

# Enhanced Rumor Detection Model for Weibo based on Dual-Layer Multi-Head Attention Mechanism

Jinglong Yao, Xiaofan Jia

School of Information Science and Technology, Xiamen University Tan Kah Kee College,  
Zhangzhou 363123, Fujian, China

---

## Abstract

The spread of rumors on Weibo severely threatens the security of the cyberspace information ecosystem. Their deceptive nature complicates the dynamic coupling mechanism of multi-dimensional semantic features in rumor detection. To address this issue, this paper proposes an enhanced rumor detection model based on a dual-layer multi-head attention mechanism. The model employs a hierarchical strategy to extract key semantic features from word-level to sentence-level in Weibo texts. Additionally, during word-level encoding, it leverages a rumor lexicon to further enhance the semantic features of relevant words, mitigating the feature sparsity problem in rumor detection to some extent. Experimental results demonstrate the model's effectiveness in optimizing feature selection and improving rumor detection performance.

## Keywords

Rumor Detection; Multi-head Attention Mechanism; Weibo; Semantic Features.

---

## 1. Introduction

The rapid development of the Weibo platform has significantly accelerated the depth and breadth of information interaction among users. While providing convenient channels for information dissemination, it has also markedly reduced the cost of rumor propagation. The openness, virtuality, and anonymity characteristic of social media content have created a unique information ecosystem, wherein the spread of rumors not only threatens social stability and public cognition but also poses risks to cyberspace security. Against this backdrop, research on rumor detection on Weibo carries multiple practical implications: it contributes to maintaining a healthy online information ecology, provides technical support for platform governance, and offers reliable references for individual decision-making [1]. Deep learning methods, leveraging their robust feature representation capabilities, have demonstrated significant advantages in the analysis of social media content, enabling the automatic capture of deep semantic features in text for rumor detection [2].

Weibo texts exhibit significant spatiotemporal correlation characteristics, with their textual information demonstrating dynamic evolutionary patterns during the dissemination process. This leads to complex semantic modeling challenges in rumor detection tasks. Rumor information constructs cognitive traps through semantic camouflage and emotional manipulation, and its confounding characteristics demand that detection models possess cross-scale feature parsing capabilities [3]. Existing research indicates that feature selection strategies directly influence the performance ceiling of detection models, yet traditional deep learning methods exhibit notable limitations in key semantic element localization.

To address the aforementioned challenges, this study proposes an enhanced Weibo rumor detection model based on a dual-layer multi-head attention mechanism. The primary contributions include: (1) Constructing a dynamically evolving rumor lexicon by employing a Word2Vec-based cosine similarity iterative algorithm to achieve automatic lexicon expansion; (2) Utilizing an improved TF-

IDF algorithm to quantify the comprehensive rumor weight of words and generate vectorized representations; (3) Proposing a dual-layer multi-head attention mechanism that collaboratively captures key dissemination features at both the word level and sentence level.

## 2. Related Work

With the rapid development of social networks, information security issues caused by online rumors have become a hotspot in interdisciplinary research. To address the challenges of feature selection in rumor detection tasks, existing studies primarily explore three technical approaches: (1) Employing deep learning-based methods to automatically capture deep semantic features of text through end-to-end learning; (2) Incorporating external knowledge bases to enhance domain-specific feature semantic spaces; (3) Applying attention mechanisms to dynamically optimize feature selection and weight allocation.

The textual content exhibits high recreatability, and publishers frequently employ anonymity, rendering rumor detection tasks exceptionally complex. Within the deep learning paradigm, time-series modeling occupies a pivotal position. The network architecture of Recurrent Neural Networks (RNN) demonstrates inherent compatibility with the dissemination characteristics of Weibo information. However, their intrinsic gradient explosion problem severely constrains the modeling capability for long-range dependencies. To address this limitation, approaches based on Long Short-Term Memory (LSTM) networks emerged [4]. The Bidirectional LSTM (Bi-LSTM) further enhances performance by combining two directional LSTM, where the forward and backward propagation layers work collaboratively to effectively resolve the information truncation issue inherent in unidirectional LSTM [5]. But existing methods still exhibit notable limitations: when individual Weibo posts contain lexically divergent word combinations, neither unidirectional nor bidirectional linear sequence modeling can adequately capture global semantic dependencies, ultimately leading to detection inaccuracies.

Lexicons, as crucial resources in the field of natural language processing, play a pivotal role in enhancing models' semantic comprehension capabilities. By constructing domain-specific lexicons that exhibit high compatibility with particular tasks, neural networks can more effectively capture task-relevant semantic features, thereby significantly improving model performance while reducing computational overhead. Existing research demonstrates that the selection of lexicon collaboration strategies directly impacts task execution efficiency [6].

The attention mechanism, as a crucial paradigm in brain-inspired computing, draws its inspiration from the human visual system. By dynamically allocating computational resources to focus on task-relevant features, it significantly enhances model processing efficiency. Well-designed attention modules can lead to substantial improvements in model performance. The multi-head attention mechanism proposed by Google researchers [7] represents a paradigm shift in attention computation, whose parallel computation across multiple subspaces enables the capture of key features at different semantic levels.

## 3. Rumor Detection Model

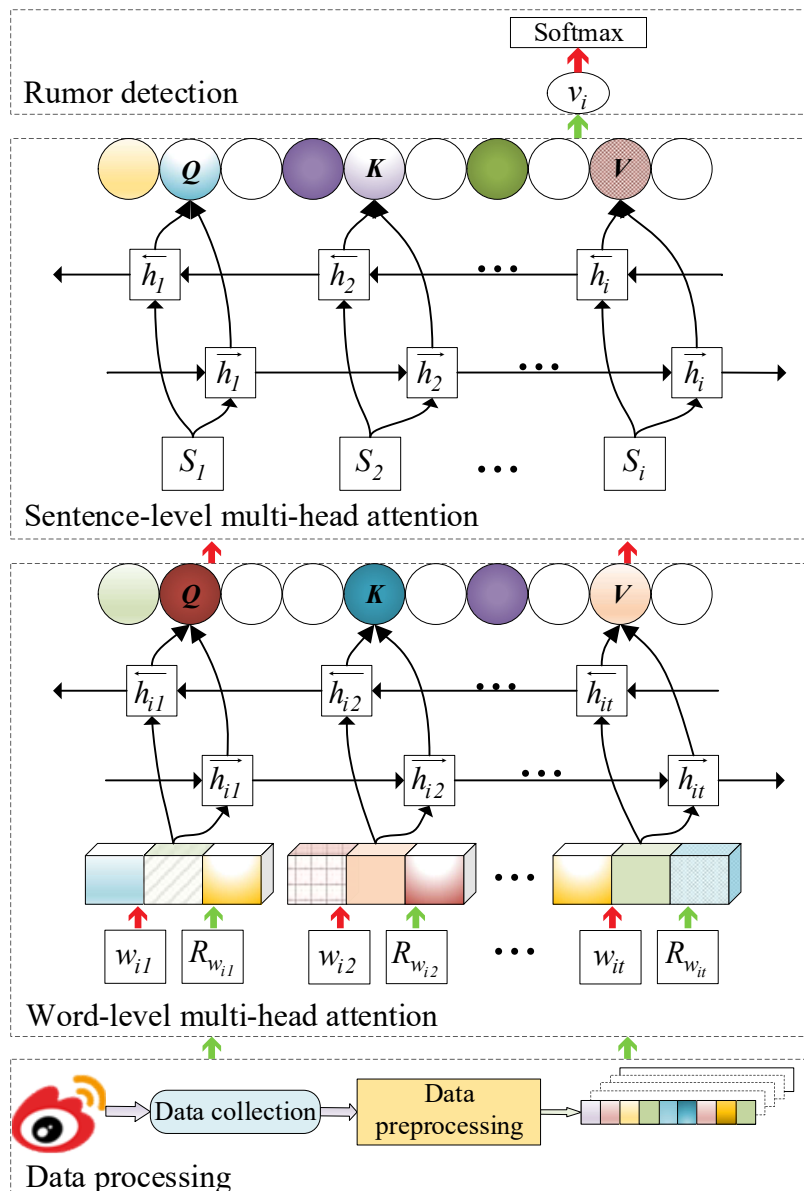
To extract the most effective rumor detection features from Weibo texts, this study employs a dual-layer multi-head attention mechanism while incorporating rumor lexicon collaboration in word encoding, thereby constructing an enhanced rumor detection model based on this architecture. The overall framework of the model is illustrated in Fig. 1. Based on functional differentiation, the model is divided into four components: data processing, word-level multi-head attention, sentence-level multi-head attention, and rumor detection.

### 3.1 Data Processing

This study collected relevant data from the Sina Weibo platform to construct a rumor detection dataset comprising 15,000 rumor samples and 15,000 authentic Weibo posts. The rumor samples were

obtained from the official public notice section of Sina Weibo Community Management Center's misinformation reporting system, while the authentic samples consisted of content published by verified user accounts.

For subsequent word vector representation, it is necessary to transform the comprehensive rumor weight of words into vector form. This process requires prior construction of a rumor lexicon based on lexical characteristics in rumor texts. The remainder of this section elaborates on four key components: rumor lexicon construction, acquisition of fundamental rumor word sets, lexicon expansion, and computation of comprehensive rumor weights.



**Fig. 1** Enhanced rumor detection model for weibo based on dual-layer multi-head attention mechanism

Construction of a rumor lexicon: This study proposes a hybrid approach for constructing a Weibo rumor lexicon by integrating manual screening with automatic expansion, based on an analysis of keyword features in rumors. First, the raw corpus undergoes preprocessing, followed by manual annotation to identify a foundational word set with distinct rumor characteristics, while the remaining words form a candidate word set. Subsequently, a Word2vec model is employed to construct a word vector space. Cosine similarity is computed between candidate words and the foundational word set

to enable automatic vocabulary expansion. A predefined threshold is applied to filter eligible candidates, which are then incorporated into the expanded lexicon. To enhance the lexicon's coverage, an iterative optimization strategy is adopted: high-confidence words from the expanded lexicon are dynamically reintegrated into the foundational word set for repeated computation until no new expansions can be identified. The detailed construction process is illustrated in Fig. 2.

Extraction of the foundational rumor word set: This study established the foundational rumor lexicon through systematic preprocessing of Weibo rumor corpora, beginning with data deduplication followed by the removal of interfering special characters (e.g., "@", "&"). Text processing was performed using the Jieba segmentation tool, with segmentation accuracy enhanced through the integration of a comprehensive dictionary combining mainstream lexicons from Baidu and Sogou. The segmented results underwent part-of-speech tagging to extract relevant lexical items. Based on linguistic feature analysis of rumors, the study identified four representative lexical categories through part-of-speech filtering: nouns were prioritized for their core role in subject description and semantic support within rumors; high-frequency verbs in specific contexts were included due to their strong correlation with rumor types; topic markers denoted by special symbols (e.g., "#", "[ ]") were emphasized for their salient informational representation; and specific word collocations exhibiting frequent co-occurrence patterns in rumor texts were also collected. Consequently, the study selected nouns, verbs, topic markers, and associated word collocations as the foundational rumor lexicon for subsequent expansion.

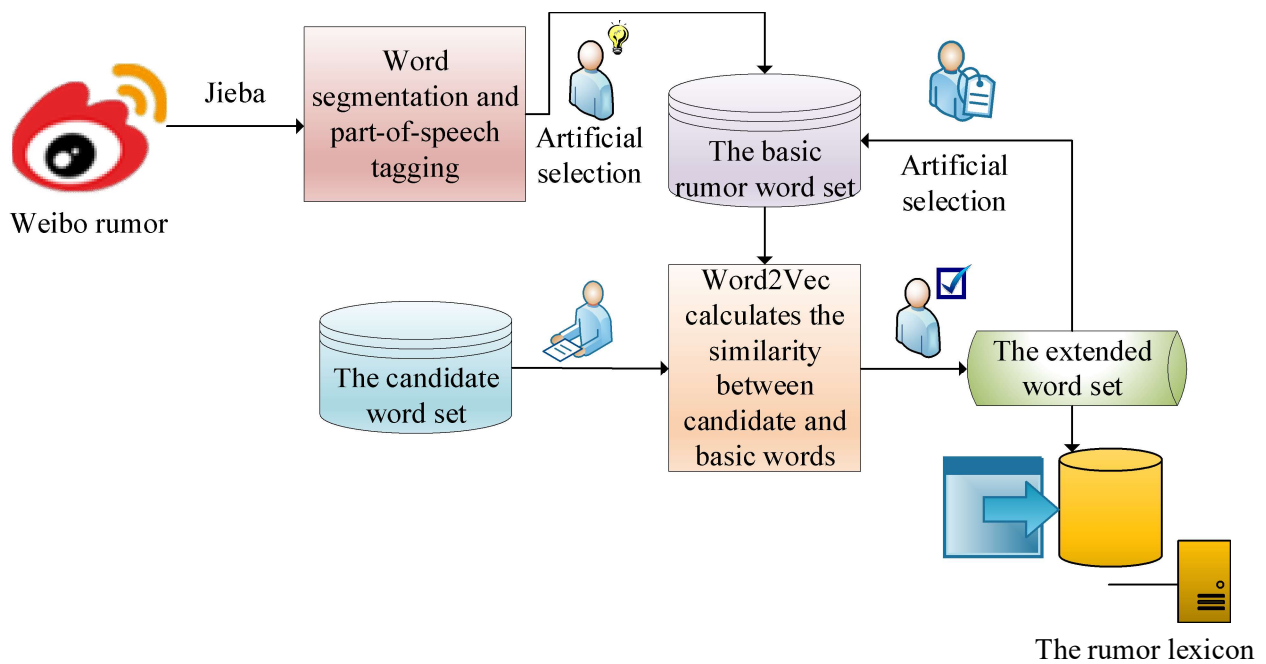


Fig. 2 Framework of Weibo Rumor Lexicon Development

To efficiently filter candidate terms from the corpus, we computed the cosine similarity between each word in the candidate set and the foundational rumor lexicon using Word2Vec embeddings, as formalized in Equation (1):

$$similarity(A, B) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n (A_i)^2} \times \sqrt{\sum_{i=1}^n (B_i)^2}} \quad (1)$$

Let candidate word set be denoted as A and the foundational rumor lexicon as B. For each term in A, we computed the cosine similarity between its word vector and all word vectors in B. Any term exhibiting similarity values exceeding the predefined threshold ( $\theta = 0.9$ ) was incorporated into the expanded lexicon. Through manual verification, qualified terms were subsequently added to the foundational lexicon B, establishing an incremental iteration process. This algorithm executed iteratively until no additional relevant terms could be identified, at which point the final rumor lexicon was output.

**Comprehensive rumor weight calculation:** This study employs a modified TF-IDF algorithm to compute comprehensive rumor weights, implementing three key optimizations tailored to rumor text characteristics: reweighting coefficients to prioritize term frequency over inverse document frequency given rumors' propagation through modified key nouns, applying cosine normalization to eliminate microblog length variations, and incorporating logarithmic transformation of term frequencies to mitigate outlier effects while neutralizing disproportionate impacts of frequency magnitudes.

The formula for computing comprehensive rumor weights is formally expressed as Equation (2):

$$R(w) = \frac{\log_2(\varphi tf + 1) \times \log_2\left(\frac{N}{(1-\varphi)idf}\right)}{\sqrt{\sum_i \left(\log_2(\varphi tf_i + 1) \times \log_2\left(\frac{N}{(1-\varphi)idf_i}\right)\right)^2}} \quad (2)$$

Where  $N$  denote the total number of rumors and  $\varphi$  represent the weighting coefficient for term frequency. The comprehensive rumor values of words computed through this formulation are transformed into vectorized representations, which are subsequently fed into the model alongside the lexical items for training.

### 3.2 Word-Level Multi-Head Attention

To capture the core semantics of sentences in Weibo texts, a multi-head attention mechanism is employed to identify pivotal lexical items. This procedure comprises three sequential stages: (1) word vector representation, where lexical units are transformed into distributed embeddings; (2) contextual encoding, which generates sequential representations through bidirectional recurrent networks; (3) lexical attention, where multiple parallel attention heads dynamically compute weight distributions to extract semantically critical terms.

**Word Vector Representation:** Vectorized representations of words effectively capture semantic features, with their quality directly influencing semantic characterization of Weibo texts and rumor detection performance. This study introduces the constructed rumor lexicon at the representation layer, adopting a dual-channel input architecture: (1) distributed vector representations encoding lexical semantics, and (2) comprehensive rumor weights derived from the enhanced TF-IDF algorithm quantifying discriminative importance for rumor detection. This integrated approach preserves inherent semantic features while incorporating domain-specific discriminative signals,

thereby synergistically enhancing the model's semantic comprehension capability and detection performance, the calculation formula is shown in Equation (3):

$$\mathbf{x}_{it} = W_e w_{it} \oplus R_{w_{it}}, t \in [1, T] \quad (3)$$

Among them,  $x_{it}$  represents the word vector of the  $t$ -th word in the  $i$ -th sentence,  $W_e w_{it}$  represents the word vector of the word  $w_{it}$ ,  $R_{w_{it}}$  represents the comprehensive rumor value of the word,  $T$  represents the sentence length, and  $W_e$  is the 200-dimensional word vector pre-trained using the Word2Vec tool.

**Contextual Encoding:** At each timestep  $t$ , the LSTM unit achieves fine-grained information regulation through gating mechanisms, comprising three core components: the input gate  $i_t$ , forget gate  $f_t$ , output gate  $o_t$ , and memory cell  $c_t$ . The forget gate  $f_t$  controls retention of prior cell state  $c_{t-1}$ , the input gate  $i_t$  modulates updates from the candidate state  $\tilde{c}_t$ , and the output gate  $o_t$  governs information exposure to the hidden state. The forward LSTM processes sequences left-to-right via chained units, with propagation formalized as: for input sequence  $x_1, x_2, \dots, x_{it}$ , iterative computation yields hidden states

$\vec{h}$ . The forward propagation equations are given by Equation (4):

$$\vec{h}_{it} = \overline{\text{LSTM}(x_{it})} \quad (4)$$

The reverse LSTM layer processes sequences right-to-left through chained units. For a given input sequence  $x_{it}, \dots, x_2, x_1$ , it generates hidden state outputs  $\overleftarrow{h}$ . The backward propagation equations are formalized in Equation (5):

$$\overleftarrow{h}_{it} = \overline{\text{LSTM}(x_{it})} \quad (5)$$

The outputs from forward and reverse LSTM layers are combined through Equation (6):

$$\overleftrightarrow{h}_{it} = \vec{h}_{it} \oplus \overleftarrow{h}_{it} \quad (6)$$

Where  $\vec{h}_{it}$  denote the forward LSTM layer output for the  $t$ -th word in the  $i$ -th sentence,  $\overleftarrow{h}_{it}$  represent the backward LSTM layer output, and  $\overleftrightarrow{h}_{it}$  signify the Bi-LSTM-encoded output.

**lexical attention:** Given the compositional heterogeneity of lexical categories in Weibo texts and the significant variation in semantic contribution among words, this study introduces a multi-head attention mechanism to capture critical semantic information. The mechanism generates discriminative sentence-level vector representations by aggregating lexical representations with high semantic weights. The multi-head attention architecture comprises multiple stacked scaled dot-product attention units, where input matrices are designated as Query (Q), Key (K), and Value (V),

with  $Q, K, V \in \mathbb{R}^{n \times d}$ . The scaled dot-product attention consists of  $h$  layers, with the attention computation at each layer formalized in Equation (7):

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \quad (7)$$

Here,  $d$  denotes the dimensionality of hidden units in the neural network's hidden layer. The proposed model is built upon the self-attention mechanism, where input vectors undergo three distinct linear transformations—each governed by its own unique parameter matrix. During computation, the association weights between query vector  $\mathbf{Q}$  and all key vectors  $\mathbf{K}$  are first calculated using a dot-product similarity function, followed by a scaling operation that divides the results by the square root of  $\mathbf{K}$ 's dimensionality to prevent excessive inner product values. Subsequently, a softmax function normalizes these weights, which are then used to compute a weighted sum with their corresponding value vectors  $\mathbf{V}$ , yielding the output of a single attention head. This process is repeated  $h$  times across parallel attention heads, after which the outputs from all heads are concatenated and transformed via a final linear layer to produce the multi-head attention mechanism's ultimate output, the calculation formula is shown in Equation (8) and Equation (9).

$$\text{head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) \quad (8)$$

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_h)\mathbf{W}^o \quad (9)$$

Here,  $\mathbf{W}^o$  denotes the weight matrix of the linear transformation, and  $s$  represents the final computed output value, which is designed to capture richer feature information learned from diverse positions or spatial contexts, the calculation formula is shown in Equation (10).

$$s = \text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \quad (10)$$

Finally, a max-pooling layer is applied to perform compression, yielding the most influential sequence  $S_i, i \in [1, L]$ ,  $L$  denotes the number of sentences in this context.

### 3.3 Sentence-Level Multi-Head Attention

Following the extraction of key word features via the word-level multi-head attention mechanism, we further construct sentence-level representations. This method directly utilizes the output of the word-level multi-head attention without requiring additional vector representation transformations, encompassing two primary steps: sentence sequence encoding and sentence attention.

**Sentence Sequence Encoding:** In rumor detection tasks, semantic features at the sentence level exert a decisive impact on detection performance. To address this, this paper employs a Bi-LSTM network to model the contextual dependencies of sentence sequences. By leveraging the cooperative operation of forward and backward LSTM layers, this network captures long-range semantic correlations and deep contextual features within the text, the calculation formula is shown in Equation (11).

$$\begin{cases} \overrightarrow{h}_i = \overrightarrow{\text{LSTM}}(S_i) \\ \overleftarrow{h}_i = \overleftarrow{\text{LSTM}}(S_i) \\ \overline{h}_i = \overrightarrow{h}_i \oplus \overleftarrow{h}_i \end{cases} \quad (11)$$

Here,  $\vec{h}_i$  denotes the output of the i-th sentence encoded by the forward LSTM,  $\overleftarrow{h}_i$  represents the output of the i-th sentence encoded by the backward LSTM, and  $\overleftrightarrow{h}_i$  stands for the output of the i-th sentence encoded by the Bi-LSTM. Subsequently, these encoded outputs are fed into the sentence attention layer to extract the most influential components.

**Sentence Attention:** In the semantic representation of Weibo texts, different sentences exhibit significant disparities in their contributions to the overall semantics. To address this, this paper employs a multi-head attention mechanism to model the importance of the sentence encoding sequence. During computation, each sentence engages in attention interactions with all other sentences in the sequence, learning inter-sentence correlations via the multi-head attention mechanism. This process ultimately yields sentence features  $v$  with importance weights, reflecting their relative contributions to the overall semantic representation, the calculation formula is shown in Equation (12):

$$v = \text{MultiHead}(\mathbf{Q}', \mathbf{K}', \mathbf{V}') \quad (12)$$

Finally, a max-pooling operation is applied to compress the features of the sentences, yielding the most influential sentence sequence  $v_i$ .

### 3.4 Rumor Detection

After processing through the word-level and sentence-level multi-head attention mechanisms, the final features of Weibo texts can maximally represent their semantic information. For rumor detection, a true-false binary classification discriminator is constructed to classify the deep-level feature information learned through multiple layers, thereby yielding the final detection results. This paper employs the Softmax function to construct the rumor detection classifier, the calculation formula is shown in Equation (13):

$$p = \text{softmax}(W_c v_i + b_c) \quad (13)$$

Here,  $p$  represents the probability that the Weibo content is a rumor. The objective function of this paper employs the negative log-likelihood function as the training loss function, the calculation formula is shown in Equation (14):

$$\mathcal{L} = -\sum_d \log(p_{dz}) \quad (14)$$

Here,  $z$  denotes the label indicating whether text  $d$  is a rumor.

## 4. Experimental Results and Evaluation

### 4.1 Experimental Data and Related Settings

The experiment employs stratified 10-fold cross-validation, with the average scores of evaluation metrics across the ten folds serving as the final performance metrics for rumor detection. The evaluation metrics include accuracy, precision, recall, and F1-score. The adjustable parameters of the model are presented in Table 1.

**Table 1.** Adjustable Parameter Settings

Tunable Parameters	values
Vector Embedding Dimension	200
Learning Rate	0.001
Optimizer	Adam
Batch_size	64
Dropout	0.3
Number of layers for multi-head attention	8

The vector embedding dimension is set to 200 dimensions, effectively balancing semantic representation capability and computational complexity, a learning rate of 0.001 paired with the Adam optimizer ensures model convergence speed and stability, the Batch\_size is configured to 64, balancing memory utilization and gradient update efficiency, the Dropout rate is set to 0.3, effectively mitigating overfitting and preserving model generalization capability, the 8-layer multi-head attention mechanism ensures comprehensive capture of deep semantic features in the text.

Additionally, to evaluate the rumor detection capability of the proposed model, experimental comparisons will be conducted with the following model methods: CNN, R-CNN, H-BLSTM, and Att-BiLSTM. The data partitioning method for these comparative experiments also adopts stratified 10-fold cross-validation.

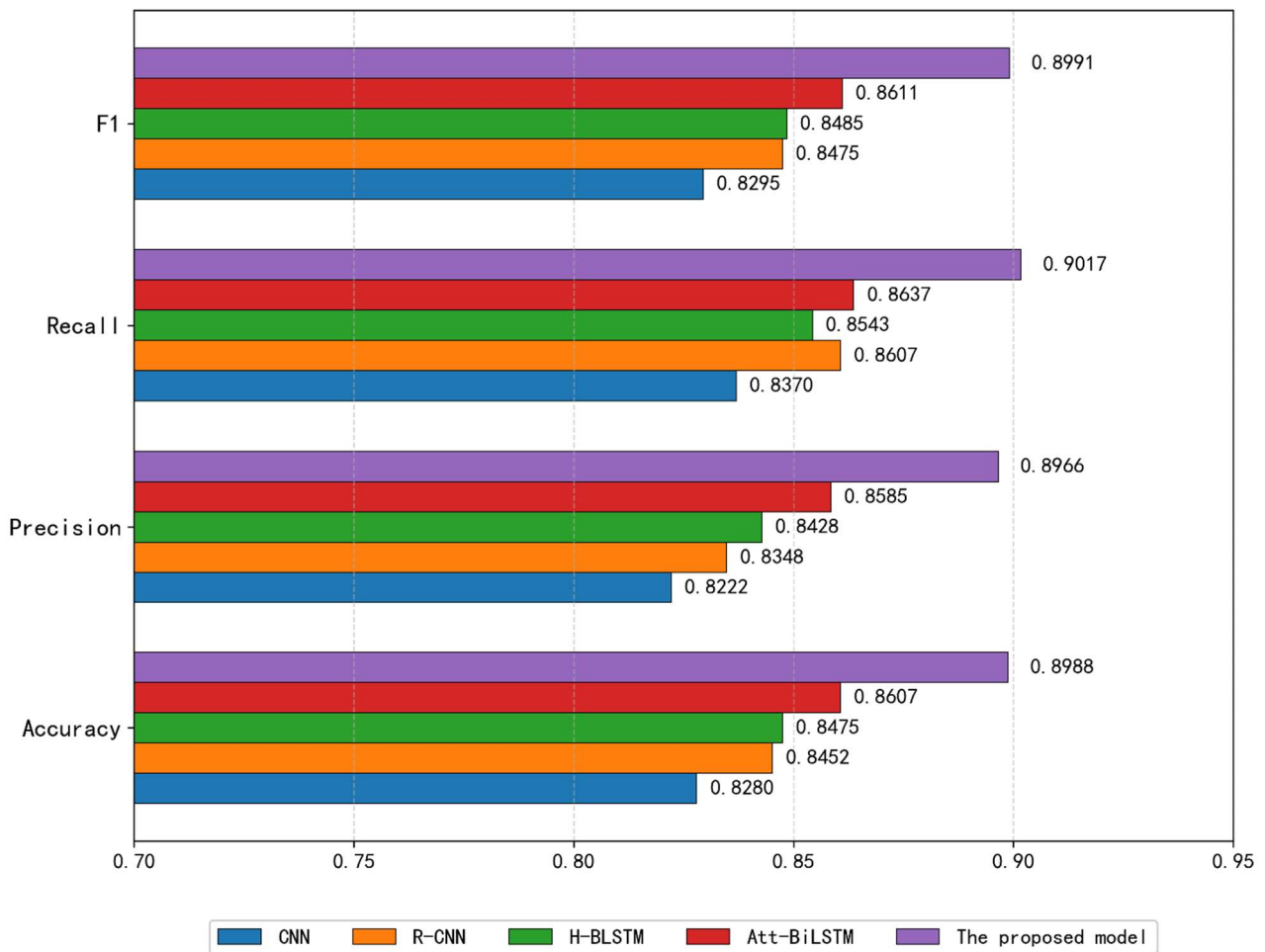
#### 4.2 Experimental Results and Analysis

Under the rumor detection experimental dataset, this paper conducts classification comparison experiments using multiple models to validate the effectiveness of the Weibo rumor detection enhanced model based on a dual-layer multi-head attention mechanism. To ensure objectivity and fairness, the data preprocessing methods are consistent with those of the proposed model, and experimental results are evaluated using rumor detection-related metrics. The final accuracy, precision, recall, and F1-score results for each model are presented in Table 2. To more intuitively reflect the numerical differences in detection performance compared to other baseline models, a bar chart is used to illustrate the performance comparison between the proposed model and the comparison models for rumor detection, as shown in Fig. 3.

This study validates the performance of various models through comparative experiments, among which the rumor detection enhanced model based on a dual-layer multi-head attention mechanism achieves the most excellent results across all evaluation metrics. This outstanding performance is primarily attributed to the synergistic configuration of the model's multi-head attention mechanism and rumor lexicon, with its advantages reflected in multiple aspects. In terms of feature learning, the multi-head attention mechanism can learn diverse text features in parallel across different subspaces, thereby capturing richer semantic information; in terms of semantic understanding, the model demonstrates strong key information extraction capabilities, which are key to its superior performance in rumor detection. The Att-BiLSTM model exhibits superior performance in the rumor detection task due to its bidirectional feature learning capability and the incorporation of attention mechanisms. Compared with the proposed model, a notable difference lies in the fact that the attention mechanism of Att-BiLSTM performs only a single computation, whereas the proposed model conducts multiple feature refinements through a multi-layer multi-head attention mechanism, enabling more comprehensive capture of discriminative features in the text. These comparative results fully demonstrate the importance of attention mechanisms in rumor detection research and indicate that deep computation of attention mechanisms can further enhance model performance.

**Table 2.** Experimental Results of the Proposed Model and Baseline Models

Model	Accuracy	Precision	Recall	F1-score
CNN	0.8280	0.8222	0.8370	0.8295
R-CNN	0.8452	0.8348	0.8607	0.8475
H-BLSTM	0.8475	0.8428	0.8543	0.8485
Att-BiLSTM	0.8607	0.8585	0.8637	0.8611
The proposed model	0.8988	0.8966	0.9017	0.8991



**Fig. 3** Model Performance Comparison

## 5. Conclusion

This study addresses the feature selection issue in Weibo rumor detection by proposing a dual-layer multi-head attention mechanism-based model for Weibo rumor detection. At the feature extraction level, the model employs a hierarchical multi-head attention mechanism architecture: the first layer of multi-head attention focuses on word-level key feature extraction, enhancing semantic representation by integrating our constructed rumor lexicon; the second layer performs sentence-level feature aggregation, automatically identifying core features in the text. Experimental results demonstrate that the model significantly outperforms baseline models across core metrics such as accuracy, precision, recall, and F1-score. Notably, it exhibits stronger identification capability in handling subtle rumors, which attributes to the multi-head attention mechanism's ability to capture subtle semantic changes.

In future research, we will deepen our investigations in three directions:

- 1) Multi-modal Feature Fusion: Explore vision-text cross-modal attention mechanisms and design dynamic weight allocation strategies tailored to Weibo's unique image-text hybrid features.
- 2) Large Model Adaptation: Investigate prompt enhancement methods based on large language models, leveraging knowledge distillation to transfer the reasoning capabilities of large models to rumor detection frameworks.
- 3) Real-time Detection Framework: Develop incremental learning algorithms for streaming data, integrating the feature selection characteristics of attention mechanisms to achieve efficient real-time detection.

## References

- [1] Otabek Sattarov, Jaeyoung Choi. Detection of Rumors and Their Sources in Social Networks: A Comprehensive Survey. *IEEE Transactions on Big Data*. 2025, Vol. 11 (No. 3), p. 1528-1547.
- [2] Barsha Pattanaik, Sourav Mandal, Rudra M. Tripathy. A survey on rumor detection and prevention in social media using deep learning. *Knowledge and information systems*, 2023, Vol. 65 (No. 10), p. 3839-3880.
- [3] Daniel Wai Kit Chin, Kwan Hui Lim, Roy Ka-Wei Lee. RumorGraphXplainer: Do Structures Really Matter in Rumor Detection. *IEEE Transactions on Computational Social Systems*, 2024, Vol. 11 (No. 5), p. 6038-6055.
- [4] Tajinder Singh, Madhu Kumari, Rumor identification and diffusion impact analysis in real-time text stream using deep learning. *The Journal of Supercomputing*, 2023, Vol. 80 (No. 6), p. 7993-8037.
- [5] Pritika Bahad, Preeti Saxena, Raj Kamal. Fake News Detection using Bi-directional LSTM-Recurrent Neural Network. *Procedia Computer Science*, 2020, Vol. 165, p. 74-82.
- [6] Daniel A. Guttentag, Maryam Mahdikhani, Chris Starr, et al. How Airbnb guest experiences differ by accommodation type: a lexicon-based sentiment analysis of Airbnb reviews. *Tourism Recreation Research*, 2024, p. 1-15.
- [7] Jian Li, Xing Wang, Zhaopeng Tu, et al. On the diversity of multi-head attention. *Neurocomputing*, 2021, Vol. 454, p. 14-24.