Q.; Shang, Y.; Qiao, X.; Dai, W. Heterogeneous Dynamic Graph Attention Network. In Proceedings of the 2020 IEEE International Conference on Knowledge Graph (ICKG), Nanjing, China, 9–11 August 2020.
31. Zhu, J.; Yan, Y.; Zhao, L.; Koutra, D. Beyond Homophily in Graph Neural Networks: Current Limitations and Effective Designs. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 7793–7804.
32. Fu, C.; Zheng, Y.; Liu, Y.; Xuan, Q.; Chen, G. NES-TL: Network Embedding Similarity-Based Transfer Learning. *IEEE Trans. Netw. Sci. Eng.* **2020**, *7*, 1607–1618. [[CrossRef](#)]
33. Datatang | StopWordsSet. Available online: <http://www.datatang.com/data/19300> (accessed on 18 March 2021).
34. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
35. Sun, Y.; Wang, S.; Li, Y.; Feng, S.; Wu, H. ERNIE: Enhanced Representation through Knowledge Integration. *arXiv* **2019**, arXiv:1904.09223.

36. Shao, Y.; Geng, Z.; Liu, Y.; Dai, J.; Yang, F.; Zhe, L.; Bao, H.; Qiu, X. CPT: A Pre-Trained Unbalanced Transformer for Both Chinese Language Understanding and Generation. *arXiv* **2021**, arXiv:2109.05729.
37. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.