Rumors Detection Based on Lifelong Machine Learning

Xiaoxia He, Gulanbaier Tuerhong, Mairidan Wushouer, Xin Dai

1 School of Information Science and Engineering, Xinjiang University, Urumqi 830017, China

ABSTRACT The amount of training data in the field of Weibo rumor detection is small, and online news changes constantly, but the existing rumor detection models do not have the ability of continuous learning, they also cannot achieve knowledge accumulation and update, they usually need a large number of training examples to improve the effect. In contrast, Lifelong Machine Learning (LML) paradigm has the ability of continuous learning, which retains the knowledge learned in the past and uses it to help future learning tasks, after learning, some knowledge is updated. With the growth and update of knowledge, the performance of each task will be better and better. Hence, we use this paradigm to build a Weibo rumor detection model. We firstly extracted three types of features based on content, user, and propagation from Weibo events, and proposed three new propagation features and Bidirectional Encoder Representations from Transformers (BERT) semantic features of source message for rumor detection. We then used Simulated Annealing (SA) to improve the Genetic Algorithm (GA), which was called GA-SA used to search for the best global minimum feature subset to improve the classification effect of the Efficient Lifelong Learning Algorithm (ELLA) on rumors. In the continuous learning process, the ELLA transfers knowledge to learn new tasks and refines knowledge over time to maximize performance across all tasks. The proposed model is called GA-SA-ELLA. The experimental results show that our model even in the case of less training data for each task could achieve superior detection results.

INDEX TERMS Lifelong Machine Learning, continuous learning, Weibo rumor, the best minimum feature.
information from source news and related posts, and then modeled the heterogeneous graph among source post and retweets to capture the global structural information. Wang et al. [7] considered the dynamic evolution of diffusion structures, and proposed Dynamic Propagation Structures (NM-DPS), which divided the structure into several segments according to posting time of each tweet and encode each segment into a vector, then the vector as the input was fed into Bi-directional Gate Recurrent Unit (BiGRU) with attention network to learn structure representation, the structure representation fusioned text representation for Twitter and Weibo rumor detection. Some researchers used a combination of manual features and neural networks to detect rumors, for example, Lv et al. [8] used CNN-LSTM model. Meanwhile, comment sentiment as an important feature was added to the model for Weibo rumor detection. In recent years, with the success of the Graph Neural Network (GNN), rumor detection algorithms based on GNN had become popular. Bian et al. [9] explored both characteristics by operating on both top-down and bottom-up propagation of rumors, and proposed Bi-Directional Graph Convolutional Networks (Bi-GCN) to identify Weibo and Twitter rumors. Lin et al. [10] proposed VAE-GCN model, the model is the first study of employing GCN as encoder to learn propagation information, then used AutoEncoder (AE) to learn the overall structure information, the detector utilized the output of AE to classify Twitter and Weibo news events as fake or not. Wang et al. [11] proposed Knowledge-driven Multimodal Graph Convolutional Network (KMGCN) model which jointly the textual information, knowledge concepts, and visual information for Twitter and Weibo fake news detection. The knowledge concepts were retrieved relevant knowledge from Knowledge Graphs (KG), the visual information was obtained with object detection techniques, and the text information was obtained with GCN. BAI et al. [12] firstly built a Source-Replies relationship graph (SR-graph) for each conversation, using nodes to represent a tweet, and edges to represent interactive tweets between them. Based on SR-graphs, an integrated graph convolutional neural network with node proportional allocation mechanism (EGCN) was proposed for Twitter rumor detection. Choi et al. [13] captured the dynamic diffusion information of the rumors using GCN and constructed a Dynamic Graph Convolutional Neural Network (DGCN) for Twitter and Weibo rumor detection. To improve the effectiveness of rumor detection, some researchers used other related information to help rumor detection. For example, Zeng et al. [14] proposed a Hybrid Stance Attention Mechanism (RDM-HSAM) model which used Weibo stance information to improve the classification effect of Weibo rumor detection tasks. The rumor detection method based on deep learning can effectively avoid the problems caused by manual design of features in the rumor detection method based on traditional machine learning. However, the rumor detection method based on deep learning has a large demand for training data. When there are few training instances, the trained classifier will have the problem of classification bias [15].

Not only do research on English or Chinese rumors, now many researchers are beginning to focus on other languages. For example, Zoleikha et al. [16] proposed the BERT-SAEnviar model for early verification of Persian rumors on Twitter and Telegram, the model extracted the contextual word embeddings (CWE), speech act features, and artificially designed writing style (WS) features, next fused these three views features for identifying Persian rumors. Ahmed et al. [17] used a set of features like Term Frequency-Inverse Document Frequency (TF-IDF) extracted from tweet contents and trained a set of machine learning classifiers to verify fake news surrounding COVID-19 in Arabic tweets. However, without the ability to accumulate knowledge, these models need a large volume of training examples to learn effectively, but there are fewer training examples in the field of Weibo rumor detection, and even if more training data is manually collected, it is time-consuming and labor-intensive. To make matters worse, the online languages on Weibo or other social media change constantly, the manual collection process is continuous. In contrast, Lifelong Machine Learning (LML) keeps the knowledge gained after learning learned tasks and uses it to help new tasks, after learning, some knowledge will be updated to refine it. This use of historical knowledge to help new task learning is called forward knowledge transfer. As more and more knowledge is retained and refined over time, the model is more effective in learning new tasks [18]. Thus, the new rumor detection task doesn’t need a large volume of training examples to improve the effect. At the same time, LML can cope with the constant change of Weibo language through its continuous learning ability.

The Efficient Lifelong Learning Algorithm (ELLA) algorithm has an LML mechanism [19]. Compared with other LML algorithms (such as CL-cbsSVM[20]), the ELLA algorithm maintains a shared knowledge among all tasks, that is it can not only perform forward knowledge transfer to learn new tasks but also perform backward knowledge transfer is that it uses knowledge to continuously improve the performance of learned tasks. In addition, the algorithm is to improve the learning effect of the task by repeatedly refining knowledge, rather than knowledge growth. Therefore, we construct a novel rumor detection model based on the ELLA algorithm to continuously improve the learning effect of all rumor detection tasks. However, we extracted many features for rumor detection, the original ELLA algorithm could not avoid overfitting of high-dimensional feature vectors in the training process and further improved effect by optimizing these features, so we must improve the algorithm. Commonly used feature selection algorithms are Genetic Algorithm (GA) and Simulated Annealing (SA). GA imitates the process of natural selection, which means that those species that can adapt to environmental changes become the optimal solution.
GA has a strong global search capability, and the optimization results have nothing to do with the initial conditions. SA algorithm adopts an iterative movement according to the variable temperature parameter which imitates the annealing transaction of the metals. The algorithm accepts inferior individuals with a certain probability in the iterative process, this increases the diversity of the population. Therefore, the algorithm can avoid falling into the local minimum in the early iterations. But they have their own disadvantages: (1) The GA is easy to fall into the local optimum because each crossover always preserves the higher fitness individuals of the father [21]. (2) The convergence speed of SA is slow, and the search capability is related to the initial temperature and other parameters [22]. Given the complementary advantages of GA and SA, and to improve the convergence speed and the global optimization ability, we use SA to improve GA. When the progeny of GA reproduction encounters inferior solutions, they are accepted according to a certain probability given by SA. This method avoids falling into the local optimum by increasing the diversity of the population and improving the convergence speed compared with SA. Finally, we obtain the best minimum feature subset of the global optimum. We name this method GA-SA. Based on the obtained best minimum feature subset, the ELLA algorithm is used to continuously learn Weibo rumors to gradually improve the effect of detection. The proposed model is called GA-SA-ELLA. In experiments, we first verify that the new proposed features and BERT semantic feature are effective for rumor detection, and then, the effects of different feature and the improved effect by feature optimization based on GA-SA algorithm, last but not least, the influence of knowledge refinement on task detection effect, then we show that our proposed GA-SA-ELLA model is comparable or even better than current state-of-art models.

Our main contributions can be summarized as follows:
1) Three new propagation features were proposed and semantic features are used for rumor detection.
2) Modified the calculation method of sentiment tendency in literature [23] and proved its validity.
3) Built a “Question and Correction” vocabulary corpus for extracting features.
4) Used SA to improve GA, which improves the convergence speed and feature optimization effect, compared to GA and SA.
5) To our best knowledge, this is the first study of employing LML in rumor detection of social media and proposed a novel continuous learning model called GA-SA-ELLA.

Our GA-SA-ELLA method outperforms several state-of-the-art approaches, and with knowledge refining, GA-SA-ELLA could achieve higher effectiveness.

II. LIFELONG MACHINE LEARNING

A. COMPONENTS OF LML AND RELATED WORK

LML (or lifelong learning) is an advanced machine learning paradigm that learns continuously, accumulates the knowledge learned in previous tasks, and uses it to help future learning. Its learning process is shown in Figure 1. It has learned \( n-1 \) tasks: \( \{T_i\}_{i=1}^{n-1} \). These tasks are called learned tasks or old tasks, and each task has its own corresponding data set: \( \{D_i\}_{i=1}^{n-1} \). These tasks can be in the same field or different fields. When the \( i \)-th task \( T_i \) appears (called the current task or the new task), and the data set at this time is \( D_i \), LML will transfer the historical knowledge in the KB to help the Knowledge-Based Learner (KBL) learn a new task \( T_i \). The role of KB is to store and update historical knowledge, and KBL is to learn current tasks and generate knowledge.

From the above description, LML contains three key features: (1) continuous learning; (2) retaining the knowledge learned in the past in KB; (3) using historical knowledge to help future task learning. Therefore, LML can continuously learn a series of tasks in the same or similar fields, extract and store useful knowledge in KB, and transfer this knowledge to help future tasks. Knowledge will be updated after learning new tasks to refine it. With the knowledge stored and refined, the better the learning effect of the model.

From the learning process of LML in Figure 1, we can also know LML is similar to MTL. The MTL has been used to achieve rumor detection. Ma et al. proposed an MTL framework that trained two highly pertinent tasks, i.e., rumor detection and stance classification, the framework trained both tasks jointly using weight sharing to extract the common and task-invariant features while each task can still learn its task-specific features [24]. Both MTL and LML can use shared information to help task learning, this can solve the problem of lack of training data in the field of rumor detection by learning other tasks. However, MTL is difficult to continue learning, and it is difficult to accumulate knowledge. Although MTL can optimize all tasks by retraining when new tasks arrive, when the number of tasks is large enough, retraining will consume a lot of time and resources, while LML can continue to learn multiple new tasks with the help of historical knowledge [18].

LML has been used in many fields, including image identification, NLP, and network security. For example, Zhou et al.[25] built a lifelong learning model for objection detection, which overcomes the catastrophic forgetting...
problem in other research methods, because the LML model contains component KB. Rajvardhan Oak et al.[26] built a lifelong anomaly detection model, aiming to improve the performance of the model through knowledge growth. Wang et al.[27] constructed a lifelong relationship extraction model, aiming to discover new data and relationships. Hong et al.[28] proposed an approach based on the LML paradigm for sentiment classification as an example to show how it adapts to different domains and keeps efficiency meanwhile. From their experimental results, LML can achieve very good results in both the image processing field and the NLP tasks. In this paper, LML is applied for Weibo rumor detection. Compared with the existing rumor detection model, it achieves a higher rumor recognition accuracy and F1 value. It can be seen that LML not only has a wide range of applications but also exhibits powerful capabilities.

We apply LML for Weibo rumor detection, firstly, Task Manager assigns a new rumor task to the KBL. Then, KBL generates a classification model for the current task with the help of past knowledge stored in KB, the KB retains the new knowledge for use in future tasks. With the knowledge accumulation or refinement, models become more and more intelligent, the effectiveness of subsequent Weibo rumor detection tasks would be improved, so the new rumor detection task does not require a large number of training samples to improve the effect. This will deal with the current problem of training examples in the field of rumor detection. At the same time, through the continuous learning ability of LML to cope with the ever-changing Weibo online language.

B. ELLA

The ELLA algorithm with the LML mechanism has a global knowledge transfer structure, that is the ELLA algorithm maintains a shared knowledge among all tasks, and the knowledge in KB is transferred and learned in all tasks, thereby improving the learning effect of all tasks. Therefore, ELLA could guarantee higher performance, compared to LML algorithms that can only perform forward knowledge transfer. In addition, this process is computationally efficient due to the use of optimization strategies, for example, the second-order Taylor expansion optimization strategy and only updating the most influential parameters [19]. The detailed learning process of the algorithm is shown as follows:

The ELLA algorithm uses a parameter model for learning. Given T tasks, the prediction function on the current task \( t \) is shown in Equation (1):

\[
f^{(t)}(x) = f(x; \theta^{(t)}) = x^T \theta^{(t)}
\]

Where \( \theta^{(t)} \) is the parameter vector of \( t \)-th task, which is a linear combination of weight vector \( s^{(t)} \) and the sparse shared matrix \( L \), and the relationship between them is shown in Equation (2):

\[
\theta^{(t)} = Ls^{(t)}
\]

Where \( s^{(t)} \in \mathbb{R}^{d} \), \( L \in \mathbb{R}^{d \times d} \), ELLA assumes that there are \( k \) latent basis model components among the models of the multiple tasks. Each basis model component \( L \) is represented by a vector of size \( d \), the \( k \) basis model components are represented by a \( d \times k \) matrix \( L \in \{ L_1, \ldots, L_k \} \).

The algorithm is optimized by reducing the average prediction loss value on all tasks. Its objective function is shown in Equation (3):

\[
e_t(L) = \frac{1}{T} \sum_{t=1}^{T} \min_{s^{(t)}} \left\{ \frac{1}{n_t} \sum_{i \in T} \mathcal{L}\left(f(x^{(i)}; \theta^{(t)}), y^{(i)}\right) + \mu \|s^{(t)}\|_1 + \lambda \|L\|_F^2 \right\} (3)
\]

Where \( \lambda \) and \( \mu \) correspond to L1 norm and L2 norm, respectively. To reduce the computational cost of task \( t \) during training, we use the second-order Taylor expansion to optimize \( s^{(t)} \). After optimization, the final objective function is shown in Equation (4):

\[
g(L) = \frac{1}{T} \sum_{t=1}^{T} \min_{s^{(t)}} \left\{ \frac{1}{n_t} \|\theta^{(t)} - Ls^{(t)}\|_2^2 + \mu \|s^{(t)}\|_1 \right\} (4)
\]

Where \( \theta^{(t)} = \min_{s^{(t)}} \frac{1}{n_t} \sum_{i \in T} \mathcal{L}\left(f(x^{(i)}; \theta), y^{(i)}\right) \) is the optimal parameter model of the task \( t \). \( D^{(t)} \) is the Hessian matrix of the loss function \( \mathcal{L} \), which is used to evaluate \( \theta^{(t)} \).

For the extracted characteristic data samples: \( \{ (X, y) \} \) of the current task \( t \), according to the optimization process of the ELLA algorithm. First, use the basic learner (logistic regression or linear regression) to calculate the parameter vector \( \theta^{(t)} \) and Hessian matrix \( D^{(t)} \) of the current task \( t \). Since our task is classification, we chose logistic regression. Their calculation methods are shown in Equation (5) and (6) (the following \( \top \) is the matrix transposition symbol):

\[
\sigma^{(t)} = \frac{1}{1 + e^{-\theta^{(t)} \cdot x^{(t)}}} (5)
\]

\[
D^{(t)} = \frac{1}{2n_t} \sum_{i=1}^{n_t} \sigma^{(t)}(1 - \sigma^{(t)})x^{(t)}x^{(t)\top} (6)
\]

After that, use Equation (7) to calculate \( s^{(t)} \):

\[
S^{(t)} \leftarrow \arg \min_{s^{(t)}} \left( \mu \|s^{(t)}\|_1 + \|\theta^{(t)} - Ls^{(t)}\|_2^2 \right) (7)
\]

Where \( L \) refers to the value of the latent components in the \( m \)-th iteration. Then use Equation (8), (9), and (10) together to adjust the basis latent model components \( L \):

\[
A \leftarrow A + \left( s^{(t)}s^{(t)\top} \right) \otimes D^{(t)} (8)
\]

\[
b \leftarrow b + \text{vec}(s^{(t)} \otimes (\theta^{(t)}D^{(t)})) (9)
\]

\[
L_{s\alpha} \leftarrow \text{mat}\left(\frac{1}{T} A + \lambda I_{s\alpha d} \right)^{-1} \frac{b}{T} (10)
\]

Where \( k \) represents the hidden layer dimension, \( d \) represents the feature dimension. After calculating \( s^{(t)} \) and
### III. THE PROPOSED METHODS

We first clean the relevant text content and then extract content features from the original post and comments, user features from users’ information, and propagation features by constructing a spreading tree according to the transmission relationship between users, next we concat the BERT semantic feature extracted from original post. Therefore, we obtain a high-dimensional feature vector. To prevent high-dimensional feature vectors from causing overfitting during training and reduce redundant features, we use GA-SA to search for the best minimum feature subset. In the process of feature optimization, the proposed algorithm can’t only overcome GA falling into the local optimum, but also improve the convergence speed of the SA algorithm. Based on obtained best minimum feature subset, the LML architecture namely the ELLA algorithm is used to construct the classification model GA-SA-ELLA to achieve the continuous learning of Weibo rumor news, and gradually improve the model effect through knowledge refinement. The output results are 0 and 1, where 0 represents non-rumors (TR) and 1 represents rumors (FR). The processing flow of the GA-SA-ELLA model is shown in Figure 2. It can be seen from the figure that the model consists of two parts: feature extraction and continuous detection of rumors using the GA-SA-ELLA model.

#### A. FEATURES EXTRACTION

There is a big difference between rumors and true news. These differences are reflected in the publisher’s user attributes, news’ propagation attributes, and content [21]. In terms of user attributes, rumor publishers are general users who have not yet been authenticated and are not highly concerned. In terms of content, the correlation between the comments and the content of the source message on the rumors is low, and the rumors news generally have inflammatory negative sentiments, so the sentiments of the comments under the rumor news are generally negative. In terms of propagation attributes, the credibility of news determines the number of reposts and the structure of the propagation network. Based on the above analysis, we have selected a total of 23 features. And these characteristics are divided into four categories: content characteristics, publisher’s user characteristics, propagation characteristics, and semantic characteristics. Table 1 gives a brief introduction to these characteristics. As far as we know, among them: BERT semantic features, \( \text{avg\_repttime} \), \( \text{firstlevel\_ratio} \) and \( \text{gini\_index} \) are considered to be the first application in rumor detection research. The new proposed features have been marked with an "*" mark.

The following will give a detailed introduction to the extraction methods and calculation methods of these features.

1) SENTIMENT CHARACTERISTIC

Compared with true news, rumors can easily arouse public anger and anxiety and thus trigger comments with negative sentiment tendencies [8]. To identify the sentiment tendency of Weibo original news content and its comments,
we use the SnowNLP sentiment analysis tool\(^1\). Since the SnowNLP pre-training model does not use Weibo corpus, a second training is required. First, select Weibo comments corpus that has been classified into emotional categories\(^2\), then use SnowNLP to perform sentiment analysis and save it after training, next use the new trained model to perform sentiment analysis on Weibo events. The sentiment tendency of Weibo’s original news content is directly obtained from this model, with a value of \([-1,1]\), where -1 means negative, and 1 means positive.

The calculation method of sentiment tendency on comments in the literature\(^23\) is:

\[
\text{comment\_sentiment} = \frac{n_{pos} - n_{neg}}{n_{pos} + n_{neg}}, \quad n_{pos} \text{ is the number of positive comments, and } n_{neg} \text{ is the number of negative comments.}
\]

However, if there is a large-scale positive hype or malicious marketing, the results obtained by this calculation method will produce errors.

---

\(^1\) https://github.com/isnowfy/snownlp
\(^2\) https://github.com/SophonPlus/ChineseNlpCorpus/blob/master/datasets/weibo_senti_100k
**TABLE I.** Used Features and their Description (The new proposed features are marked with the "***" mark).

| Content characteristics | Publisher's user characteristics | Propagation characteristics |
|--------------------------|--------------------------------|-----------------------------|
| **Name** | **Description** | **Name** | **Description** | **Name** | **Description** |
| **comment _sentiment** | Sentiment orientations on Weibo comments | **status _count** | Number of user posts | **avg _friends** | The average number of followers who reposted comments and reposts |
| **content _sentiment** | Sentiment orientation on original message content | **followers _count** | Number of user fans | **avg _depth** | The average node depth of the propagation tree |
| **correlation** | Comment relevance | **friends _count** | Number of users following | **repost _count** | The number of Weibo original news reposted |
| **url _count** | Number of URLs in Weibo original news | **bifollowers _count** | The number of mutual fans | **avg _createtime** | The average sign-up time for commented and reposted users |
| **url** | Whether to include URL in Weibo original news | **verified** | Whether the user is authenticated | **avg _replytime** | The average time between all comments and the sub-comments that reply to it |
| **question _ratio** | Rate of correction and question comments | **avg _status** | The average number of posts per day | **firstlevel _ratio** | The percentage of replies to first-level comments |
| **text _len** | Weibo original news’ content length | **followersperday** | The average number of followers per day | **gini _index** | The Gini coefficient in response to top-level comments |
| **attention _rate** | User’s attention rate | **avg _status** | The average number of posts per day | |

According to the literature [33], hype news is generally accompanied by a large number of reposts with small transmission paths, so we assign more weight to comments with large transmission paths, and the value of the weight is consistent with the depth of the comment, thereby reducing the hype behavior of news to a certain extent. The sentiment tendency calculation method is obtained by Equation (11):

$$
\text{comment \_sentiment} = \frac{\sum_{i \in \text{comment}} \text{sentiment}_{i} \times \frac{\text{depth}_{i}}{\text{max\_depth}}}{N}
$$
Where \( N \) is the total number of comments on the Weibo event, \( reply_i \) represents the \( i \)-th comment, \( sentiment_{reply_i} \) is the sentiment polarity of the \( i \)-th comment, its value is the same as the original message content, and \( depth_{reply_i} \) is the depth of the \( i \)-th comment, \( max\_depth \) represents the max depth on all comments or reposts. It can be seen from Equation (11) that the lower the value, the stronger the negative emotion in the comment area.

2) CONTENT RELEVANCE
Content relevance refers to the degree of relevance between the comment content and the source Weibo content. Compared with true news, the content of the comment on the rumor news has a weaker correlation with the content of the source Weibo message [34]. We use Latent Dirichlet Allocation (LDA) to model the source Weibo message content and each comment into the topic space through Gibbs sampling, and then uses the Jensen-Shannon distance to calculate the similarity between the content. Finally, the average of the similarities of all reviews is used as the result for rumor detection.

3) COMMENT CHALLENGE CORRECTION RATE
Compared with true news, rumor news is more questioned and corrected by netizens [35]. In Twitter, researchers often use regular expressions to identify these signals, but using this method in Weibo cannot achieve a high recognition rate. Because people may use direct expressions such as “真的假的” (really?), “胡扯” (baloney), “瞎编” (trump up) and “造谣” (cook up a story and spread it around) when expressing doubts or corrections in Chinese, or they may use idioms, such as “危言耸听” (say frightening things just to raise an alarm), “捕风捉影” (make groundless accusations), etc. In order to identify these complex questioning and correcting signals, we use a self-built questioning and correcting vocabulary corpus. The detailed steps are as follows:
1) Select words with questioning and correcting meanings from the dictionary or the Internet and add them to the initial vocabulary corpus;
2) Segment all comments in the dataset;
3) Use the pre-trained model word2vec in the gensim library to calculate the similarity of words, find the top 100 words in the segmented vocabularies with the similarity of each word from the initial corpus, and add these words to the questioning and correcting vocabulary corpus;
4) Manually screen the words in the vocabulary corpus to remove words that do not contain doubts, correct meanings, and repetitions;
5) Repeat steps 3 and 4 until no new vocabulary can be found.

After the above steps, the established vocabulary corpus contains a total of 394 related words. If a Weibo comment contains a word in the vocabulary corpus, the comment is marked as a questioned and corrected comment. Count the number of questioned and corrected comments under each Weibo event, and use Equation (12) to calculate the questioned correction rate:

\[
\text{question\_ratio} = \frac{n\_quest}{N}
\]  
(12)

Where \( n\_quest \) is the number of queried and corrected comments, and \( N \) is the number of comments under the Weibo event.

4) USER ATTENTION RATE
The account that usually posts the rumors has fewer fans than followers, while the official accounts are the opposite [2]. This shows that the attention of normal users is generally higher than that of rumor-spreading users. To distinguish between normal users and rumor-spreading users, we use user attention rate as one of the indicators to measure the credibility of Weibo news. The calculation method of user attention rate is shown in Equation (13):

\[
\text{attention\_rate} = \frac{\text{followers}}{\text{followers} + \text{friends}}
\]  
(13)

Where \( \text{followers} \) is the number of followers of the publisher, and \( \text{friends} \) is the number of friends of the publisher.

5) NEW PROPOSED TRANSMISSION FEATURES AND BERT SEMANTIC FEATURE
1. Three new transmission features
Compared with true news, rumors can attract the attention of disseminators. Users are more active in the comment area of rumors, and the dissemination network of rumors is more concentrated [36]. The visual transmission diagram of rumors and non-rumors is shown in Figure 3, (a) is the spread chart of non-rumor, the news id corresponding to this figure is "343564243700876", and (b) is the spread chart of rumor, the news id corresponding to this figure is "345646540977567". Moreover, rumors contain a lot of wrong and unconfirmed information in content, which can easily cause doubt and denial when spreading. Compared with normal news events, rumors are more controversial [35]. In the literature[37], the author proposed many features for controversial detection, but some features have not been used for rumor detection (i.e., \text{avg\_replytime}, \text{firstlevel\_ratio} and \text{gini\_index}). Therefore, we use these three features as new features for rumor detection.
FIGURE 3. The visual transmission diagram of rumors and non-rumors chart.

1) avg_replaytime and firstlevel_ratio: The avg_replaytime means the average logged parent-child reply time over pairs of comments, the firstlevel_ratio means the proportion of comments that were top-level (i.e., made in direct reply to the original post). Because users are more active in the comment area of rumors, so the avg_replaytime and firstlevel_ratio value of rumor news are generally greater than normal news;

2) gini_index: The feature means the Gini coefficient of replies to top-level comments and is used to measure the degree of aggregation of discussions in the comment area. The calculation method is to first sort by the number of replies from small to large, and then use Equation (14) to calculate:

$$Gini = 1 - \frac{1}{n} \sum_{i=0}^{n} \left( \frac{\sum_{j=0}^{i} y_j}{\sum_{j=0}^{n} y_j} \right)$$  \hspace{1cm} (14)$$

Where $i$ stands for the $i$-th top-level comment, $y_i$ represents the number of comments that responded to the $i$-th top-level comment, $n$ represents the total number of top-level comments. For example, we calculate the Gini coefficient of Figure 3 (a) as 0.094, and that of Figure 3 (b) as 0.745. Generally speaking, the larger the value, the more the discussion concentrated.

2) BERT semantic features

Compared with other traditional word embedding models, such as Word2vec, Glove, ELMo and BERT can more accurately learn the semantic information and context information of the text, so we use the BERT pre-training model to extract semantic features from the source news for rumor detection.

6) OTHER FEATURES

In addition to the features in Table 1 described in the previous section from 1 to 5, the rest of the features in Table 1 can also effectively distinguish between rumors and non-rumors.

1) Other propagation characteristics: If the news is trustworthy, the farther it will spread, that is, the greater the avg_depth value is;

2) Other content characteristics: Rumors often contain url external links, and the proportion of them is larger than that of normal news;

3) Other user characteristics: Rumor publishers generally use unverified accounts to post false news on the network platform, in other words, the value of verified usually takes the value 0 (1 represents verified); Rumor publishers are generally lower than ordinary users in the number of posts (status_count), the number of fans (followers_count), and the number of mutual followers (bifollowers_count), but are more than ordinary users in the number of followers (friends_count). These features are obtained directly from the corresponding fields in the dataset.

B. RUMOR DETECTION ALGORITHM BASED ON GA-ELLA

As shown in Figure 2, the detailed steps of the GA-ELLA rumor detection algorithm are as follows:

1) Set the initial temperature, end temperature, and cooling coefficient. The cooling procedure of SA corresponds to finding the best solution. When the temperature is set lower, the optimal solution may be missed, but when the temperature is set higher, the search time for the optimal solution is so long that the efficiency of the algorithm is very low.

2) Population coding. We use binary to encode population chromosomes. The dimension of the feature determines the length of the chromosome. Each feature corresponds to a gene on the dye. If the gene is "1", it means that the feature is selected, and "0" means that the feature is not selected. The initial chromosome code sets each gene to "1", which means that all features are selected.

3) Individual fitness calculation. The fitness is determined by the average prediction accuracy on the test set of all tasks based on ELLA. When the average prediction accuracy value on the training set of all tasks is greater, the fitness is greater. The fitness calculation method of an individual is shown in Equation (15).

$$f(x)=\frac{1}{T} \sum_{t=0}^{T} Acc_{x_t}$$  \hspace{1cm} (15)$$

Where $f(x)$ is the individual fitness, $T$ is the total number of tasks, and $Acc_{x_t}$ is the prediction accuracy on the $i$-th task. The description of the algorithm for calculating individual fitness is shown in Algorithm 1.

4) Probability selection based on SA. If the fitness of the offspring is higher than that of the parent, the parent is directly replaced by the offspring. Otherwise, the inferior offspring are accepted according to the probability, and the
probability is determined according to the SA algorithm. The calculation method is as shown in Equation (16):

\[
P(x \rightarrow x') = \begin{cases} 
1, & f(x') < f(x) \\
\exp(-\frac{f(x') - f(x)}{T_i}), & \text{otherwise} 
\end{cases}
\] (16)

Where \( f(x') \) is the fitness of the offspring, and \( f(x) \) is the fitness of the parent. And \( T_i \) is the temperature of iteration \( i \), and \( T_{i+1} \) is the temperature of the last iteration. The determination of \( T_i \), that is, the cooling strategy, is shown in Equation (17):

\[
T_{i+1} = \alpha T_i
\] (17)

Where \( \alpha \) is the cooling coefficient. Analyzing Equations (16) and (17), when the fitness of the offspring is lower than that of the parent, the offspring still has a certain probability of being accepted. The process of cooling is the process of global optimization. If the initial temperature is higher, the probability of finding the global optimal solution is greater, which prevents GA from falling into the local optimal.

5) Population reproduction. The population produces the next generation through crossover and mutation, where the crossover mode selects multipoint crossover, and the mutation operator selects bit mutation.

6) Get the best minimum feature subset. After reaching the end temperature, the chromosome is decoded, and the feature corresponding to the gene "1" constitutes the best minimum feature subset. Based on the best minimum feature subset, the corresponding rumor recognition accuracy rate through ELLA’s classification should also be globally optimal.

The above algorithm description of our model GA-SA-ELLA is shown in Algorithm 2.

**Algorithm 1: CalculateIndividualFitness**

**Input:** \( k, d, \lambda, \mu, \{(X_t, y_t), 1 \leq t \leq T\} \)

**Output:** fitness

```
getFitness(X_t) {
1 model=ELLA(k,d,\lambda,\mu)
2 for( i = 0; i < T; i++)
3 model.train(X_t, y_t)
4 for( i = 0; i < T; i++)
    //Get the accuracy on the test set
5 Fitness_i = model.getscore(X_t, test)
6 fitness = Avg( Fitness_i)
7 return fitness
}
```

**Algorithm 2: GA-SA-ELLA for Rumor detection**

**Input:** \( T_o \) (initial temperature), \( T_{end} \) (end temperature), \( \alpha \) (cooling coefficient), \( P_c \) (crossover probability), \( P_m \) (mutation probability), \( N \)(the number of generations)

```
1 X=getInitialPopulation() //get initial population
while iteration < N
2 F_i = getFitness(X)
3 X=crossover(X)
4 X' =Mutation(X')
5 F_j =getFitness(X ')
Repeat
//Randomly generating new individuals
6 X'='getNewChrom(X')
//Algorithm 1 was used to obtain fitness
7 F_j =getFitness(X ')
//Calculate the acceptance probability using Equation (15)
8 P_r=AcceptancePro(F_j,F_i,T)
9 r=Random()
Until P_r \geq r
10 X' = X'
//Calculate the current temperature using equation (16)
11 T_{i+1} = cooling(\alpha , T_i)
12 thebestfeatureset = X : X' F_j > F_i ?
    iteration ← iteration+1
end while
Return outputs:thebestfeatureset
```

IV. EXPERIMENT AND RESULTS

A. EXPERIMENTAL DATA

In our study, we used a set of Weibo rumor events data publicly available in Ma [4]. The data was collected from the Sina Weibo Community Management Center and API interface. There were a total of 4664 Weibo events, of which 2313 were rumor events and 2351 were non-rumor events. Table 2 shows the statistics of the dataset. The information of each Weibo event was stored in a JSON file, which contains the following information: the number of reposts, the number of comments, the number of likes, timestamps of original news released, original post content and its comments and reposts, publisher, forwarders and commenters’ user information and released original news’ tags, etc.

In order to reduce the noise of extracted features, the experimental data is cleaned first. Numbers, symbols, and English characters in the Weibo events that are irrelevant to the content are removed, only Chinese characters (except for
TABLE 2. Dataset statistics.

|                         | Weibo |
|-------------------------|-------|
| Events                  | 4664  |
| Fake News               | 2313  |
| True News               | 2231  |
| Avg posts / event       | 816   |
| Min of posts/event      | 10    |
| Max of posts / event    | 59318 |
| Earliest release time   | 2010-12-02 00:34:33 |
| Latest release time     | 2015-12-20 15:02:32 |

B. FEATURE NORMALIZATION AND EVALUATION INDEX

In addition to the BERT semantic feature, the other three types of statistical feature values extracted from the data set are too different in the weight range to be directly used for classification, so these feature data need to be normalized first. We use Z-Score Normalization to normalize the features, and the processed feature value range is [-1,1], which takes into account the negative sentiment feature value.

To measure the detection effect of the model, the Accuracy rate (Acc) and F1 score are selected as evaluation indicators. Acc and F1 score are calculated as follows:

\[
\text{Acc} = \frac{TP+TN}{TP+TN+FP+FN} \tag{18}
\]

\[
F1 = \frac{2 \cdot R \cdot P}{P+R} \tag{19}
\]

C. EXPERIMENTAL PLATFORM AND ENVIRONMENT

Our research uses Pycharm as the programming IDE, and the programming language is Python 3.6. A computer with an operating system of Mac OS, a memory of 8 GB, and an Apple M1 CPU is used as the experimental hardware environment.

D. MODEL PARAMETER SETTING

1) DATASET RELATED PARAMETERS

There are a total of 4664 Weibo events in the data set. To reflect the performance improvement of our model for future rumor detection tasks, we first sort all events according to the release time, and then divide all events into T tasks, the number of Weibo events contained in each task \(i\) is \(n_i\). To have enough training data for all tasks, we set the number of events in each task to 1166, so \(T = 4\). The statistics of the earliest and latest release time of news in each task are shown in Table 3. The experiment divides the data samples of each task into the training set and the test set according to the ratio of 5:5, uses 5-fold cross-validation and takes the average of the five experimental results as the experimental results to reduce the contingency of the experimental results.

| Task sequence number | Earliest release time     | Latest release time       |
|----------------------|---------------------------|---------------------------|
| 1                    | 2010-12-02 00:34:33       | 2012-12-11 03:20:45       |
| 2                    | 2012-12-11 03:30:05       | 2013-07-22 17:29:58       |
| 3                    | 2013-07-22 23:56:52       | 2015-11-20 11:21:08       |
| 4                    | 2015-11-20 11:23:52       | 2015-12-20 15:02:32       |

2) DETERMINE THE INITIAL TEMPERATURE

| Initial Temperature | Fitness | Period(seconds) |
|---------------------|---------|-----------------|
| 2000°C              | 0.963   | 271.662         |
| 1500°C              | 0.962   | 230.130         |
| 1000°C              | 0.961   | 163.518         |
| 500°C               | 0.958   | 83.664          |

TABLE 5. GA-SA-ELLA model parameter settings.

| Algorithm | Parameter | Value |
|-----------|-----------|-------|
| GA        | Population size | 100   |
|           | The number of reproducing generation | 100   |
|           | Chromosome length | 122   |
|           | Crossover probability | 0.5   |
|           | Mutation probability | 0.01  |
| SA        | The initial temperature | 1000°C |
|           | End temperature | 0.1°C |
|           | Cooling coefficient | 0.9   |
| ELLA      | \(\mu\) | 0.01  |
|           | \(k\) | 1     |

To find a suitable initial temperature, we set the end temperature as 0.1°C, the cooling coefficient is 0.9, and \(\lambda\) is \(e^{0.5}\). The parameter settings of other ELLA algorithms and GA are shown in Table 5. Based on fitness and optimization time, a traverse search is performed. The detailed results are shown in Table 4.

It can be seen from Table 4 that when the temperature is 500°C, the fitness is 0.958. As the temperature increases, the fitness increases, but the fitness of individuals above 1000°C is not much different. To reduce the optimization time and
improve the efficiency of the algorithm, the initial temperature is chosen to be 1000°C.

3) MODEL PARAMETER SETTING
In the ELLA algorithm, because $k \in [1, \min(10, T/4)]$ and the total number of tasks that is $T$ is 4, the parameter $k$ is 1. The value of the parameter $d$ depends on the feature dimension of GA-SA in the process of feature global optimization. For the selection of the parameter $\lambda$, we traverse from its value range according to the fitness to obtain the optimal $\lambda$. The specific parameter settings of the model GA-SA-ELLA are shown in Table 5.

E. EXPERIMENTAL RESULTS AND ANALYSIS

1) NEW FEATURES AND BERT FEATURE EFFECT ANALYSIS
For all propagation features, including the three new propagation features (avg_replytime, avg_replytime and gini_index), we use Python library treelib to construct according to the propagation relationship between users, next use some methods of the library function to get the relevant information and bring it into the relevant formula to calculate the result. The data results of extracted three new features, we use box plots to display, as shown in Figure 4. Where (a) is avg_replytime, (b) is avg_replytime, (c) is gini_index. It can be seen from Figure 4 that the data statistics obtained are consistent with the theoretical analysis above.

![Diagram showing new features box diagram](Image)

**FIGURE 4.** New features box diagram.

BERT semantic features, using the BERT Chinese pre-training model to perform feature extraction on short Weibo events. According to the feature that the length of Weibo news does not exceed 140, we set the maximum length parameter of the BERT pre-training model is 140, the pooling strategy adopts Maxpooling, and finally obtain a 768-dimensional sentence vector matrix. In order to avoid the curse of dimensionality, reduce the complexity and resource consumption of the algorithm. On the premise of not losing the original data information, we use Principal Component Analysis (PCA) to reduce the dimensionality of the BERT semantic features, and the feature dimension is 100 5 dimensions after dimensionality reduction.

For the three new propagation features proposed and BERT semantic features, we use an unimproved ELLA algorithm to verify their effect. We use the average classification accuracy (Avg_Acc) of all tasks to show whether these new features can effectively identify Weibo rumors. The results of the experiment before and after are shown in Table 6.

| Feature set                  | Algorithm   | $d$   | $\lambda$ | Avg_Acc |
|------------------------------|-------------|-------|-----------|---------|
| 19 basic features            | ELLA (unimproved) | 19    | $e^{-5}$  | 0.935   |
| 19 basic features + 3 new propagation features | ELLA (unimproved) | 22    | $e^{-5}$  | 0.946   |
| 19 basic features + 3 new propagation features + BERT semantic features | ELLA (unimproved) | 122   | $e^{-5}$  | 0.954   |

It can be seen from Table 6 that on the basis of traditional features, when the original ELLA algorithm is used for classification and the parameter $\lambda$ of the algorithm is adjusted to $e^{-5}$, the Avg_Acc of three new propagation features plus the original 19 basic features have reached 0.946, an increase of 1.1% on top of the original basic features, indicating that the three new propagation features can be effectively used for rumor detection. The addition of BERT semantic features (100 dimensions) to 22 features increases the Avg_Acc by 0.8%, which shows that BERT semantic features can effectively improve the classification effect of the model.

2) FEATURE INFLUENCE ANALYSIS
In order to analyze the influence of each feature on the recognition of rumors, based on the ELLA algorithm, the parameter is set to $e^{-5}$, and the average classification accuracy rate of all tasks Avg_Acc is used as the benchmark for comparison. The experimental results are shown in Figure 3.

It can be seen from Figure 5 that the characteristics used in the experiment can distinguish between rumors and non-rumors to a certain extent, but their influence is different.

---

4 https://github.com/caesar0301/treelib

5 The model effect obtained in 100 dimensions is similar to that in 200 dimensions.
Among them, the BERT semantic feature has the greatest influence on classification. This is because the BERT semantic feature dimension is higher than other features, and the semantic feature is to identify rumors from the news content itself. The comment sentiment uses a new calculation method, but shows a good classification effect, indicating that the calculation method is correct. For the question ratio feature, our self-built questioning and correction vocabulary corpus has only 394 words but based on these words to calculate the feature, the classification accuracy achieved on the data set is 0.885. It shows that a vocabulary corpus of this scale has exerted a good classification effect. Other features such as content sentiment, url count, and question ratio have a great influence on the model classification effect, which shows the effectiveness of these features. However, the importance of features such as text len, friends count, repost count, and avg createtime etc. are not as high as that of other features, indicating that these features cannot effectively distinguish between rumors and non-rumors on the dataset.

3) THE INFLUENCE OF FEATURE SELECTION ON MODEL EFFECT

In order to verify that the best minimal feature subset obtained by our model GA-SA-ELLA is more globally optimal than the best minimal feature subset obtained by GA-ELLA without SA improvement, and prove that feature selection can improve the effect of model classification, we use the ELLA algorithm without feature selection to compare the experimental results with GA-ELLA and GA-SA-ELLA. The comparison is shown in Table 7.

| Algorithm     | Optimal $d$ | Optimal $\lambda$ | Avg_Acc |
|---------------|-------------|-------------------|---------|
| ELLA          | 122         | $e^{-5}$          | 0.954   |
| GA-ELLA       | 91          | $e^{-2}$          | 0.957   |
| SA-ELLA       | 97          | $e^{-5}$          | 0.956   |
| GA-SA-ELLA    | 82          | $e^{-5}$          | 0.961   |

It can be seen from Table 7 that the GA-ELLA, SA-ELLA, and GA-SA-ELLA algorithms that can perform feature selection have better detection effects on rumors than the ELLA algorithm without feature selection. In particular, the effect of GA-SA-ELLA is 0.7% higher than that of ELLA. This is because after feature selection, irrelevant and redundant features are removed, which not only reduces the feature dimension, but the degree of improvement of the three models is not the same.

It can be seen from Figure 6 that due to the strong search capability of GA, the optimal fitness in the population increased from 0.9 to 0.943 in less than 10 iterations, but the search capability decreased in the later period and converged in the 62nd iteration. However, compared with GA-SA, although the search speed of GA-SA is not as fast as GA in the early stage, the convergence speed of GA-SA is fast, and
the individual fitness obtained during the convergence is higher. This shows that GA falls into local optimum when it converges.

It can be seen from Figure 7 that the search speed of SA is slow, and there are many flattens in the cooling process, because SA conducts local search near the solution space. When SA was cooled to the termination temperature, the fitness was 0.956, lower than that obtained by GA-SA. If you want to improve the initial temperature or cooling coefficient can be increased, but this will lead to the low efficiency of the algorithm.

Check the best minimal feature subset obtained by GA-ELLA and SA-ELLA model, which remove the text_len, friends_count, and some columns of BERT semantic features. However, the best minimum feature subset obtained by GA-SA-ELLA removes the text_len, friends_count, repost_count, avgcreatime and partial columns of BERT semantic features.

4) CONTINUE LEARNING EFFECT EVALUATION
The model can improve the learning effect of the new task with the help of past knowledge, and the learning effect of the old task does not decrease after learning the new task. In this case, we think the model has the continuous learning ability. To prove the opinion, we use Figure 8 and Figure 9 to show the change in the detection effect of the learned tasks and new tasks respectively through continuous learning. In the two figures, KB_i represents the new KB corresponding to the i-th (i ∈ [1,4]) task after training.

It can be seen from Figure 8 and Figure 9 that in the process of model learning, with the shared knowledge, that is the sparse shared matrix L refined (KB_1 → KB_4), the classification accuracy of learned tasks and new tasks has shown a rising trend. For example, when the KB was updated from KB_2 to KB_4, the accuracy of the learned first task was first improved to 0.873, and finally to 0.895. With the help of historical knowledge, the accuracy of the new task gradually increased from 0.86 to 0.979. In other words, the shared knowledge is refined through continuous learning, the learning effect of the new task will be gradually improved through forwarding knowledge transfer and backward knowledge transfer, the old task will not only overcome catastrophic forgetting but also further improve its model effect.

In addition, although each task only has 1164 Weibo events, the accuracy of model detecting has reached 0.96 since the second task and has slowly improved since then. Therefore, we believe that the GA-SA-ELLA model based on the LML paradigm can achieve the ideal rumor detection effect through knowledge refining, and in the case of having fewer training examples.

5) COMPARISON OF MODEL CLASSIFICATION EFFECT
In order to verify the effectiveness of the GA-SA-ELLA rumor detection model, according to the current mainstream rumor detection algorithm, we choose baseline models from traditional machine learning algorithms and deep learning.
algorithms respectively for comparison. The selected models include: SVM [1], GLAN [6], NM-DPS [7], RDM-HSAM [14], Bi-GCN [9], DGCN [13], VAE-GCN [10] and KMGCN [11]. The detailed experimental comparison results are shown in Table 8. The model is marked with * whose results are taken from the corresponding paper.

A. Traditional machine learning method
(1) SVM. SVM is a commonly used classification algorithm based on traditional machine learning rumor detection algorithms. This algorithm does not carry out feature selection. Through experiments on the feature set extracted in this paper, the results show that it is inferior to our model.

B. Deep learning methods
(1) GLAN. This model not only considers the semantic information of the text, but also considers the global structural information, and uses the attention mechanism to enhance the feature representation.

(2) NM-DPS. The study focus on existing works only utilizing the limited static propagation structures rather than dynamic information. Therefore, the model extracts features of dynamic propagation structures and then integrates structural and textual information for rumor detection.

(3) RDM-HSAM. The model not only considers content, user, and temporal features but also information related to rumor detection, which is the news stance, thus improving the effectiveness of the model.

(4) Bi-GCN. This is the first study of employing GCN for rumor detection. The model not only considers the propagation relationship chains from top to bottom but also considers the propagation feature from bottom to top.

(5) DGCN. The study thought Bi-GCN was limited to representing rumor propagation as a static graph and captured the dynamics of rumor propagations using sequential snapshots and temporal snapshots.

(6) VAE-GCN. The model obtains textual, propagation, and structure information for rumor detection, which uses GCN to learn textual and propagation information and firstly employs VAE to obtain structure information.

(7) KMGCN. The model not only obtains the semantic-level features for each post based on GCN like other researchers, but also retrieves relevant knowledge from KG to help rumor detection, and focus on the visual information in news.

The above deep learning models, their training examples are basically more than 64%, that is, more than 2984 news, the accuracy of best baseline model reaches 0.949. However, when our model learns the second task and has 583 training instances of each task. The average accuracy has reached 0.96. Therefore, compared to these deep learning models, our model is more suitable for dealing with the problem of lack of training data.

It can be seen from Table 8 that our proposed GA-SA-ELLA model is significantly better than other models in the classification index. Because the GA-SA-ELLA model uses the GA-SA algorithm to obtain the best global minimum feature subset, by eliminating redundant features, not only the feature dimension is reduced, but the classification effect of the model can also be improved. And our model has the characteristics of continuous learning, as long as new task data arrives, it would refine the past shared knowledge in KB, so the classification accuracy of the model will become higher and higher until it reaches a larger classification accuracy. Moreover, whether it is models based on traditional machine learning methods or deep learning models, they cannot generate knowledge or do not use past knowledge, and when encountering new rumor data, they can only learn from scratch, especially in the complex and ever-changing online language environment, re-learning will be time-consuming and resource-wasting.

V. CONCLUSIONS AND OUTLOOK

We extracted three types of features of content, users, and propagation from Weibo source content, comments, and reposts, and merged the BERT semantic features of the original message to form feature vectors. On this basis, we used a GA-SA-ELLA model to achieve Weibo rumor detection. The experimental results show that the three new propagation features and the BERT semantic feature could effectively improve the effect of rumor recognition. The best minimal feature subset obtained by using GA-SA is better than the result obtained by using GA and SA alone. At the same time, the GA-SA-ELLA model, with the knowledge refined over time, could result in effectively learning tasks even in the case of less training data for each task. However, there is still considerable room to improve the effectiveness of the rumor detection method. The future work can be considered as follows:

(1) Our work lacks early detection of rumors. Only by discovering rumors as early as possible can the harm caused by the propagation of rumors be further reduced. Therefore, our next step is to further study the early detection of rumors. For example, extracting content features from Rumor Text and storing them in KG as knowledge, which can be retrieved for early detection.

(2) Focus on cross-language and cross-platform rumor detection, such as other minority languages or Chinese minority languages, such as Uygur.

(3) Use other tasks to assist rumor detection. LML can not only learn tasks in the same field, but also cross-field learning. Therefore, LML can be used for continuous learning of rumor detection tasks and other related tasks, so as to continuously improve the learning effect of rumor detection tasks.
REFERENCES

[1] F. Yang, Y. Liu, X. Yu, and M. Yang, “Automatic Detection of Rumor on Sina Weibo,” in Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics, New York, NY, USA, 2012, vol. 13, pp. 1–7. doi: 10.1145/2350190.2350203.

[2] G. Liang, W. He, C. Xu, L. Chen, and J. Zeng, “Rumor Identification in Microblogging Systems Based on Users’ Behavior,” IEEE Trans. Comput. Soc. Syst., vol. 2, pp. 99–108, 2015, doi: 10/ggvsv6.

[3] S. Kwon, M. Cha, K. Jung, W. Chen, and Y. Wang, “Prominent Features of Rumor Propagation in Online Social Media,” 2013 IEEE 13th Int. Conf. Data Min., pp. 1103–1108, 2013, doi: 10/ggrp5h.

[4] J. Ma, W. Gao, Z. Wei, Y. Lu, and K.-F. Wong, “Detecting Rumors Using Time Series of Social Context Information on Microblogging Websites,” Proc. 24th ACM Int. Conf. Inf. Knowl. Manag., pp. 1751–1754, 2015.

[5] F. Yu, Q. Liu, S. Wu, L. Wang, and T. Tan, “A Convolutional Approach for Misinformation Identification,” in IJCAI, 2017, pp. 3901–3907. doi: 10/gm87p7.

[6] C. Yuan, Q. Ma, W. R. Zhou, J. Han, and S. Hu, “Jointly Embedding the Local and Global Relations of Heterogeneous Graph for Rumor Detection,” 2019 IEEE Int. Conf. Data Min. ICDM, pp. 796–805, 2019, doi: 10/gm87qf.

[7] S. Wang, Q. Kong, Y. Wang, and L. Wang, “Enhancing Rumor Detection in Social Media Using Dynamic Propagation Structures,” 2019 IEEE Int. Conf. Intell. Secur. Inform. ISI, pp. 41–46, 2019, doi: 10/gh5rmv.

[8] S. Lv, H. Zhang, H. He, and B. Chen, “Microblog Rumor Detection Based on Comment Sentiment and CNN-LSTM,” 2020, vol. 572, pp. 148–156. doi: 10/gm87q7.

[9] T. Bian et al., “Rumor Detection on Social Media with Bi-Directional Graph Convolutional Networks,” ArXiv, vol. 34(01), pp. 549–556, 2020, doi: 10.1609/aaai.v34i01.5393.

[10] H. Lin, X. Zhang, and X. Fu, “A Graph Convolutional Encoder and Decoder Model for Rumor Detection,” in 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA), 2020, pp. 300–306. doi: 10.1109/DSAA49011.2020.00043.

[11] Y. Wang, S. Qian, J. Hu, Q. Fang, and C. Xu, “Fake News Detection via Knowledge-Driven Multimodal Graph Convolutional Networks,” in Proceedings of the 2020 International Conference on Multimedia Retrieval, New York, NY, USA: Association for Computing Machinery, 2020, pp. 540–547. [Online]. Available: https://doi.org/10.1145/3372278.3390713

[12] N. Bai, F. Meng, X. Rui, and Z. Wang, “Rumor Detection Based on Graph Convolutional Neural Net,” IEEE Access, vol. 9, pp. 21686–21693, 2021, doi: 10.1109/ACCESS.2021.3050563.

[13] J. Choi, T. Ko, Y. Choi, H. Byun, and C. Kim, “Dynamic graph convolutional networks with attention mechanism for rumor detection on social media,” PLoS ONE, vol. 16, 2021, doi: 10/gm87q8.

[14] L. Zeng, B. Wu, and B. Wang, “Detecting Rumor on Microblogging Platforms via a Hybrid Stance Attention Mechanism,” in ICWE, 2020, vol. 12128, pp. 347–364. doi: https://doi.org/10.1109/978-3-030-50578-3_24.

[15] L. Tolosi, A. Tagarev, and G. Georgiev, “An Analysis of Event-Agnostic Features for Rumour Classification in Twitter,” in SMN@ICWSM, 2016, pp. 151–158.

[16] Z. Jahanbakhsh-Nagadeh, M. R. Feizi-Derakhshi, and A. Sharifi, “A semi-supervised model for Persian rumor verification based on content information,” Multimedia Tools Appl., pp. 1–29, 2020.

[17] A. R. Mahlous and A. Al-Laith, “Fake News Detection in Arabic Tweets during the COVID-19 Pandemic,” Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 6, 2021, doi: 10.14569/IJACSA.2021.0120691.

[18] Z. Chen and B. Liu, “Lifelong Machine Learning,” Synth. Lect. Artif. Intell. Mach. Learn., vol. 10, no. 3, pp. 1–145, 2016.

[19] P. Ruvolo and E. Eaton, “ELLA: An efficient lifelong learning algorithm,” in International Conference on Machine Learning, 2013, pp. 507–515.

[20] G. Fei, S. Wang, and B. Liu, “Learning Cumulatively to Become More Knowledgeable,” Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., pp. 1565–1574, 2016, doi: 10/gfx45v.

[21] Y. Song, F. Wang, and X. Chen, “An improved genetic algorithm for numerical function optimization,” Appl. Intell., vol. 49, pp. 1880–1902, 2018, doi: 10/giq5d2.

[22] M. M. Mafarja and S. M. Mirjalili, “Hybrid Whale Optimization Algorithm with simulated annealing for feature selection,” Neurocomputing, vol. 260, pp. 302–312, 2017, doi: 10/gmk9q9.

[23] A. Dang, A. Moh’D, A. Islam, and E. Milios, “Early Detection of Rumor Veracity in Social Media,” in Hawaii International Conference on System Sciences, 2019, pp. 1–10. doi: 10/gm88n2.

[24] J. Ma, W. Gao, and K.-F. Wong, “Detect Rumor and Stance Jointly by Neural Multi-Task Learning,” in Companion Proceedings of the The Web Conference 2018, Republic and Canton of Geneva, CHE, 2018, pp. 585–593. doi: 10.1145/3184558.3188729.

[25] W. Zhou, S. Chang, N. Sosa, H. Hamann, and D. Cox, “Lifelong Object Detection,” ArXiv, vol. abs/2009.01129, 2020.

[26] M. Du, Z. Chen, C. Liu, R. Oak, and D. Song, “Lifelong anomaly detection through unlearning,” in Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, 2019, pp. 1283–1297. doi: 10/gkvnj.
[27] H. Wang, W. Xiong, M. Yu, X. Guo, S. Chang, and W. Y. Wang, “Sentence Embedding Alignment for Lifelong Relation Extraction,” ArXiv, vol. abs/1903.02588, 2019.

[28] X. Hong, S. Guan, K. L. Man, and P. W. H. Wong, “Lifelong Machine Learning Architecture for Classification,” Symmetry, vol. 12, p. 852, 2020, doi: 10/gm87q6.

[29] J. Kirkpatrick et al., “Overcoming catastrophic forgetting in neural networks,” Proc Natl Acad Sci U A, vol. 114, no. 13, pp. 3521–3526, 2016.

[30] L. Shu, B. Liu, H. Xu, and A. Kim, “Lifelong-RL: Lifelong Relaxation Labeling for Separating Entities and Aspects in Opinion Targets,” Proc. Conf. Empir. Methods Nat. Lang. Process. Conf. Empir. Methods Nat. Lang. Process., vol. 2016, p. 225—235, Nov. 2016, doi: 10.18653/v1/d16-1022.

[31] Q. Liu, B. Liu, Y. Zhang, D. S. Kim, and Z. Gao, “Improving Opinion Aspect Extraction Using Semantic Similarity and Aspect Associations,” in Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, Arizona, 2016, pp. 2986–2992.

XIAOXIA HE is currently pursuing the MA.Sc. degree with the Xinjiang University. Her research interests include social network and machine learning.

Gulanbaier Tuerhong received a Ph.D. degree from the Information Management and Security Research Institute, Korea University, in 2013. She is currently an Associate Professor at Xinjiang University. Her research interests include machine learning and network security.

Mairidan Wushouer received a Ph.D. degree from Kyoto University, in 2015. He is currently an Associate Professor at Xinjiang University. His research interests include machine learning and natural language processing.

XIN DAI is currently pursuing the MA.Sc. degree with the Xinjiang University. Her research interests include sentiment analysis and machine learning.

[32] S. Lee, S. Behpour, and E. Eaton, “Sharing Less is More: Lifelong Learning in Deep Networks with Selective Layer Transfer,” in Proceedings of the 38th International Conference on Machine Learning, Jul. 2021, vol. 139, pp. 6065–6075. [Online]. Available: https://proceedings.mlr.press/v139/lee21a.html

[33] C. Castillo, M. Mendoza, and B. Poblete, “Information credibility on twitter,” 2011. doi: 10/bm4s9p.

[34] K. Wu, S. Yang, and K. Q. Zhu, “False rumors detection on sina weibo by propagation structures,” in 2015 IEEE 31st international conference on data engineering, 2015, pp. 651–662. doi: 10/ggvs8w.

[35] M. Mendoza, B. Poblete, and C. Castillo, “Twitter under crisis: can we trust what we RT?,” 2010. doi: 10/d6gc94.

[36] X. Zhou and R. Zafarani, “A survey of fake news: Fundamental theories, detection methods, and opportunities,” ACM Comput. Surv. CSUR, vol. 53, no. 5, pp. 1–40, 2020, doi: 10/gg6q23.

[37] J. Hessel and L. Lee, “Something’s Brewing! Early Prediction of Controversy-causing Posts from Discussion Features,” 2019, pp. 1648–1659. doi: 10.18653/v1/N19-1166.