



Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.



Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

A novel rumor detection with multi-objective loss functions in online social networks

Pengfei Wan^{a,b}, Xiaoming Wang^{a,b,*}, Guangyao Pang^{a,b}, Liang Wang^{a,b}, Geyong Min^c^a School of Computer Science, Shaanxi Normal University, West Chang'an Avenue, Xi'an, 710119, Shaanxi province, China^b Key Laboratory of Modern Teaching Technology, Ministry of Education, West Chang'an Avenue, Xi'an, 710119, Shaanxi province, China^c Department of Computer Science, University of Exeter, The Queen's Drive, Exeter, EX44QF, Devonshire, United Kingdom

ARTICLE INFO

Keywords:

Online social networks
Rumor detection
Neural network
Multi-objective optimization
Sliding interval

ABSTRACT

COVID-19 quickly swept across the world, causing the consequent infodemic represented by the rumors that have brought immeasurable losses to the world. It is imminent to achieve rumor detection as quickly and accurately as possible. However, the existing methods either focus on the accuracy of rumor detection or set a fixed threshold to attain early detection that unfortunately cannot adapt to various rumors. In this paper, we focus on textual rumors in online social networks and propose a novel rumor detection method. We treat the detection time, accuracy and stability as the three training objectives, and continuously adjust and optimize this objective instead of using a fixed value during the entire training process, thereby enhancing its adaptability and universality. To improve the efficiency, we design a sliding interval to intercept the required data rather than using the entire sequence data. To solve the problem of hyperparameter selection brought by integration of multiple optimization objectives, a convex optimization method is utilized to avoid the huge computational cost of enumerations. Extensive experimental results demonstrate the effectiveness of the proposed method. Compared with state-of-art counterparts in three different datasets, the recognition accuracy is increased by an average of 7%, and the stability is improved by an average of 50%.

1. Introduction

At the end of 2019, a disaster caused by the novel coronavirus (COVID-19) quickly swept across the world. As a result, the public health crisis caused by the pandemic brought unprecedented losses to the world (Wu et al., 2020). Different from the past, while the virus is spreading rapidly, the false information represented by rumors proliferates through the flood of globalized online social media (Zarocostas, 2020). The diffusion of information is faster than the virus, which has directly spawned the “second battlefield” for countries around the world. As the Director-General of the World Health Organization (WHO) Tedros said that: “We are not just fighting an epidemic, we are fighting an infodemic” (Lancet, 2020).

The vigorous rising of large-scale online social medias such as Weibo, Twitter and Facebook have replaced traditional media as an important platform for people to obtain and release information. Due to its fast speed, wide range, and strong immediacy, these platforms have become a hotbed of rumors. Lacking of effective supervision, unprocessed rumors may be quickly distorted and amplified which creates a highly uncertain information environment, thereby misleading the

public, affecting social stability, and even threatening national security. Rumors have become a more and more serious social problem than ever that requires the attentions and efforts of the whole society (Leng et al., 2021). Effectively identifying rumors is a very challenging task. Some rumors have been processed layer by layer and carefully packaged, and sometimes they are enough to “make the false true”. For instance, some rumormongers fabricate the rumor that the vaccine is being used to track or microchip people, which has terrible lethality. It has further spread the epidemic, caused widespread panic among the public and seriously jeopardized national stability (Maryland, 2021). But in fact, there is no vaccine microchip, and the vaccine will not track people or gather personal information into a database. The knowledge, experience, and emotions of different users varies greatly in all aspects, leading to the fact that it is almost impossible for users to identify rumors merely based on their cognitive. The knowledge, experience, and emotions of different users varies greatly in all aspects, leading to the fact that it is almost impossible for users to identify rumors merely based on their cognitive.

* Corresponding author at: School of Computer Science, Shaanxi Normal University, West Chang'an Avenue, Xi'an, 710119, Shaanxi province, China.

E-mail addresses: wanpf@snnu.edu.cn (P. Wan), wangxm@snnu.edu.cn (X. Wang), panguangyao@snnu.edu.cn (G. Pang), wangliang@snnu.edu.cn (L. Wang), g.min@exeter.ac.uk (G. Min).

<https://doi.org/10.1016/j.eswa.2022.119239>

Received 1 February 2022; Received in revised form 5 November 2022; Accepted 5 November 2022

Available online 11 November 2022

0957-4174/© 2022 Elsevier Ltd. All rights reserved.

Fortunately, more and more researchers are devoted to exploring and addressing this issue. Some anti-rumor platforms have been constructed by government departments and civil organizations, such as “Piyao.com” established by the Cyberspace Administration of China and “Snopes.com” established by an American couple. These platforms take advantages of crowd sensing to encourage users to actively identify and report suspicious information, then deliver suspicious information to professional researchers to refute rumors with scientific explanation. Although this manual-based detection method has high accuracy (Li, Li, Wang, & Wang, 2018; Xu, Rao, Xu, Yang, & Li, 2019; Zhou et al., 2022), it needs to go through a series of processes which has obvious time-lag and cannot adapt to the massive data in online social networks (OSNs). Moreover, the crowd sensing technology faces a cold start problem which requires numerous human efforts. Some researchers are starting to utilize machine learning techniques to automatically identify rumors (Alkhodair, Ding, Fung, & Liu, 2020; Liang, Yang, & Xu, 2016). These rumor detection methods based on traditional machine learning depend to a large extent on the features extracted and selected manually, which consumes tremendous manpower and time. The robustness of the feature vector obtained is not robust enough, which is difficult to deal with complex and changeable scenarios.

Motivated by the success of deep learning, many research utilize various neural networks to detect rumors (Bian et al., 2020; Ma, Gao, & Wong, 2019; Yang et al., 2020). The deep learning methods can obtain better and more essential representative features than feature engineering methods, so as to achieve better classification results. From a more practical point of view, researchers are not limited to the pursuit of rumor detection accuracy, but also hope that they can be detected earlier (Liu, Jin, & Shen, 2019).

However, to effectively achieve precise detection of rumors, there are several issues that need to be solved effectively. Firstly, the natural end-to-end structure of neural networks renders the difficulty to grasp the key components of the rumor information, resulting in that training lacks controllability and efficiency. Therefore, it is necessary to extract key information and optimize the network structure. Secondly, we notice that most of the studies only consider a single perspective of rumor detection such as accuracy, while ignoring that the time spent on detection is equally important in practice. Although some studies have made efforts in the early detection of rumors, they have sacrificed the partly stability and accuracy of predictions. Therefore, how to balance and compromise between different goals and achieve the best performance is a significant challenge. Thirdly, with the deepening of the research, various models have been proposed but become more and more complex and their computational consumption is also increasing. Therefore, whether an efficient method can be found in the complex network model to reduce the computational consumption in training is also a problem worthy of further research. For the existence of hyperparameters in the model, the common practice is to try different values through enumeration, and then choose one parameter that can be used to get the best result from the limited choices, which cannot guarantee the optimality.

In this paper, we investigate a novel rumor detection method in OSNs. To grasp the key information and optimize the network structure, an attention mechanism is introduced. From different perspectives of detection performance, we put forward three optimization objectives. To solve the problem of hyperparameters caused by the above optimization, a more concise determination method is introduced. The noteworthy contributions of this paper are summarized as follows.

- We focus on textual rumors in online social networks and propose a novel rumor detection method. An efficient spatial attention mechanism is introduced to extract the intrinsic characteristics of the text content. An effective framework is constructed to extract the evolutionary characteristics of the rumor in the propagation process, making it possible to characterize diffusion law of rumors on OSNs.

- We treat the detection time, accuracy and stability as the three training objectives, and continuously adjust and optimize them instead of using a fixed value during the entire training process, thereby enhancing its adaptability and universality.
- To improve the efficiency, we design a sliding interval-based detection method by constructing different optimization loss objectives to intercept the required data rather than using the entire sequence data.
- To solve the problem of hyperparameter selection brought by integration of multiple optimization objectives, a convex optimization method is utilized to parameterize them so that they can adaptively change during the entire model learning process and avoid the huge computational cost of enumerations.

The remainder of this paper is organized as follows. We review the related works in Section 2. In Section 3, we present the problem formulation. In Section 4, the proposed method and implementation framework for rumor detection are presented in detail. In Section 5, experimental results and performance evaluation are conducted. Finally, we summarize this paper in Section 6.

2. Related works

With the rapid development of OSNs, rumor detection has received more and more attentions. The task of rumor detection is to distinguish whether an information in OSNs is a rumor through some related information and various computational methods have been proposed, mainly including four types: crowd sensing related methods, feature engineering related methods, propagation mode related methods and deep learning methods.

2.1. Crowd sensing related methods

The crowd sensing related method is the mainstream rumor detection method of the current social network platform. The platform delivers suspicious information reported by users to experienced editors or industry experts, then they use their knowledge and experience to refute rumors with scientific explanation.

Mohler and Brantingham (2018) proposed a crowd-sourced framework based on the novel online Hawkes process estimation algorithm, using crowd-sourced information such as reports, tips and neighbor posts to construct a prediction model, which provides convenience for collecting rumors. Considering the problem of dynamic participant selection with heterogeneous tasks, Li et al. (2018) minimized the cost of sensing while maintaining a certain level of probability coverage, thereby providing a solution to the problem of fewer users in rumor collection. To encourage users to participate in rumor-reporting, Xu et al. (2019) designed the incentive mechanism of a crowd awareness system with multiple collaborative tasks to minimize the social cost, so that each collaborative task can be performed by a compatible set of users.

Although the accuracy of crowd sensing related method is very high, it causes time-lag because it requires users to take the initiative to report. Moreover, numerous data are generated in OSNs every day, it is impossible for human beings to process all the data, which may omit important rumors.

2.2. Feature engineering related methods

Feature engineering related method selects and extracts the features that can represent the data effectively from the training dataset and utilizes the features to train to obtain the classification model.

Liang et al. (2016) found that the behavior of rumormongers is different from that of ordinary users, then proposed a rumor detection method based on user behavior characteristics and analyzed the differences between different contents and types. Guo, Cao, Zhang,

Guo, and Li (2018) analyzed leveraging hierarchical representations at different levels and the social contexts, and proposed a two-layer neural network in which important semantic information (such as account and text features) is introduced. Based on the comprehensive consideration of semantic features such as topics and emotions, and the structural features of information dissemination, Wu, Yang, and Zhu (2015) proposed a hybrid support vector machine classifier based on a graphics kernel.

However, feature engineering related method relies on manual feature selection, and it is difficult to obtain high-dimensional, complex and abstract feature data. Therefore, the robustness of the feature vector obtained is poor, and it is difficult to comprehensively and systematically summarize the features of the rumor.

2.3. Propagation mode related methods

The feature that only extracts a single piece of information often ignores the connection between rumors, but propagation mode related method can reflect the potential connection between rumors through its hierarchical structure.

Jin, Cao, Jiang, and Zhang (2014) utilized a three-layer reputation network composed of events, sub-events and messages to represent the occurrence of an event, and established connections through semantics and social relationships, and then proposed a hierarchical propagation model. Wan et al. (2021) proposed a rumor diffusion model by exploring the coupling relationship between rumors and anti-rumors, and then predicted the spread of rumors and proposed the corresponding intervention measures. Ma, Gao, and Wong (2017) proposed a kernel-based propagation tree method to identify rumors by evaluating the similarity between propagation trees, which finds and captures the salient substructures in the propagation tree of Ru-MORS.

Propagation mode related method is one of the hotspots of research, but the diffusion of rumors is affected by many factors. At present, the consistent structure of rumors transmission has not been well explored, and further research is still needed.

2.4. Deep learning methods

Motivated by the success of deep learning, many research utilize various neural networks to detect rumors. Yu et al. (2017) proposed a convolutional neural network-based rumor detection method (CAMI), which extracts scattered key features and forms high-level interactions between them. Yuan, Ma, Zhou, Han, and Hu (2019) explored local semantic relations and global structural information, and proposed a global and local attention-based network (GLAN) that jointly encodes local semantics and global structural information for rumor detection. Song et al. (2021) put forward the concept of a “credible detection point” and started detection through to realize the early detection of rumors.

We noticed that the key point of most research only focuses on the accuracy of detection at present. Although some studies (Song et al., 2021; Yuan et al., 2019) have begun to focus on early detection, their approaches are usually to set a fixed threshold. However, the content of rumors is varied and diverse, the fixed threshold may not be able to meet the needs of its detection. Furthermore, as the network structure becomes more and more complex, its calculation complexity is getting higher and higher. When hyperparameters arise in an experiment, it is common approach to enumerate, keep trying different values, and then choose a parameter that works best from a limited selection. However, this method can only find a relatively suitable hyperparameter and cannot guarantee optimality. To solve these problems, we treat the detection point as a training objective, and continuously adjust and optimize it instead of using a fixed value during the training process, thereby enhancing its universality. At the same time, accuracy and stability are regarded as the other two training objectives, and the detection of rumors can be realized as early as possible while improving the accuracy. In order to decrease the amount of calculation in the training process, we use a sliding interval to intercept the required data instead of using the entire sequence data.

3. Problem statement

We comprehensively consider two aspects: the content of the text and the characteristics of reposting. To state clearly, we present the following definitions.

Definition 1. We use a sequence $\mathbb{M} = \{m_1, m_2, \dots, m_n\}$ to represent the set of source microblogs, where n is the number of source microblogs and the source microblog m_i represents the text content.

Definition 2. Each source microblog m_i has a relevant repost sequence and a corresponding set of time denoted as $\mathbb{R}_i = (r_i, t_i)$. The repost microblogs $r_i = \{r_{i1}, r_{i2}, \dots, r_{iv_i}\}$ represent the text content, $t_i = \{t_{i1}, t_{i2}, \dots, t_{iv_i}\}$ represents the timestamp when content is published, v_i represents the length of the repost sequence of the source microblog m_i .

The purpose of rumor detection task is to train a model $f : \mathbb{M} \Rightarrow P(y = 1 | \mathbb{R}_i, \theta) \in (0, 1)$ that can predict accurately whether the source microblog is a rumor or not. Among them, y is class label and θ is all parameters of the model, and we set $y = 1$ for rumor and $y = 0$ otherwise.

4. Methodology

To solve the problem defined in Section 3, we propose an interval detection method based on multi-objective loss (IDMO) which extracts hidden features of microblog contents and reposts sequence, then completes the detection based on the hidden features in a small time interval. The overall framework of IDMO mainly includes the following functional modules:

- **Data preprocessing:** This module contains text segmentation and word embedding, which divides a piece of text content into different words, then filters out important words and converts them into the form of vectors that can be recognized by CNN model.
- **CNN model based on spatial attention:** The spatial attention mechanism overcomes the lack of memory by intuitively giving the contribution of each word to the results from different dimensions. This module can utilize text contents to extract hidden features. The main operations include convolution, pooling and fully-connected layer.
- **GRU model with sequence feature:** This module can utilize repost sequences to extract long-distance characteristic features, where the main operations include reset and update.
- **Multi-objective loss function:** This module evaluates the training results from the perspective of multiple loss functions, and adjusts the parameters of the model accordingly through back-propagation.
- **Parameterization of hyperparameters:** It is difficult to determine the hyperparameters carried by multiple loss functions. To avoid the huge amount of calculation brought by the enumeration method, this module converts it into an optimization problem, and through theoretical analysis, we derive the optimal solution that guarantees the Pareto optimum.

4.1. Data preprocessing

A word is a basic element of language that carries an objective or practical meaning, which can be used on its own. Therefore, cutting the text into words, reducing the coupling between them and characterizing them as input can make the semantic analysis more accurate. In recent years, Jieba segmentation package (Lai et al., 2019) is widely used by many researchers because of its simplicity and efficiency. In this paper, Jieba segmentation is utilized to segment text data and generate a generalized word cloud based on the score of each word

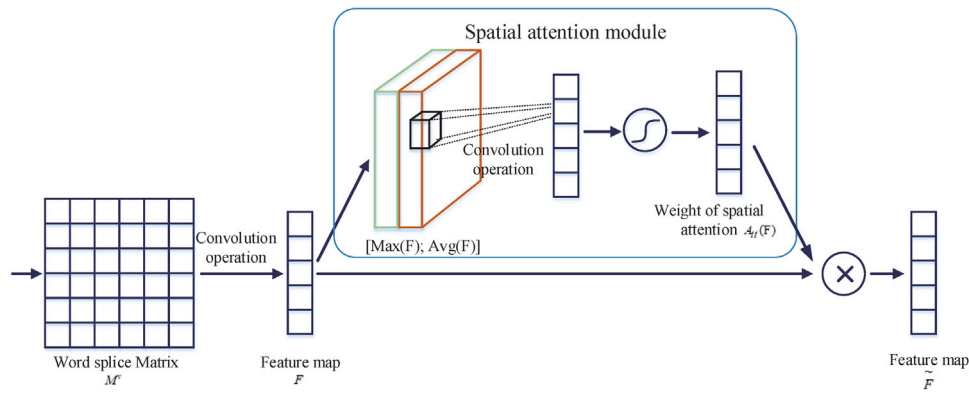


Fig. 1. An illustration of the spatial attention mechanism. After process of segmentation and word embedding, the word splice matrix is constructed first. Then, the feature map F is extracted after convolution operation and processed by the spatial attention module to obtain the final feature map \tilde{F} .

in the text. This approach is reasonable because this method can filter out common words, retain important words, and avoid meaningless words being input into the model to interfere with the final result. The way (He, Chang, Lim, & Banerjee, 2010) to calculate the score is as follows,

$$TF_{w,D_i} = \frac{count(w)}{|D_i|}, \quad (1)$$

$$IDF_{w,D_i} = \log \frac{N}{1 + N(w)}, \quad (2)$$

$$TF-IDF_{w,D_i} = TF_{w,D_i} \times IDF_{w,D_i}, \quad (3)$$

where w is the keyword, $count(w)$ is the number of w 's occurrences, $|D_i|$ is the number of all words in the D_i document, $N(w)$ is the number of documents in which the word w appears in the corpus, and N is the total number of documents in the corpus, respectively.

Then, we need to transform the segmented words into a feature vector which can be recognized by neural networks, so the efficient word embedding named word2vec (Ji, Satish, Li, & Dubey, 2019) is used. Word2vec model can take text contents as the input and produces real-valued low-dimensional vector representations for the words that appear in that contents. Let a sequence of words is $W = \{w_1, w_2, \dots, w_i, \dots, w_{n_w}\}$ and make a set of words around w_i within the specified window size z to be the context of w_i . By maximizing the average log-likelihood conditional probability function between w_i and its context words together to learn their word representation as follows,

$$\frac{1}{n_w} \sum_{i=1}^{n_w} \sum_{-z \leq j \leq z} \log p(w_j | w_i). \quad (4)$$

4.2. CNN model based on spatial attention

The convolutional neural network model is a type of feedforward neural network that includes convolution calculation and a deep structure, which has achieved great success in the feature extraction of static data such as images (Husain & Bober, 2019). However, due to the lack of memory of historical words with CNN model in the process of natural language, the weight of historical words will be ignored, and the location information of important words will be lost, resulting in low accuracy of feature extraction.

To solve this problem, we introduce the word-level spatial attention mechanism to extend the CNN model. By considering the importance of words in sentences, extracting more refinement features of the same data from different dimensions, thereby more information can be obtained to improve the memory of historical words in the CNN model. The contribution of each word to the target feature can be obtained from this mechanism, so the proposed IDMO has a certain interpretability.

Through text segmentation and word embedding, we can obtain a series of d -dimensional word vectors v_i for each text content (including the source microblog and reposting microblogs). Then the word splice matrix M^v can be constructed as follows,

$$M^v = v_1 \oplus v_2 \oplus \dots \oplus v_{n_v}, \quad (5)$$

where \oplus is the concatenation operation, $M^v \in R^{d \times n_v}$, where n_v is the number of word vectors.

The CNN model includes the convolutional layer, the pooling layer and the fully-connected layer. In the convolutional layer, we utilize the convolution kernel $K_i \in R^{d \times d_k}$ to scan M^v and the activation function ReLU (Glorot, Bordes, & Bengio, 2011) is applied to obtain the feature map as follows,

$$c_i = ReLU((W_c, M_{i:i+d_k-1}^v) + b_i), \quad (6)$$

where W_c is the corresponding weight matrix and b_i is bias. Then, in the pooling layer, the downsampling operation is applied to extract the most obvious features,

$$\tilde{c}_i = \max\{c_i, c_{i+1}, \dots, c_{i+d_k-1}\}. \quad (7)$$

In the fully-connected layer, we recombine the final feature map \tilde{C} with ReLU to express the total features F of text content as follows,

$$F = ReLU(W_f \tilde{C} + b). \quad (8)$$

Then, we conduct more detailed processing on the source microblog content from different dimensions. The word-level spatial attention mechanism is introduced as shown in Fig. 1, which utilizes the inter-spatial relationship. Through the two operations of maximize pooling and average pooling, we further refine the features extracted by CNN. We concatenate and calculate them by a convolution operation as follows,

$$A_t(F) = \sigma(K_2[\max(F), \text{avg}(F)]), \quad (9)$$

where σ is the sigmoid function and K_2 is the convolution kernel. The existing study (Woo, Park, Lee, & So Kweon, 2018) showed that the spatial attention mechanism has the best performance when the size of the convolution kernel is 7×7 .

Finally, the spatial attention process can be summarized as follows,

$$\tilde{F} = A_t(F) \otimes F, \quad (10)$$

where \otimes is the Hadamard product and \tilde{F} denotes the final feature map output by CNN model with spatial attention.

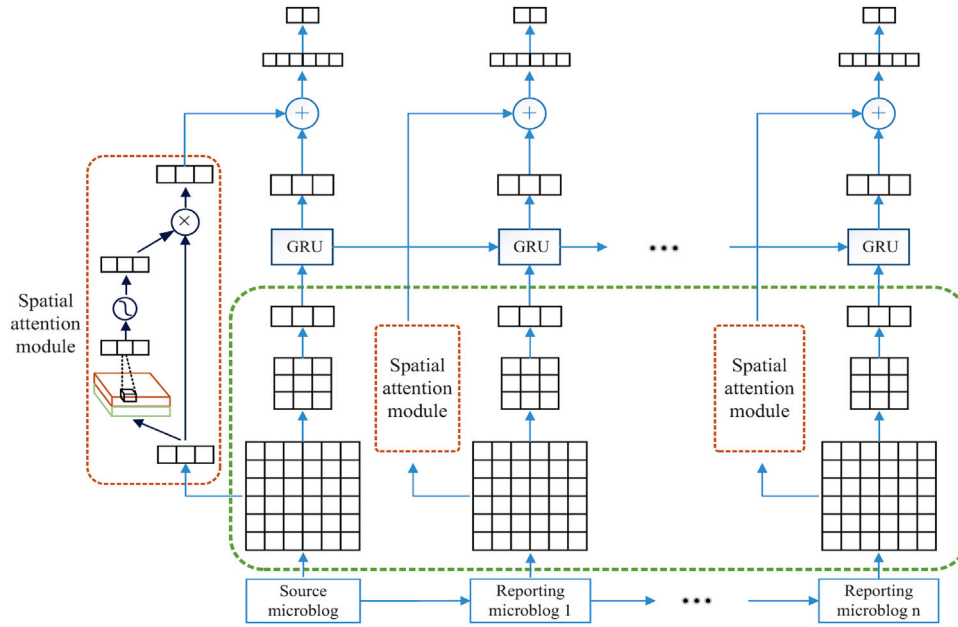


Fig. 2. An illustration of the structural framework. The red dotted box represents the spatial attention module and the green dotted box represents convolution operation.

4.3. GRU model with sequence feature

Recurrent neural network is a neural network for processing sequence data which has achieved great success in the field of natural language processing (NLP). Compared with other neural networks, it can process sequential data with time-dependent, which is particularly suitable for processing the reposting sequence of microblogs, so as to extract the characteristics of forwarding in OSNs. Gate Recurrent Unit (GRU) is a special RNN, mainly to solve the problem of gradient disappearance and gradient explosion in the training process of long sequences, which can perform better in longer sequences than the ordinary RNNs.

Through CNN, the features of each microblog in the reposting sequence have been extracted and we feed them to GRU for training. GRU unit includes reset gate (r), and update gate (z) as follows,

$$z = \sigma(W_z[H(t-1), x(t)]), \quad (11)$$

$$r = \sigma(W_r[H(t-1), x(t)]), \quad (12)$$

$$H'(t-1) = H(t-1) \otimes r, \quad (13)$$

$$H'(t) = \tanh(W_H[H'(t-1), x(t)]), \quad (14)$$

$$H(t) = (1-z) \otimes H'(t-1) + z \otimes H'(t), \quad (15)$$

where \otimes is the Hadamard product, $H(t)$ is the output of hidden layer, W_z , W_r and W_H are the weight matrices.

By integrating the above modules, the structural framework of IDMO as shown in Fig. 2. After process of segmentation and word embedding, the word splice matrix is constructed first. Then, the feature map F is extracted from word concatenation matrix by convolution operation and processed by the spatial attention module to obtain the feature map \tilde{F} . After that, the feature map F is fed into GRU module, and the output is obtained. Further, the feature map \tilde{F} and $H(t)$ are spliced, we put it into the classifier to get the final prediction result. Focal loss \mathcal{FL} is a method proposed by Lin, Goyal, Girshick, He, and Dollár (2020) to solve the imbalance of positive and negative sample ratio in one-stage target detection, which reduces the weight of a large number of simple negative samples in training, we introduce it as the following loss function,

$$\mathcal{FL}(P(i|\theta)) = -(1 - P(i|\theta))^\gamma \log(P(i|\theta)), \quad (16)$$

where $P(1|\theta) = \sigma(H, s)$ and $P(0|\theta) = 1 - P(1|\theta)$, s is the weight vector of the corresponding hidden layer H .

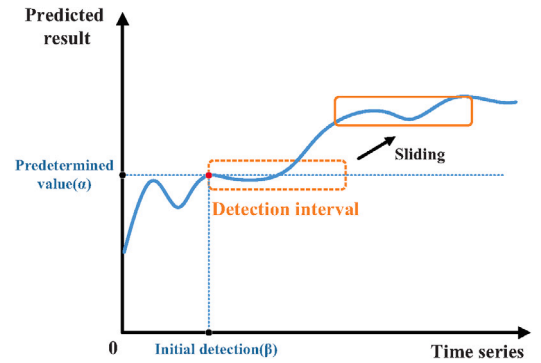


Fig. 3. An illustration of the detection interval and detection point. The blue line represents the predicted results, the red point represents the initial detection point and the orange dashed box indicates the detection interval.

4.4. Multi-objective loss function

To identify rumors as early as possible according to the source microblog in the process of rumor forwarding, we need to find a initial detection point for each source microblog. As illustrated in Fig. 3, before the detection point, the result of identifying curve fluctuates frequently, which means that the rumors are difficult to distinguish during this period. After this point, the detection result is relatively stable and tends to be accurate. Once the initial detection point is determined, we will use it with a fixed-length interval to start the detection process. This detection interval method effectively saves computing resources without employing the predictive results of the entire sequence in Song et al. (2021).

It is worth noting that due to the differences in the content and forwarding sequence of different source microblogs, the corresponding detection intervals cannot be completely the same. Therefore, the method of artificially given detection thresholds (Song et al., 2021) is no longer applicable. In this section, we propose an adaptive interval detection method to solve this problem.

Firstly, we introduce two parameters $\alpha \in [0, 1]$ and $\beta \geq 0$ for each source microblog to determine the location of the initial detection point, in which α is a predetermined threshold and β is the time point

when the predictive result reaches the threshold. When the threshold is reached for the first time by the predictive result $P(y|\mathbb{R}_t, \theta) > \alpha$, the detection process is started and the corresponding time point is β .

Our purpose is to train a model that can accurately, quickly and stably identify rumors through the adaptively adjusting the detection interval. We make adjustments from two aspects: inside the detection interval and outside the detection interval. Inside the detection interval, the first part of our objective aims to achieve the accuracy of prediction, so it is necessary to maximize the predictive result in the detection interval as follows,

$$L_1(\theta) = \sum_{\beta \leq i \leq \beta + len} FL(P(i)), \quad (17)$$

where len is the length of the detection interval.

To ensure the stability of the final predictive results and avoid excessive volatility of the predictive trend, the second objective is to minimize the difference in predictive results within the detection interval. Considering the convenience of derivation during back propagation, we use smooth Gaussian radial basis function as follows,

$$L_2(\theta) = \sum_{\beta \leq i \leq \beta + len} e^{(\varepsilon r)^2}, \quad (18)$$

where ε is the Gaussian coefficient, $r = \|P(i|\theta) - \bar{P}\|_2$ refers to the L-2 norm of the predictive result minus the average and $\bar{P} = \frac{1}{len} \sum_{\beta \leq i \leq \beta + len} P(i|\theta)$.

To achieve effective identification as early as possible in the diffusion of rumors, we introduce time constraint into the third objective and use the form of log-likelihood to reduce the computational complexity of derivation as follows,

$$L_3(\theta) = -\log \frac{\beta}{F}, \quad (19)$$

The initial value of β is determined by α , and then β will be continuously updated according to the screening of the detection interval. F is the length of the entire time series.

To take into account the above three objectives in the training process, we integrate them through three weight parameters λ_i ($i = 1, 2, 3$) as follows,

$$Loss(\theta) = \sum_i \lambda_i L_i(\theta). \quad (20)$$

Since these weight parameters need to be set prior to the learning process, rather than obtained through training, they are also called hyperparameters. Specific method of determining them will be described in detail in the next section.

Outside the detection interval, through continuously sliding the detection interval, the model is trained to adaptively find the most suitable time point β_{new} for detection as follows,

$$\beta_{new} = \min_{\beta} \left\{ \sum_i \lambda_i L_i(\theta) \right\}. \quad (21)$$

4.5. Parameterization of hyperparameters

In the previous section, we aggregate three loss objectives into a loss function by introducing three weight parameters. Usually, these weight parameters are hyperparameters that are set before starting the learning process, not the values obtained through training. Therefore, different given weight parameters will greatly affect the training process, which in turn affects the final predictive result. The way to deal with hyperparameter is through enumeration, constantly trying different values, and then selecting a parameter that can achieve the best result from a limited choice. However, this approach can only find a relatively suitable hyperparameter, which cannot guarantee the optimality. Moreover, as the number of hyperparameters increases, the computational consumption for its selection increases exponentially.

In this section, we introduce these hyperparameters into the training process, so that they can adaptively change during the entire model

learning process like non-hyperparameters. The method of learning to rank (LTR) is used to deal with the weight parameters introduced in (20). The proposed method theoretically guarantees that these weight parameters are Pareto optimal, which is concise and easy to calculate.

To ensure that all three loss objectives can play the role in the learning stage of the model, we add the boundary constraints to its hyperparameters and the optimal problem can be formulated as follows,

$$\begin{cases} Loss(\lambda_1^*, \lambda_2^*, \lambda_3^*|\theta) = \min_{\lambda_1, \lambda_2, \lambda_3} \sum_i \lambda_i L_i(\theta), \\ \text{s. t. } \sum_i \lambda_i = 1 \text{ and } \lambda_i > 0, \forall i \in \{1, 2, 3\}. \end{cases} \quad (22)$$

The solution pair that satisfies the KKT (Karush–Kuhn–Tucker) conditions is called Pareto stationary as follows,

Theorem 1 (Karush–Kuhn–Tucker Condition). *The solution pair $(\lambda_1^*, \lambda_2^*, \lambda_3^*)$ is the optimal solution of problem (22) if there exist multipliers $\lambda_i > 0, i = 1, 2, 3$, such that*

$$\begin{cases} \sum_{i=1}^3 \lambda_i \nabla_{\theta} L_i(\theta) = 0, \\ \text{s.t. } \sum_i \lambda_i = 1. \end{cases} \quad (23)$$

The detailed proof can be easily found in Ruszczyński (2011). The conditions can be transformed into the following quadratic optimization problem,

$$\begin{cases} \min_{\lambda_1, \lambda_2, \lambda_3} \left\| \sum_{i=1}^3 \lambda_i \nabla_{\theta} L_i(\theta) \right\|_2^2, \\ \text{s. t. } \sum_i \lambda_i = 1 \text{ and } \lambda_i > 0, \forall i \in \{1, 2, 3\}. \end{cases} \quad (24)$$

Sener and Koltun (2018) have proved the solution that satisfies the condition will minimize the loss function along the gradient. Then, we first deal with the equality constraints and rewrite the above problem in the form of vector to derive the optimal solution as follows,

$$\begin{cases} \min_{\lambda} \frac{1}{2} \lambda^T G G^T \lambda \\ \text{s. t. } e^T \lambda - 1 = 0 \text{ and } \lambda_i > 0, \forall i \in \{1, 2, 3\}. \end{cases} \quad (25)$$

where λ is the concatenated vector of λ_i , G is the stacking matrix of gradient $\nabla_{\theta} L_i(\theta)$, and e is the 3-dimensional column vector with element of 1. Next, we construct the Lagrangian \mathbb{L} as follows,

$$\mathbb{L}(\lambda, \mu, g, H) = \frac{1}{2} \lambda^T G G^T \lambda + \mu(e^T \lambda - 1) + (H - \lambda)^T g \quad (26)$$

where μ is the Lagrange multiplier of equality constraint and $g = (g_1, g_2, g_3)^T$ is the Lagrange multiplier vector of inequality constraint, where $g_i \geq 0, \forall i \in \{1, 2, 3\}$. $H = (h_1^2, h_2^2, h_3^2)^T$ is the vector of slack variable, the purpose of introducing h_i^2 is to change the inequality constraint into an equality constraint.

The solution of the problem is given as follows,

$$\nabla_{\lambda} \mathbb{L}(\lambda, \mu, g, H) = 0, \quad (27)$$

$$\nabla_{\mu} \mathbb{L}(\lambda, \mu, g, H) = 0, \quad (28)$$

$$\nabla_g \mathbb{L}(\lambda, \mu, g, H) = 0, \quad (29)$$

$$\nabla_H \mathbb{L}(\lambda, \mu, g, H) = 0. \quad (30)$$

And we can derive the solution by solving the following linear system,

$$\begin{bmatrix} G G^T & e & -e & 0 \\ e^T & 0 & 0 & 0 \\ -e & 0 & 0 & e \\ 0 & 0 & e & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ g \\ H \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 1 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}. \quad (31)$$

According to the Moore–Penrose inverse (Wan, Wang, Han, & Wu, 2019), we have

$$\begin{bmatrix} \lambda \\ \mu \\ g \\ H \end{bmatrix} = (QQ^T)^{-1}Qz. \quad (32)$$

where $Q = \begin{bmatrix} GG^T & e & -e & 0 \\ e^T & 0 & 0 & 0 \\ -e & 0 & 0 & e \\ 0 & 0 & e & 0 \end{bmatrix}$ and $z = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$. Finally, the solution pair of the hyperparameters can be derived as follows,

$$\lambda = (QQ^T)^{-1}Qz[1 : 3]. \quad (33)$$

In this paper, we adopt the Adam algorithm (Kingma & Ba, 2015) to optimize loss function. Although Adam with stochastic gradient in batches, our proposed method still provides a theoretical guarantee of convergence as gradient descent (Lin et al., 2019), which specific procedure is illustrated in Algorithm 1.

Algorithm 1: Multi-objective detection interval

Input:

- The loss objectives $L_1(\theta)$, $L_2(\theta)$ and $L_3(\theta)$;
- Pre-determined value α for the initial detection;
- Initial value of weight parameters λ_1, λ_2 and λ_3 ;

Output:

- The updated parameters of the model θ ;
 - 1: Locate the initial detection point β according to α ;
 - 2: Integrate three loss objectives as $Loss(\theta)$;
 - 3: **for** each batch **do**
 - 4: Update the parameters θ of the model by adopting Adam to optimize $Loss(\theta)$;
 - 5: Update the weight parameters λ_1, λ_2 and λ_3 according to Eq. (33);
 - 6: Update the detection point β according to Eq. (21);
 - 7: **end for**
 - 8: **return** θ ;
-

5. Experiments

In this section, the interval detection method based on multi-objective loss is evaluated using numerical experiments. Firstly, we verify the effectiveness of IDMO with real-world datasets. Secondly, we conduct an indepth discussion based on experimental results. Finally, we analyze the sensitivity of the parameters that influence the experimental results.

5.1. Datasets

To evaluate the performance of the proposed methods, we utilize three datasets about COVID-19 in the experiments which are extracted from the real world large scale social network such as Weibo and Twitter and Snopes.com. Weibo dataset (Song et al., 2021) is published by Tsinghua University, which contains the rumor content, non-rumor content and its reposting record. Twitter dataset (Elhadad, Li, & Gebali, 2020) and Snopes dataset (Hanselowski, Stab, Schulz, Li, & Gurevych, 2019) are released under the MIT license, which collects information about COVID-19, including rumors, non-rumors and unverified content from Twitter and Snopes.com, respectively. These three datasets include Chinese language and English language, news media and self-media, which comprehensively simulate application scenarios of rumor detection. Details of the datasets description are listed in Table 1.

Table 1
Datasets description.

Dataset	Weibo	Twitter	Snopes
Type	self-media	self-media	news media
Language	Chinese	English	English
Number of rumor	3851	2540	4208
Number of non-rumor	4199	1040	1660
Number of unverified	–	125	429
Minimum reposting length	3	4	2
Maximum reposting length	59,317	32,315	10,751
Average reposting length	624	482	204

In the process of model training and testing, we employ the holdout verification method (Pang et al., 2019) to alleviate the problem of over-fitting, which divides the datasets into training set, verification set and test set randomly according to the proportion of 80%, 10% and 10%. The training set is utilized to train the parameters of the model, the verification set is utilized to make a preliminary assessment of the ability, and the test set is utilized to evaluate the generalization ability of the model.

5.2. Baselines and evaluation metrics

To fairly compare the performance of different methods, three evaluation metrics (Yuan et al., 2019) are adopted in this section: the accuracy, precision and recall. Moreover, to evaluate the performance of the proposed method in prediction time and predict stability, we introduce the “Early Rate (ER)” from Song et al. (2021) and propose the Measure of Stability (MS) as follows,

$$ER = \frac{1}{|Test|} \sum_{i \in Test} \frac{t_i}{F_i}, \quad (34)$$

$$MS = \frac{1}{|Test|} \sum_{i \in Test} \sum_{j=i}^{t_i+length} \|P_i(j) - \bar{P}_i\|_2, \quad (35)$$

where t_i refers to the time node when the prediction result reaches a fixed value (we set the value to 0.875 in the experiment) for the first time, F_i is the length of reposting sequence, $Test$ is the test set, $length$ refers to the length of the interval that we want to detect stability after t_i and \bar{P}_i is the average of the prediction results in its interval.

Then, we compare our method with a series of representative baselines as follows,

- **DSTS** (Ma, Gao, Wei, Lu, & Wong, 2015): An SVM with dynamic time series structure model which can capture the changes of various social context characteristics over time.
- **CAMI** (Yu et al., 2017): To form the interaction between important features, CAMI utilizes CNN to extract key features scattered in the input sequence, so as to effectively identify rumors.
- **GLAN** (Yuan et al., 2019): A rumor detection method with global-local attention network, which combines local semantic information with global structural information to encode.
- **CED** (Song et al., 2021): An early detection method with CNN+RNN, which proposes “credible detection point” and a multi-objective loss function.
- **IDMO**: We propose the interval detection method of rumor based on multi-objective loss, which does not use the spatial attention mechanism when dealing with text.
- **IDMO-SA**: The interval detection method of rumor based on multi-objective loss with the spatial attention mechanism.

5.3. Parameters optimization

We repeat the experiment to provide the optimal hyperparameters for each baseline method. During the training process, we set the initial dropout rate to 0.5 and search in the range of 0.05. The experimental

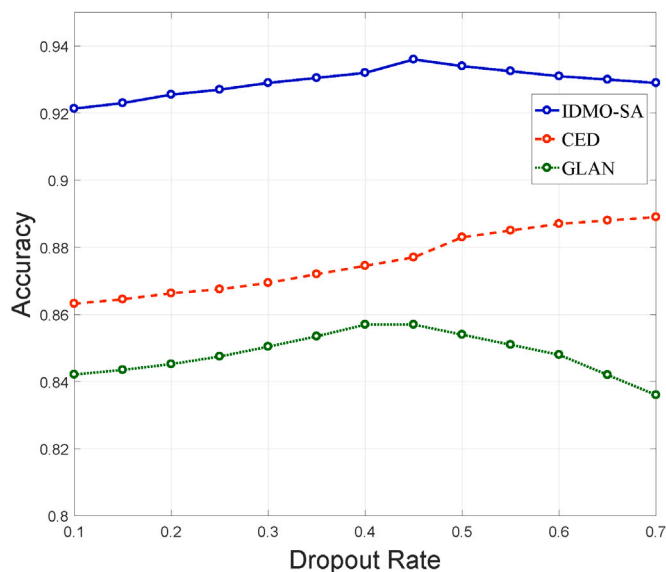


Fig. 4. Dropout rate optimization process in Weibo dataset.

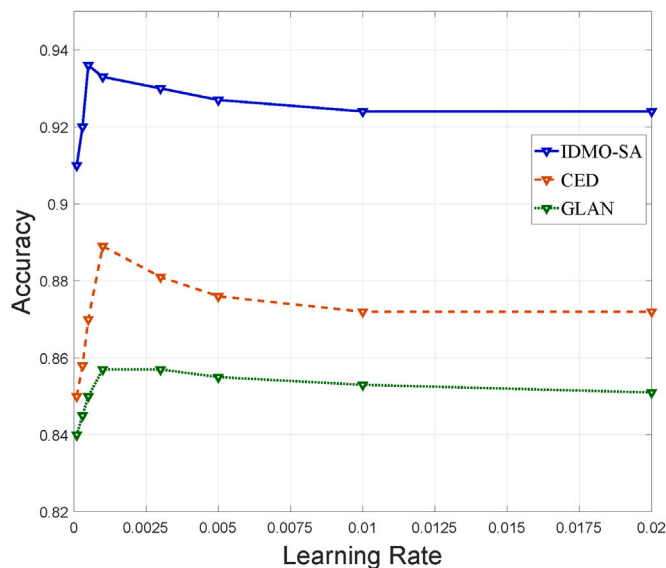


Fig. 5. Learning rate optimization process in Weibo dataset.

results are shown in Fig. 4, the performance of the methods are the best when the dropout rate of IDMO-SA is 0.45, CED is 0.7 and GLAN is 0.4. Learning rate is explored from the following range of {0.0001, 0.0005, 0.001, 0.005, 0.01, 0.1}, the corresponding experimental results are illustrated in Fig. 5. We notice that the accuracy rate reaches the highest when the learning rate of IDMO-SA is 0.0005. When the learning rate of CED and GLAN is 0.001, the methods have the best effect. Regularization coefficient starting from a very small value ($10e-6$), and is searched with 10 times increase. The experimental results will increase with the increase of the regularization coefficient. When the optimal result is reached, the experimental results will decrease with the increase of the regularization coefficient and tend to be stable. Fig. 6 summarizes the accuracy of IDMO-SA, CED and GLAN when the regularization coefficient in the range of [$10e-6$, $10e-3$], where the corresponding optimal coefficient values are $6e-5$, $1e-4$ and $5e-5$, respectively.

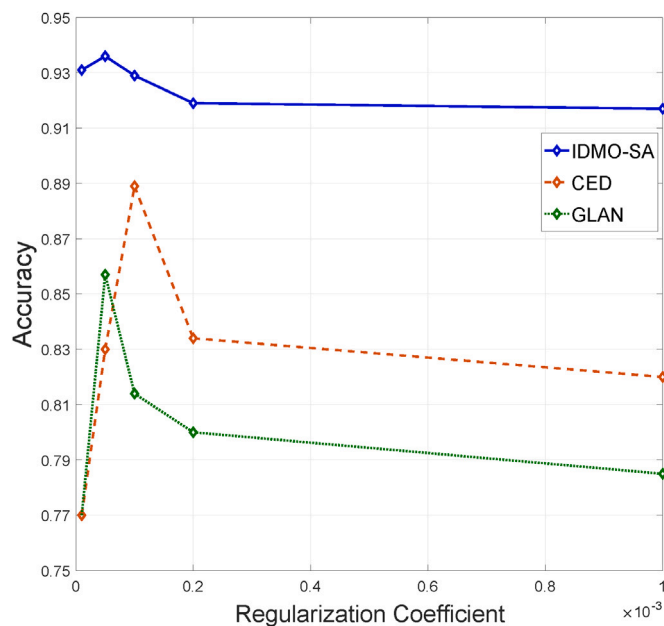


Fig. 6. Regularization coefficient optimization process in Weibo dataset.

Table 2
The experimental results of Weibo dataset.

Methods	Metrics				
	Accuracy	Precision	Recall	ER	MS
DSTS	0.711	0.738	0.731	N/A	N/A
CAMI	0.786	0.835	0.832	N/A	N/A
GLAN	0.857	0.912	0.871	N/A	N/A
CED	0.889	0.876	0.919	0.327	0.641
IDMO	0.910	0.891	0.914	0.544	0.334
IDMO-SA	0.936	0.913	0.946	0.376	0.187

5.4. Performance evaluation

In this subsection, we provide detailed comparison results in Tables 2–4, which corresponds to Weibo dataset, Twitter dataset and Snopes dataset, respectively. Note that, Snopes dataset is the dataset of online news media, which does not have the reposting function of traditional social networks. But the news media are also rife with rumors and cannot be ignored. Therefore, we rank their comments by the time they were published and treat the comment sequence as a reposting sequence. Considering that the DSTS, CAMI and GLAN cannot obtain the data required by ER and MS, the proposed method is only compared with the CED method. We bold all the best results of each evaluation metric in the tables.

Table 2 illustrates the detailed experimental results of Weibo dataset, in which the proposed IDMO-SA method achieves the highest accuracy rate of 93.6%. In terms of precision and recall, the performance of IDMO-SA is obviously better than that of baselines, reaching 91.3% and 94.6% respectively. The early rate of IDMO-SA reaches 37.6%, which higher than that of CDE (32.7%). However, the stability of IDMO-SA reaches 18.7%, which is significantly lower than that of CDE (64.1%). This is because CED ignores partly stability of the experimental results when pursuing earlier detection, which may cause errors in the detection results. Therefore, we can see that it is worthwhile to sacrifice 14.9% of the detection time in exchange for 2.5 times the stability.

Table 3 illustrates the detailed experimental results of Twitter dataset. Unlike the Weibo dataset, Twitter introduces unverified information, which increases the difficulty of learning to a certain extent. Since the amount of data contained in Twitter is lower than that of

Table 3
The experimental results of Twitter dataset.

Methods	Metric				
	Accuracy	Precision	Recall	ER	MS
DSTS	0.552	0.578	0.860	N/A	N/A
CAMI	0.691	0.735	0.532	N/A	N/A
GLAN	0.741	0.612	0.731	N/A	N/A
CED	0.773	0.696	0.849	0.418	0.610
IDMO	0.765	0.731	0.813	0.561	0.494
IDMO-SA	0.818	0.749	0.896	0.499	0.374

Table 4
The experimental results of Snopes dataset.

Methods	Metric				
	Accuracy	Precision	Recall	ER	MS
DSTS	0.461	0.563	0.710	N/A	N/A
CAMI	0.677	0.615	0.653	N/A	N/A
GLAN	0.589	0.702	0.731	N/A	N/A
CED	0.732	0.660	0.769	0.436	0.729
IDMO	0.699	0.671	0.734	0.516	0.427
IDMO-SA	0.801	0.714	0.790	0.437	0.371

the Weibo dataset, which may cause insufficient learning. Therefore, the values of various metrics in Table 3 are relatively lower than those in Table 2. Nonetheless, the performance of IDMO-SA are still significantly improved compared to baselines. In terms of accuracy, precision and recall, IDMO-SA gets the best results in the experiment, reaching 81.8%, 74.9% and 89.6% respectively. The early rate of IDMO-SA has reached 49.9%, which is 19.3% lower than that of CED. In terms of stability, IDMO-SA reaches 37.4% which increases 63.1% that of CDE.

Table 4 illustrates the detailed experimental results of Snopes dataset, which also contains unverified information and less amount of data. Unlike Twitter that each tweet has a limit on the number of words, news media usually publish content with more text. This is very conducive to the training of the attention mechanism. Therefore, the accuracy of IDMO is increased by about 10% compared to baselines, which far exceeds the performance on Weibo and Twitter datasets. In terms of early rate, IDMO-SA (43.7%) achieves the same performance as CED (43.6%). What is more, it is obviously better than CDE with respect to stability, and the MS value is reduced by about 96.5%.

5.5. Sensitivity analysis

5.5.1. The initial point of the detection interval

The initial point of the detection interval determines whether to start the interval detection process, which in turn determines the overall performance of IDMO-SA. If the α value is too high, it may cause the interval detection to fail to work normally, leading to the rise of ER and MS. If the value of α is too low, the detection threshold will be too low, in which premature start interval detection for prediction without insufficient training may reduce its accuracy and increase MS value.

Different values of α are compared from the following range of {0.5, 0.6, 0.7, 0.8, 0.9}, as shown in Fig. 7. We notice that with the increase of α , the accuracy of IDMO-SA gradually increases and tends to converge at 0.8. Comparing $\alpha = 0.8$ and 0.9, we recognize that the accuracy rate has stabilized to 93.6%, but both ER and MS have increased to a certain extent. This shows that α is not the larger the better. If α is too large, the multi-objective loss module runs too late, which causes insufficient training in early detection and stability. Comparing $\alpha = 0.5, 0.6, 0.7$ and 0.8, we also find that α too small will also affect its ER and MS. This is because the results in the early stage of prediction are very unstable (see Fig. 3). When α is small, these unstable results will turn on the multi-objective detection module prematurely,

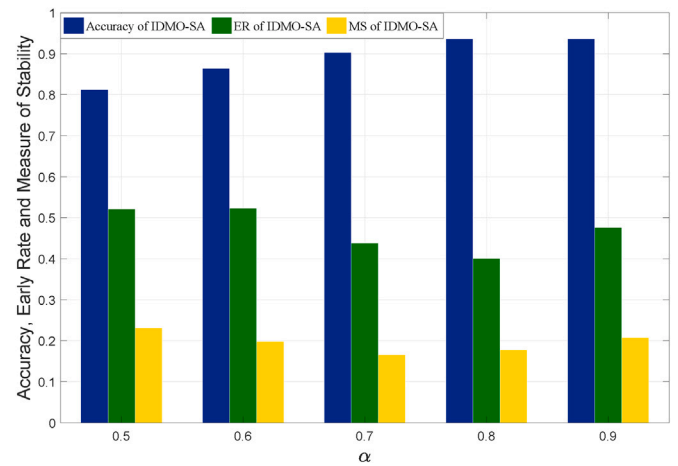


Fig. 7. Accuracy rate, ER and MS corresponding to different values of α in Weibo dataset.

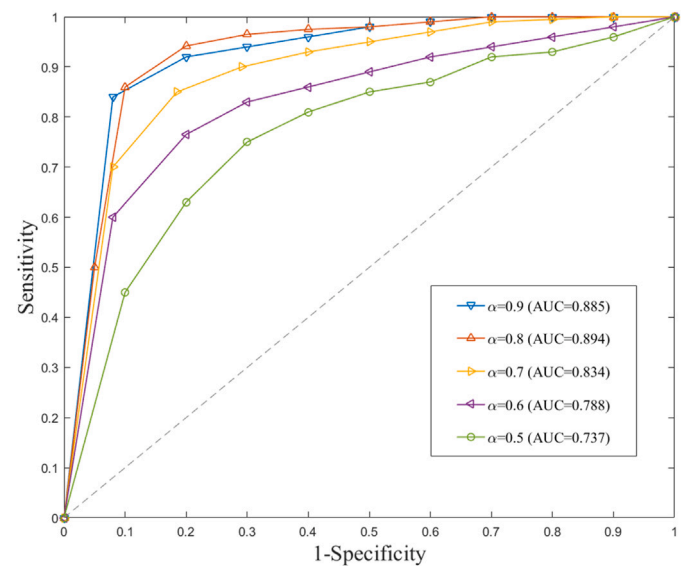


Fig. 8. ROC curves for different values of α in Weibo dataset.

then causing its ER and MS to increase. Therefore, it is very necessary to find an appropriate α in the experiment. ROC curves for different values of α are shown in Fig. 8. The larger the value of sensitivity, the better the detecting results of the model on true positive samples; the smaller the value of 1-specificity, the better the detecting results of the model on false positive samples. We notice that as α increases, its ROC curve is closer to (1, 0), indicating that the performance of the model is also improving. When $\alpha = 0.8$ and $\alpha = 0.9$, it can no longer be judged directly by ROC curve. We calculate the area under the ROC curve (AUC), and notice that when $\alpha = 0.8$, AUC reached a maximum of 0.894, the performance of the proposed model is best.

5.5.2. The length of the detection interval

The length of the detection interval determines the amount of training in the detection process, which in turn determines the overall performance of IDMO-SA. Intuitively, the longer the detection interval, the more adequate the training result and the better its performance. However, a too long detection interval may lead to training redundancy, which will not increase its accuracy while increasing the computational load.

To find a suitable detection interval length, we compared the experimental results of different lengths from the following range of {25,

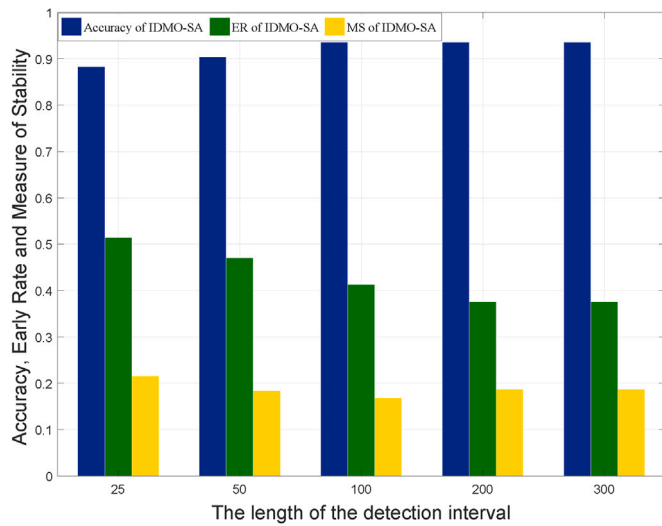


Fig. 9. Accuracy rate, ER and MS corresponding to different lengths of the detection interval in Weibo dataset.

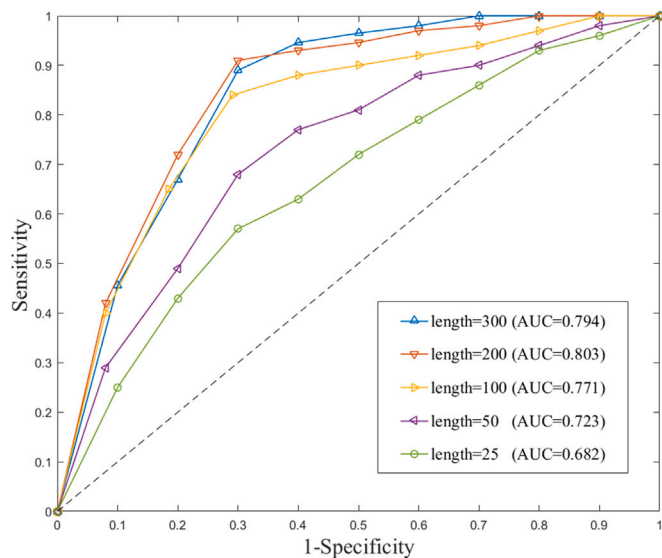


Fig. 10. ROC curves for different lengths of the detection interval in Weibo dataset.

50, 100, 200, 300}. Fig. 9 illustrates the accuracy rate, ER and MS of different lengths. From the experimental results, we notice that as the length of interval increases, the accuracy rate is gradually increasing, the ER and MS is gradually decreasing. When the length is 100, the accuracy rate tends to converge. When the length is 200, although the length of the interval is increasing, its accuracy rate, ER and MS will not change anymore. ROC curves for different lengths of the detection interval are shown in Fig. 10. The larger the value of sensitivity, the better the detecting results of the model on true positive samples; the smaller the value of 1-specificity, the better the detecting results of the model on false positive samples. We notice that as length increases, its ROC curve is closer to (1, 0), indicating that the performance of the model is also improving. When length = 100, length = 200 and length = 300, it can no longer be judged directly by ROC curve. We calculate the area under the ROC curve (AUC), and notice that when length = 200, AUC reached a maximum of 0.803, the performance of the proposed model is best.

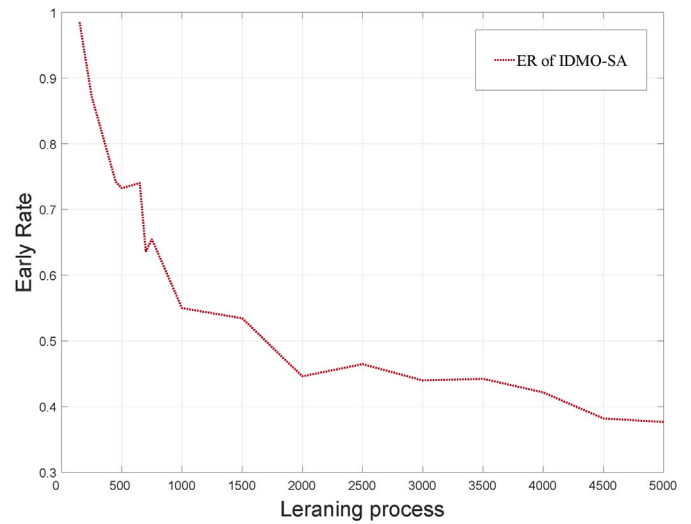


Fig. 11. The early rate change curve with the training process in Weibo dataset.

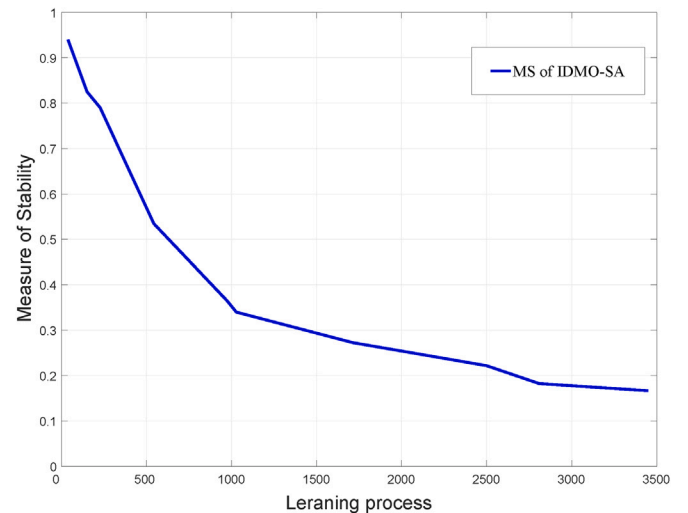


Fig. 12. The measure of stability change curve with the training process in Weibo dataset.

5.5.3. The early detection

The early rate is proposed to evaluate the time taken when the prediction result is larger than a certain fixed value (this value is set to 0.875). The smaller ER value means that the prediction result can reach 0.875 in a shorter time. Fig. 11 illustrates the detailed trend of ER value changes during the training in Weibo dataset. In the early stage of the training process, although ER increases slightly in some regions, it still keep a rapidly decreasing trend on the whole. As the training process continues, the multi-objective loss function continuously adjusts the prediction results. As the prediction results continue to increase, the detection interval moves and adjusts to finally converge to 0.376.

5.5.4. The stability analysis

Measure of stability in (34) is used to evaluate the fluctuation degree of the prediction results. When a prediction value is larger than a fixed value (0.875), we take the moment as the initial point and construct an interval with a fixed length (100), then difference between each predicted value and the average value in this interval is calculated. Therefore, the smaller MS is, the more stable prediction results are.

Fig. 12 illustrates the detailed trend of MS value changes during the training in Weibo dataset. In the early stage of the training process, the MS value drops sharply which means that at this stage, under the continuous adjustment of the multi-objective loss function, the gap between the prediction results is continuously reduced. In the middle and late stages of the training process, the MS value converged to 0.187, and the prediction results became stable, which undoubtedly shows the effectiveness of the multi-objective loss function.

6. Conclusion

In this paper, we propose a novel detection method (IDMO-SA) of rumors in OSNs. We utilize the text content and timestamp of the source microblog and corresponding reporting sequence as input, and extract the key feature maps through the CNN network with spatial attention mechanism, which solves the problem of missing local important information. Then, these feature maps are inputted into the GRU module to further explore more features of the rumor in the spreading process. To achieve early detection, we treat the detection point as a training objective and continuously adjust and optimize it instead of using a prefixed value during the training process, thereby enhancing its adaptability and universality. Meanwhile, accuracy and stability are regarded as the other two training objectives to ensure the reliability of detection. Due to the introduction of hyperparameters when aggregating multi-objectives which affect the final result, we utilize a concise and effective method based on convex optimization techniques to parameterize them so that they can adaptively change during the entire model learning process. Unlike the traditional method that requires the entire sequence data set as the test sample, in order to reduce the computational cost, we propose a sliding interval detection method, which only needs to find the detection point and perform the detection within the detection interval. Through continuous learning of features, the detection points are adaptively adjusted to make it more universal. Through the experiments, we systematically verify the effectiveness of the proposed method and the results show that the proposed method outperforms state-of-art methods.

It is worth noting that we think the more important thing is whether rumors are intentionally created. Many scientific findings are later discovered wrong, but we do not consider they are rumors. In this paper, we mainly focus on textual rumors in online social networks. However, in recent years, rumors have begun to appear in some new forms: picture rumors, rumors that pictures are mixed with text, video rumors, and rumors that video and text are mixed, not limited to the form of text. In the future, we will study the detection of multimedia rumors (such as pictures, videos, etc.), and further explore from the perspective of user relationships (such as relationship network structure).

CRedit authorship contribution statement

Pengfei Wan: Conceptualization, Methodology, Software, Writing – original draft. **Xiaoming Wang:** Conceptualization, Data curation. **Guangyao Pang:** Software, Visualization, Investigation. **Liang Wang:** Investigation, Validation. **Geyong Min:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant No. 61872228, 62071283) and the Shaanxi Provincial Key R&D Plan of China (Grant No. 2020ZDLGY10-05).

References

- Alkhodair, S. A., Ding, S. H. H., Fung, B. C. M., & Liu, J. (2020). Detecting breaking news rumors of emerging topics in social media. *Information Processing & Management*, 57(2), Article 102018. <http://dx.doi.org/10.1016/j.ipm.2019.02.016>.
- Bian, T., Xiao, X., Xu, T., Zhao, P., Huang, W., Rong, Y., et al. (2020). Rumor detection on social media with bi-directional graph convolutional networks. In *Proc. AAAI conf. artif. intell.* Vol. 34, (01), (pp. 549–556). <http://dx.doi.org/10.1609/aaai.v34i01.5393>.
- Elhadad, M. K., Li, K. F., & Gebali, F. (2020). COVID-19-FAKES: A twitter (Arabic/english) dataset for detecting misleading information on COVID-19. In *Proc. int. conf. intell. netw. collaborative syst.* (pp. 256–268). Springer, http://dx.doi.org/10.1007/978-3-030-57796-4_25.
- Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. In *Proc. int. conf. artif. intell. statist.* (pp. 315–323). <http://proceedings.mlr.press/v15/glorot11a/glorot11a.html>.
- Guo, H., Cao, J., Zhang, Y., Guo, J., & Li, J. (2018). Rumor detection with hierarchical social attention network. In *Proc. ACM int. conf. inf. knowl. manage.* (pp. 943–951). <http://dx.doi.org/10.1145/3269206.3271709>.
- Hanselowski, A., Stab, C., Schulz, C., Li, Z., & Gurevych, I. (2019). A richly annotated corpus for different tasks in automated fact-checking. In *Proceedings of the 22nd conference on computational natural language learning* (pp. 1–12). <http://dx.doi.org/10.18653/v1/K19-1046>.
- He, Q., Chang, K., Lim, E. -P., & Banerjee, A. (2010). Parallelizing word2vec in shared and distributed memory. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(10), 1795–1808. <http://dx.doi.org/10.1109/TPDS.2019.2904058>.
- Husain, S. S., & Bober, M. (2019). REMAP: Multi-layer entropy-guided pooling of dense CNN features for image retrieval. *IEEE Transactions on Image Processing*, 28(10), 5201–5213. <http://dx.doi.org/10.1109/TIP.2019.2917234>.
- Ji, S., Satish, N., Li, S., & Dubey, P. K. (2019). Parallelizing word2vec in shared and distributed memory. *IEEE Transactions on Parallel and Distributed Systems*, 30(9), 2090–2100. <http://dx.doi.org/10.1109/TPDS.2019.2904058>.
- Jin, Z., Cao, J., Jiang, Y. -G., & Zhang, Y. (2014). News credibility evaluation on microblog with a hierarchical propagation model. In *Proc. IEEE int. conf. on data mining* (pp. 230–239). IEEE, <http://dx.doi.org/10.1109/ICDM.2014.91>.
- Kingma, D., & Ba, J. (2015). Adam: A method for stochastic optimization. In *Proc. int. conf. learn. representations* (pp. 1–15). <http://dx.doi.org/10.48550/arXiv.1412.6980>.
- Lai, Y., Feng, Y., Yu, X., Wang, Z., Xu, K., & Zhao, D. (2019). Lattice CNNs for matching based chinese question answering. 33, In *Proc. AAAI conf. artif. intell.* (pp. 6634–6641). <http://dx.doi.org/10.1609/aaai.v33i01.33016634>.
- Lancet, T. (2020). COVID-19: Fighting panic with information. *Lancet*, 395(10224), 537–538. [http://dx.doi.org/10.1016/S0140-6736\(20\)30379-2](http://dx.doi.org/10.1016/S0140-6736(20)30379-2).
- Leng, Y., Zhai, Y., Sun, S., Wu, Y., Selzer, J., Strover, S., et al. (2021). Misinformation during the COVID-19 outbreak in China: Cultural, social and political entanglements. *IEEE Transactions on Big Data*, 7(1), 69–80. <http://dx.doi.org/10.1109/TBDATA.2021.3055758>.
- Li, H., Li, T., Wang, W., & Wang, Y. (2018). Dynamic participant selection for large-scale mobile crowd sensing. *IEEE Transactions on Mobile Computing*, 18(12), 2842–2855. <http://dx.doi.org/10.1109/TMC.2018.2884945>.
- Liang, G., Yang, J., & Xu, C. (2016). Automatic rumors identification on Sina Weibo. In *Proc. int. conf. natural comput. fuzzy syst. knowl. discovery* (pp. 1523–1531). <http://dx.doi.org/10.1109/FSKD.2016.7603402>.
- Lin, X., Chen, H., Pei, C., Sun, F., Xiao, X., Sun, H., et al. (2019). A pareto-efficient algorithm for multiple objective optimization in e-commerce recommendation. In *Proc. ACM conf. recommender syst.* (pp. 20–28). <http://dx.doi.org/10.1145/3298689.3346998>.
- Lin, T. -Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2020). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2), 318–327. <http://dx.doi.org/10.1109/TPAMI.2018.2858826>.
- Liu, Y., Jin, X., & Shen, H. (2019). Towards early identification of online rumors based on long short-term memory networks. *Information Processing & Management*, 56(4), 1457–1467. <http://dx.doi.org/10.1016/j.ipm.2018.11.003>.
- Ma, J., Gao, W., Wei, Z., Lu, Y., & Wong, K. -F. (2015). Detect rumors using time series of social context information on microblogging websites. In *Proc. ACM int. conf. inf. knowl. manage.* (pp. 1751–1754). <http://dx.doi.org/10.1145/2806416.2806607>.
- Ma, J., Gao, W., & Wong, K. -F. (2017). Detect rumors in microblog posts using propagation structure via kernel learning. In *Proc. annu. meeting assoc. comput. linguistics* (pp. 708–717). Association for Computational Linguistics, <http://dx.doi.org/10.18653/v1/P17-1066>.
- Ma, J., Gao, W., & Wong, K. -F. (2019). Detect rumors on twitter by promoting information campaigns with generative adversarial learning. In *Proc. world wide web conf.* (pp. 3049–3055). <http://dx.doi.org/10.1145/3308558.3313741>.

- Maryland (2021). A case of rumours about COVID-19 released by the Maryland government, USA. [Online] Available: <https://govstatus.egov.com/md-coronavirus-rumor-control/vaccine-rumors><https://govstatus.egov.com/md-coronavirus-rumor-control/vaccine-rumors>.
- Mohler, G., & Brantingham, P. J. (2018). Privacy preserving, crowd sourced crime Hawkes processes. In *Proc. int. workshop social sens.* (pp. 14–19). IEEE, <http://dx.doi.org/10.1109/SocialSens.2018.00016>.
- Pang, G., Wang, X., Hao, F., Xie, J., Wang, X., Lin, Y., et al. (2019). ACNN-FM: A novel recommender with attention-based convolutional neural network and factorization machines. *Knowledge-Based Systems*, 181, Article 104786. <http://dx.doi.org/10.1016/j.knosys.2019.05.029>.
- Ruszczynski, A. (2011). *Nonlinear optimization*. Princeton Univ. Press, ISBN: 1400841054.
- Sener, O., & Koltun, V. (2018). Multi-task learning as multi-objective optimization. *CoRR*, 15(12), 6492–6499. <http://dx.doi.org/10.5555/3326943.3326992>.
- Song, C., Yang, C., Chen, H., Tu, C., Liu, Z., & Sun, M. (2021). CED: Credible early detection of social media rumors. *IEEE Transactions on Knowledge and Data Engineering*, 33(8), 3035–3047. <http://dx.doi.org/10.1109/TKDE.2019.2961675>.
- Wan, X., Wang, Z., Han, Q. -L., & Wu, M. (2019). A recursive approach to quantized H_∞ state estimation for genetic regulatory networks under stochastic communication protocols. *IEEE Transactions on Neural Networks and Learning Systems*, 30(9), 2840–2852. <http://dx.doi.org/10.1109/TNNLS.2018.2885723>.
- Wan, P., Wang, X., Wang, X., Wang, L., Lin, Y., & Zhao, W. (2021). Intervening coupling diffusion of competitive information in online social networks. *IEEE Transactions on Knowledge and Data Engineering*, 33(6), 2548–2559. <http://dx.doi.org/10.1109/TKDE.2019.2954901>.
- Woo, S., Park, J., Lee, J. -Y., & So Kweon, I. (2018). Cbam: Convolutional block attention module. In *Proc. Eur. conf. comput. vision* (pp. 3–19). http://dx.doi.org/10.1007/978-3-030-01234-2_1.
- Wu, K., Yang, S., & Zhu, K. Q. (2015). False rumors detection on sina weibo by propagation structures. In *Proc. IEEE int. conf. data eng.* (pp. 651–662). <http://dx.doi.org/10.1109/ICDE.2015.7113322>.
- Wu, F., Zhao, S., Yu, B., Chen, Y. -M., Wang, W., Song, Z. -G., et al. (2020). A new coronavirus associated with human respiratory disease in China. *Nature*, 579(7798), 265–269. <http://dx.doi.org/10.1038/s41586-020-2202-3>.
- Xu, J., Rao, Z., Xu, L., Yang, D., & Li, T. (2019). Incentive mechanism for multiple cooperative tasks with compatible users in mobile crowd sensing via online communities. *IEEE Transactions on Mobile Computing*, 19(7), 1618–1633. <http://dx.doi.org/10.1109/TMC.2019.2911512>.
- Yang, X., Lyu, Y., Tian, T., Liu, Y., Liu, Y., & Zhang, X. (2020). Rumor detection on social media with graph structured adversarial learning. In *Proc. int. joint conf. artif. intell.* (pp. 1417–1423). <http://dx.doi.org/10.24963/ijcai.2020/197>.
- Yu, F., Liu, Q., Wu, S., Wang, L., Tan, T., et al. (2017). A convolutional approach for misinformation identification. In *Proc. int. joint conf. artif. intell. org.* (pp. 3901–3907). <http://dx.doi.org/10.24963/ijcai.2017/545>.
- Yuan, C., Ma, Q., Zhou, W., Han, J., & Hu, S. (2019). Jointly embedding the local and global relations of heterogeneous graph for rumor detection. In *Proc. IEEE int. conf. data mining* (pp. 796–805). IEEE, <http://dx.doi.org/10.1109/ICDM.2019.00090>.
- Zarocostas, J. (2020). How to fight an infodemic. *The Lancet*, 395(10225), 676. [http://dx.doi.org/10.1016/S0140-6736\(20\)30461-X](http://dx.doi.org/10.1016/S0140-6736(20)30461-X).
- Zhou, H., Ma, T., Rong, H., Qian, Y., Tian, Y., & Al-Nabhan, N. (2022). MDMN: Multi-task and domain adaptation based multi-modal network for early rumor detection. *Expert Systems with Applications*, 195, Article 116517. <http://dx.doi.org/10.1016/j.eswa.2022.116517>.