Pattern Recognition 113 (2021) 107818

Contents lists available at ScienceDirect

Pattern Recognition

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

BCMM: A novel post-based augmentation representation for early rumour detection on social media

Yongcong Luo[a,b], Jing Ma[a,*], Chai Kiat Yeo[b]

[a] College of Economics and Management, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
[b] School of Computer Science and Engineering, Nanyang Technological University, 50 Nanyang Avenue 639798, Singapore

ARTICLE INFO

Article history:
Received 15 May 2020
Revised 27 October 2020
Accepted 29 December 2020
Available online 12 January 2021

Keywords:
Early rumour detection
Topology network
Metadata
Backward compression mapping
Semantic augmentation

ABSTRACT

Online social media (OSM) has become a hotbed for the rapid dissemination of disinformation or rumour. Therefore, rumour detection, especially early rumour detection (ERD), is very challenging given the limited, incomplete and noisy information. Although there are some researches on earlier rumour detection, most of their studies require a larger dataset or a longer detection time span, i.e., the rumour detection efficiency needs to be improved. In this paper, we focus on a shorter detection time span which also means fewer online posts to achieve the task of ERD. We proposed a novel post-based augmentation representation approach to process post content of rumour events in the early stages of their dissemination, i.e., backward compression mapping mechanism (BCMM). In addition, we combine BCMM with gated recurrent unit (GRU) to represent post content, topology network of posts and metadata extracted from post datasets. We apply a three-layers GRU to enhance the representation of dataset within one hour after the occurrence of a social media event, i.e., BCMM-GRU. The steps are as follows: (1) we input the first-hour data into the first layer; (2) the first 40 min of data are channelled into the second layer with the output of the first layer making a full mapping to the second layer simultaneously; (3) the first 20 min of data are sent to the third layer while the output of the second layer applies a full mapping to the third layer simultaneously. The evaluation of BCMM-GRU's performance entails applying k-fold cross-validation (CV) set-up on four available real-life rumour event datasets. The experimental results are superior to the baselines and model variants and achieve a high accuracy of 80.09% and F1-score of 80.18%.

1. Introduction

Rumour or fake news disseminate misinformation on fake/unsubstantiated events that are mostly related to hot issues and events as well as emergencies. Rumour may sow social panics and discord which may be intended by those who disseminate them or may be the tools for some people to achieve some ulterior motives [1]. There are many well-known and large-scale OSM, such as the Sina MicroBlog, WeChat, Twitter, etc. [2], which provide an excellent platform for personal and mass communication all over the world. Thus, OSM is also an ideal breeding ground for rumour to spread. It has become increasingly popular to understand the emergence and development of rumour events on OSM. At the time of writing this paper, rumours and misinformation on the coronavirus COVID-19 spread faster than the virus throughout the OSM spreading fear and panic. Therefore, in order to weaken or limit the spread of rumour and the adverse effects [3], how to quickly and effectively identify rumours has become an urgent problem to be addressed.

A typical rumour detection process can be learnt from [4]: suspected rumour detection, rumour tracking, stance classification and veracity classification, i.e., once a post is recognized as a suspected rumour, it will track the post source, classify the rumour stance, and finally verify whether the post is a rumour or non-rumour. Recent research has shown that the ratio of stances on posts extracted from OSM can be useful in the recognition of the post or fake news [5,6]. In other words, if the stances are recognized as "disagreed" in a certain percentage of posts which is akin to "majority opinion", then the post is most likely to be fake, a direct result of the so-called crowd wisdom [7,8]. Therefore, we will consider the application of the rumour stance to the entire ERD. In our work, we skip the "suspected rumour detection" module and focus directly on the simple but timely rumour detection process.

The traditional research on rumour only focuses on the textual representation of information sources and related processing

* Corresponding author.
E-mail address: search@nuaa.edu.cn (J. Ma).

https://doi.org/10.1016/j.patcog.2021.107818
0031-3203/

methods [7], and lacks attention to the many special characteristics of OSM [4]. These characteristics include the social topology network of the poster (we define it as the person who posted the post), number of "subscribes" or repost, etc. [9] In this case, we not only pay attention to the text content of the post, but also consider the topology architecture of social network and post metadata (e.g., the number of followers, reposts, and following etc.). Textual analysis of post content faces a difficult challenge, i.e., semantic consistency after mathematical processing. Hence, for post content, we consider content semantic augmentation (CSA) mechanism which can improve the semantic representation performance of post content. Unlike simple post content feature semantic representation, a network topology can better show the architecture extracted from the posters community on OSM [10,11], which will improve the performance of rumour detection. In addition, the traditional methods just apply word2vec to construct the adjacency matrix to show the relationship among the posters on OSM and ignore the overall community connection of rumour participants. For the topology network of posters, we adopt the random walk with restart (RWR) method to capture the global connection information among posters' interaction in OSM [12,13] using the high-dimensional network structural information [14]. For post metadata, it presents rich important and special information related to followers, likes and replies (where the replies can construct a chain of information flow just like a frame of tree structures) which can provide extremely useful supplementary data to enhance semantic representation of post content and topology network (tree-structure of reply chain contributes to the construction of the topology network) [15,16]. Unlike existing works which just focus on post content [17], we believe that different data types and data sources are more conducive to semantic representation [7,18], i.e., metadata and textual content are different expressions of the same content. This multi-type, multi-source approach is more helpful to enhance the semantic representation of the data content [19].

Hence, we still consider constructing features-based vector to further enhance the semantic representation of post content, topology network and metadata [20,21]. Then, we put these vectorization matrices into BCMM-GRU for processing, to prepare for the subsequent binary classification of rumour and non-rumour. The BCMM-GRU is "Backward Compression Mapping Mechanism-Gated Recurrent Unit", in which, we segment the one-hour dataset into three sets, i.e., data of one hour, data of 40 min and data of 20 min: (1) we send the vectorization matrices representation of post content, topology network and metadata within an hour to their respective first-layer GRU for processing, where the initial vector dimension is equal to the first-layer GRU, and the first-layer output is $Layer_{out\_1}$, namely first stage; (2) we put vectorization matrices representation of data of 40 min into the second-layer GRU for processing, the dimension of this layer is two thirds of the first layer while $Layer_{out\_1}$ makes a full mapping to the second layer, and the second-layer output is $Layer_{out\_2}$, namely second stage; (3) similar to the first and second stages, we put data of 20 min into the third-layer GRU, the dimension of this layer is half of the second layer while $Layer_{out\_2}$ makes a full mapping to the third layer, and the third-layer output is $Layer_{out\_3}$, namely third stage. The core of early rumor detection lies in its early nature, and the small amount of data in the early stage is not conducive to ensuring the performance (i.e., accuracy) of the experiment. Therefore, we chose a time span of 60 min for the experiment, which can reflect the early characteristics. However, in order to improve the performance of the experiment, we cut the length of 60 min into three segments (i.e., 60-min, 40-min and 20-min), and at the same time increases the amount of data used for the experiment. Thereafter, we combine $Layer_{out\_3}$ from post content, topology network and metadata respectively into a joint representation, which is then integrated with the weight vector calculated from the stance recognition for element-wise multiplication to detect rumour and non-rumour. The overview of model architecture can be seen in Section 3.1.

The main contributions of this paper are as follows:

- Compared with traditional textual representation methods, we not only study the representation of the entire post sentence, but also consider the enhancement of each word and sentiment in the post sentence to the text representation, i.e., we construct the CSA mechanism.
- In addition, we abandon the single textual representation, and at the same time pay attention to the posters' interactive information on the rumour topic on OSM platform, i.e., we consider the topology network attribution constructed by RWR to represent the distribution graph of these posters.
- Meanwhile, we also consider the post metadata (e.g., number of followers, following, replies and repost, etc.), which comprises rich explicit and implicit information used to enhance semantic representation of post content and construct tree-structure of reply chain, where the tree-structure can help enrich the topology network architecture.
- In particular, we construct BCMM-GRU mechanism to process vectorization representation calculated from post content, topology network and post metadata respectively. The BCMM-GRU can achieve semantic augmentation representation of posts in shorter time span or fewer posts for ERD.

The remainder of this paper is organized as follows. Section 2 reviews the related work of rumour detection, as well as the traditional applied methods and features expression. We describe the details of our proposed method in Section 3. The method of CSA, features, topology network, especially the BCMM-GRU mechanism and the principle of final rumour detection integrated with weight parameter derived from stance recognition are also detailed. Section 4 gives the experimental settings and description of baselines and variants of our proposed method. Section 5 presents the results as well as the comparison with the baselines. Section 6 concludes the paper.

2. Related work

The current red hot interest in data science and artificial intelligence has led to many researchers adopting machine learning methods for the rumour detection [22]. Hence, many studies pay attention to employing machine learning methods to research text mining [23], i.e., considering the context vector generated from posts or tweets. Rehman et al. [24] integrated CNN and LSTM by applying LSTM and very deep CNN model for sentiment analysis in movie reviews. In addition, Zhao et al. [25] learned deep emotion features combining one layer CNN-LSTM and two-layer CNN-LSTM for speech emotion recognition. However, it is not clear how the designed networks recognize the emotion. Luo et al. [26] proposed a TWs-LSTM method using the commodity titles extracted from Tmall platform in the processing of commodity entity name recognition. The result is not so good and this may be due to them not enhancing the semantics of the commodity. Moreover, these combinations of NN models ignore the time complexity of integrating two NN models with the same dimension vector [27], which will lead to high computational cost [28]. It is to be noted that, RWR is able to address these shortcomings and is easy to capture the topological architecture information [29].

For post content, there are some works on content semantic augmentation representation. Zhang et al. [30] believe that highly condensed text contain rich knowledge information which is useful for rumour detection. In view of this, they constructed Multimodal Knowledge-aware Network (MKN) to enhance semantic representation of short texts of posts by retrieving external knowl-

edge from real-world knowledge graph. In order to construct and enhance accurate semantic structure representation, Yang et al. [31] proposed a dynamic slide-window based text feature scoring and extraction mechanism, which also contributes to the design of effective rumour detection scheme. However, the size of slide-window is difficult to select, and the feature scoring mechanism may not support the semantic augmentation representation well. There is also interesting research on text semantic augmentation representation. Zheng et al. [32] enhance the semantic augmentation representation by projecting features twice into the constructed medium semantic enhancement space (MMSES), which can also enhance the discrimination capability of textual features. However, the "twice" projection incur additional computational effort, which is not conducive to effective operation in subsequent experiment. Cai et al. [33] regarded user content as temporal text data instead of plain text to extract latent temporal patterns to construct behavior enhanced deep model for malware detection.

Tam et al. [34] applied graph-based scan integrated with multimodal approach and also considered different types of features (e.g., users, posts, etc.) for rumour detection. However, they only employed textual features without considering the topology network (i.e., the relationship of posters), which is an important component that we focus on to improve the performance of rumour detection. In this case, there are some works about topology network on OSM. Wang et al. [35] named OSM as "We the Media" which is full of many different views or topics surrounding a specific public opinion event, where these public opinions are multidimensional, multilayered and possess multiple attributes. They built a multidimensional network topic detection model oriented towards the topology. This model can also help identify the post stance related to rumour events. In order to research the impact of social media on diffusion of sustainable mobility opinions, Borowski et al. [36] proposed a dynamic agent-based model by comparing topology network architecture and social media influence on opinion diffusion, where the topology network contributes to the most effective influence. Lavender et al. [37] designed an Online Resource for Social Omics mechanism to connect biology dataset, in which, they consider topology of network graph (representing hosted biology dataset) to cluster samples from related systems in different biology datasets. These studies only considered topology network and a bit of topology network-based auxiliary features. Hence these features are not sufficient to support the representation of the topology attributes [38]. In our proposed model, we consider the topology network of posters on OSM and enrich the topology-based features setup simultaneously.

For post metadata, Morgan-Lopez et al. [39] considered linguistic and metadata features to predict the age of Twitter users. Their results show that although the performance of the single metadata-based model is not very good, the combination of metadata-based features and linguistic-based features have excellent performance. This also indicates that the metadata helps improve the performance of the combined model, which will enhance text semantic representation. Topic-based or content-based constructed semantic features often present the problem where the corresponding word cannot be accurately matched [40]. In order to cope with this problem, some studies employed multiple layers of the text data defined by metadata attributes to construct features for text semantic representation [41], which is generally more effective than the conventional features representation. Pandya et al. [42] devised an innovative model based on CNN to study age prediction from Twitter dataset in the aging population where the social media specific metadata help to extract content semantic information. This effort only considers the common text information of metadata ignoring the auxiliary features related to metadata. Metadata contains a lot of basic profile information about posters (e.g., screen name, number of friends, number of followers, retweet count, and age of the account) which can provide useful complementary semantic features and have the potential advantage of language portability [43]. This is the reason that we consider the post metadata as an important component in our model.

We also employ one special idea ("crowd wisdom") [8], i.e., post stance. Traditional methods on stance recognition employ post content features to present the semantic representation, e.g., bad words or exclamation mark, etc. Lukasik et al. [44] exploited Hawkes Process to classify rumour stance by considering textual information and temporal features. However, they only achieved an average accuracy of 69.43% on four datasets, and they only consider three types of stance, which is not comprehensive. Hence, Zubiaga et al. [45] classified stances into four categories and test the effectiveness of four sequential classifiers on eight rumour datasets. They do not provide any insight into the network topology of users in the conversation [46]. There are also other works on stance recognition, such as Lin et al. [47] who considered the comment topic which is useful to mine latent information in rumour comments to recognize stances in web texts. For us, we employ the stance as the weight vector of joint representation ($Layer_{out\_3}$ from post content, topology network and metadata respectively) for element-wise multiplication to detect rumour and non-rumour.

3. Methodology

3.1. Overview of model architecture

Our study focuses on improving the recognition performance of rumour detection extracted from the posts on different hot button issues and events as well as emergency events or news in OSM. The overall architecture of the proposed model is shown in Fig. 1.

We consider three categories (i.e., post content, topology network and post metadata) to represent the overall post data semantic for subsequent rumour detection. Firstly, we put these categories into a vectorization representation (see Section 3.2), i.e., post content with content semantic augmentation (CSA) and considering the content-based features simultaneously; topology network which exploits the topology-based features generated from RWR and tree-structure attribute (extracted from metadata), and post metadata employ metadata-based features. Then these vectorization representations are fed into the proposed BCMM-GRU mechanism (see Section 3.3) for processing so as to detect rumour in a shorter time span and with less experimental data. Thereafter, we combine $Layer_{out\_3}$ of the three categories that are generated from post content, topology network and metadata respectively for joint vectorization representation. Finally, we exploit the ratio of different stance types as the parameter to adjust the joint vectorization representation for the last task of determining rumour or non-rumour (see Section 3.4). All parameters will be introduced in the corresponding section.

3.2. Post content, topology network, and post metadata

For **post content**, we combine words and post sentence, and employ sentiment score (calculated from a constructed sentiment dictionary) as a parameter W_{ss} (which is the weight vector of sentiment words in the current post sentence) to adjust the content semantic representation for joint text content representation, namely content semantic augmentation (CSA). The calculation of W_{ss} is as follows:

$$W_{ss}\begin{cases}\omega_{po}=\omega+i, & positive, 0<i<1\\ \omega_{ne}=\omega-i, & negative, -1<i<0\end{cases}, \tag{1}$$

where ω is initial weight $\omega=\frac{1}{k}$ (k is the number of words in current post), ω_{po} represents the positive word (e.g., agree, yes, right,

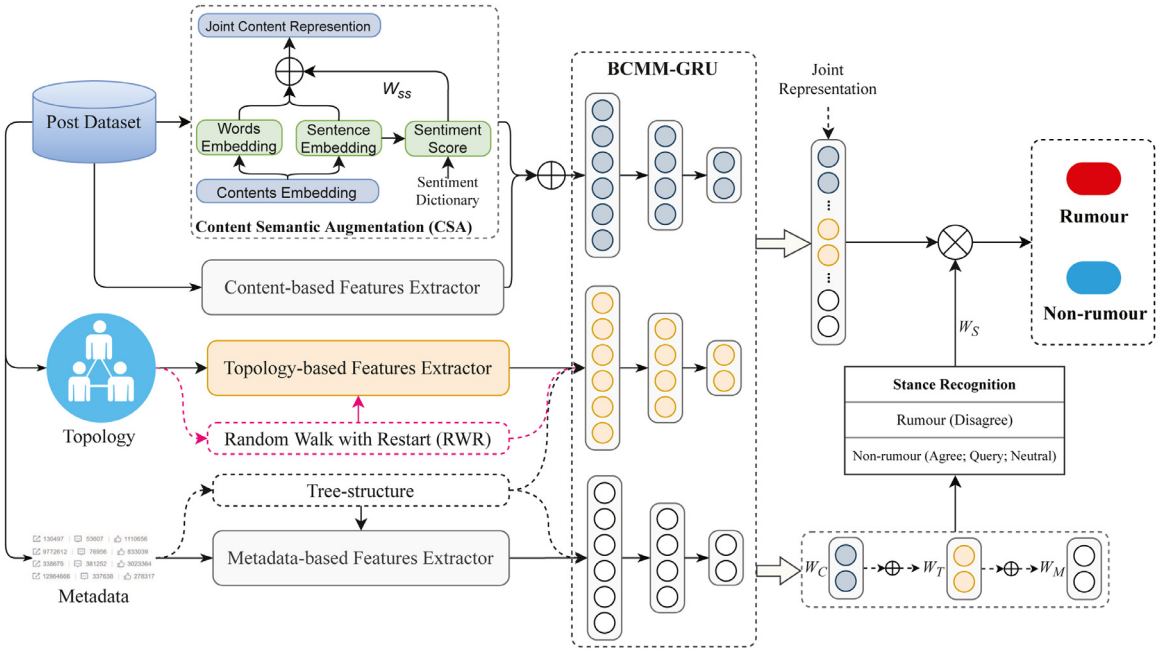

Fig. 1. Overview of the proposed model architecture.

etc.) weight of a sentiment word type, ω_{ne} is negative word (e.g., disagree, no, none, etc.) weight and i is the sentiment score calculated from the post content using the sentiment dictionary.

Here, we instead of using the usual the method that only applies to the entire sentence content, we exploit each word or vocabulary and sentiment coefficient to enhance the semantic representation of the post content, which will more accurately present the post content semantics.

$$V_{ws} = W_{ss}(V_s \times (V_{w1} \odot V_{w2} \odot \cdots \odot V_{wk})), \tag{2}$$

where V_{ws} is the vectorization of the joint content representation, V_s is the vectorization representation of a sentence, "$\times$" is cross product, $\odot$ refers to element-wise multiplication and V_{wk} is the vectorization representation of the kth word in the current sentence.

The mathematical vector generated by Word2Vec is much denser. Therefore, the Word2Vec method is applied to vectorization of words, so that each word can be mapped to a set of vectors for mathematical computation and simulation. The Word2Vec working process can be illustrated as follows.

The words are put into the Word2Vec model to calculate word vector of the target word V_{wk} which the words are generated from the current sentence. The Word2Vec has two language conversion models, CBOW and Skip-gram, where the CBOW model can output the word vector of the target word V_{wk} according to the adjacent words information $\left(V_{w(k-2)}, V_{w(k-1)}, \ldots, V_{w(k+1)}, V_{w(k+2)}\right)$ of the target word. The word vector of the target word V_{wk} corresponds to the sentence containing the word V_{wk}. After word2vec processing, a vector model is built for subsequent experiment calls. And the parameter setting will be introduced as follows.

The Vector dimension: size=100. The distance of words: window=2. The learning rate: alpha=0.0001. The Number of iterations: iter=10. And other parameters will be set default.

We also adopt the features in text content studies and called them content-based features as shown in Table 1. As we can see from Fig. 1, we combine V_{ws} and the vectorization representation of the content-based features to represent the post content named as V_C.

$$V_{Cf} = \left[V_{Cf\_1}, \ldots, V_{Cf\_k} \ldots, V_{Cf\_15}\right]$$
$$V_C = V_{ws} \times V_{Cf}, \tag{3}$$

where V_{Cf} is the combination vectorization representation of the content-based features, V_{Cf_k} is vectorization representation of the kth content-based features, "$\times$" is cross product.

For **post metadata**, as mentioned above, it can provide rich special cues related to a post (e.g., number of replies, reposts, comments on post, etc.) which can then provide useful complementary information and a potential advantage for ERD. Thus, we consider the metadata to employ features named as metadata-based features (see Table 2) which are designed as V_{Mf}, i.e., the vectorization representation of the metadata-based features.

$$V_{Mf} = \left[V_{Mf\_1}, \ldots, V_{Mf\_k} \ldots, V_{Mf\_11}\right], \tag{4}$$

where V_{Mf_k} is vectorization representation of the kth metadata-based feature.

Interestingly, there is a special attribute in the chain of replies (construction of the chain of information flow is like a frame of tree structures) that has a positive effect on the construction of the topology network. The tree-structure information flow can be seen in Fig. 2. In the figure, we can see that the red node is the source post (root node is like a tree root) which generates all the subsequent reposts or replies (we also name them as comments on the current post). The blue nodes are the branch nodes, i.e., the second layer nodes which represent reposts or replies to the root node. The black nodes are the last nodes related to the source post. In particular, the last one is named as the leaf node which is like a tree leaf. The repost direction is the propagation process of the rumour. All reposts or replies related to a source post constitute a tree-structure frame of the information flow, which will help construct the topology network. We believe that the tendency of the types of replies observed in a branch might also be indicative of the distribution in other branches, and hence useful to enhance the performance of model when using the tree as a whole.

We define these tree-structure posts as a graph $G = (V, E)$ (undirected), where V is the aggregation of all vertices (i.e., posts)

Table 1
Content-based features and description.

Features	Description
Bad_cnt	Number of bad words in the post.
Emoji_cnt	Number of emoji labels in the post.
Excla_cnt	Number of exclamation marks in the post.
Ment_cnt	Number of @/mentions in the post.
Neg_cnt	Number of negative words in the post.
Qu_mark	Number of question marks in the post.
Qu_word	Number of question words in the post.
RT_cnt	Number of RT marks in the post.
Sent_cnt	Number of sentiment words in the post.
Sent_score	Sentiment score of the post.
Source	How the post was posted.
Tag_cnt	Number of #/hashtags in the post.
Url_cnt	Number of URL labels in the post.
Veri_acc	Whether the account is a verified name.
W_cnt	Number of words in the tweet except source author's screen name.

Table 2
Metadata-based features and description.

Features	Description
Cm_cnt	Number of users who commented on this post.
Fli_cnt	Number of following in the source poster.
Flr_cnt	Number of followers in the source poster.
Frd_cnt	Number of friends in the source poster.
Geo_loc	Whether the post displays the geographic location.
Lk_cnt	Number of users who liked this post.
Pul_cnt	Number of users who have liked this post so far (e.g., "user favourites/likes").
Pup_cnt	Number of users who have posted.
Rel_ti	Reply time interval (time difference between comment and current post in minutes).
Rep_cnt	Number of users who have reposted this post.
Reo_ti	Repost time interval (time difference between repost and source post in minutes).

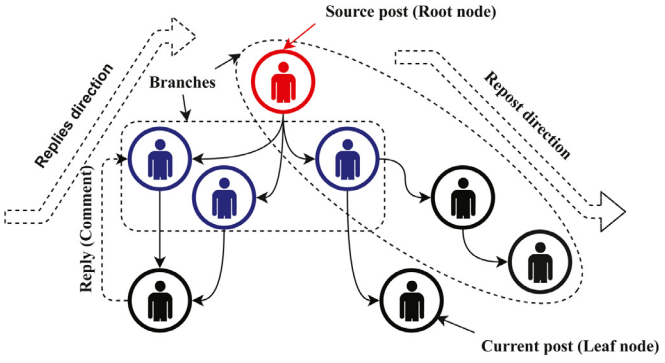

Fig. 2. Example of tree-structure.

in the graph, E is aggregation of all edges (the connection relationship between vertices and each vertex). Hence, having a posts sequence vectorization of p_{ts} as input, the output is a sequence of weight vectors, where the output of each element W_{ts_i} (ith W_{ts}) will not only depend on its tree-structure but also on the relationship of other tree-structure frames. The weight vector is shown in Eq. (5),

$$W_{ts} = \left(p_{ts\_ij}\right)_{N\times N} = \begin{pmatrix} p_{ts\_11} & p_{ts\_12} & \cdots & p_{ts\_1N} \\ p_{ts\_21} & p_{ts\_22} & \cdots & p_{ts\_2N} \\ \vdots & \vdots & \vdots & \vdots \\ p_{ts\_N1} & p_{ts\_N2} & \cdots & p_{ts\_NN} \end{pmatrix}, \tag{5}$$

where W_{ts} is the weight matrix of the tree-structure frame, p_{ts_ij} is the connection status between vertex p_{ts_i} and vertex p_{ts_j} in the tree-structure frame, N is the number of vertices. If vertices p_{ts_i} and p_{ts_j} are adjacent (interconnect), $p_{ts\_ij} = 1$; otherwise, $p_{ts\_ij} = 0$. Hence, from Eqs. (4) and (5), the vectorization representation of the metadata is shown in Eq. (6),

$$V_M = V_{Mf} * W_{ts} \tag{6}$$

where $*$ is hadamard product.

For **topology network**, its feature vector is constructed by the RWR method that captures the entire network topology information and extracts more characteristics of the nodes. The computation of RWR is as follows:

$$s^k_{a\to b} = rLs^{k-1}_{a\to b} + (1-r)e_a, \tag{7}$$

where $r, 0 \le r \le 1$ is the restart probability. We set the start node as α for random walk, and the leap probability relies on the weight of the relevant nodes. Furthermore, before each skip, it has a probability r to go back to the node α for the next loop of leap. $s^{k-1}_{a\to b}$ is the $n \times 1$ vector that represents the relevant score of node b visited after $k-1$ leap steps in random walk from node a. e_a is the start vector and is described as follows:

$$e_a(a) = \begin{cases} 1, & a \\ 0, & b, \forall b \ne a \end{cases} \tag{8}$$

L, which comprises L_{ab}, is the leap probability matrix derived from the weight of all the relevant nodes in the network graph, where the weight is calculated from the communication times, i.e., the reply times of the relevant nodes, namely v_{ab}. L_{ab} stores the relevant score of v_{ab} (i.e., node a and b) and is computed as follows:

$$L_{ab} = \frac{v_{ab}}{\sum_1^k v_{ak}}, 1 \le a, b \le k, \tag{9}$$

where $\sum_1^k v_{ak}$ is the sum of all the reply times of nodes interacting with node a, in which, v_{ak} contains node a's reply from itself, i.e., v_{aa}. Finally, according to Eqs. (7)–(9), we can obtain the relevant score matrix T consisting of $s_{a\to b}$, i.e., the topology matrix.

We call the features employed in the topology-based features (see Table 3) and they are designed as V_{Tf}, i.e., the vectorization

Table 3
Topology-based features and description.

Features	Description
Atti_cnt	Number of users who liked the comment.
Cos_simi	The cosine similarity between the leaf post and the root post.
Comm_cnt	Number of communication times between the current leaf and the root.
Deg_dist	The probability distribution of each node in the topology network.
If_rp	Whether the leaf node reposts the event.
If_sp	Whether the poster is the source post's author
If_src	Whether the post is the root post.
L_follow_R	Whether the leaf node follows the root node.
Post_cnt	Number of posts that a user has posted.
R_follow_L	Whether the root node follows the leaf node.
Sh_pa	The shortest path between a nodes and the connected nodes.
SL_follow_R	Whether the sub-leaf node follows the root node.

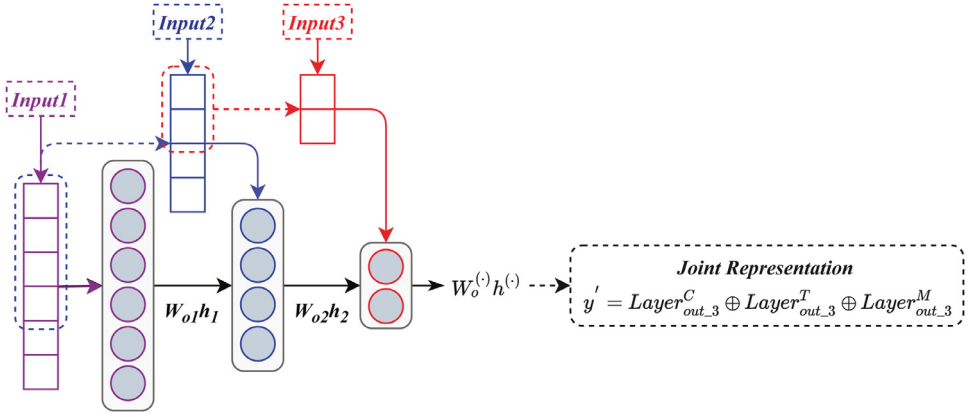

Fig. 3. The diagram of BCMM-GRU.

representation of the topology-based features. Hence, we can obtain the vectorization representation of the topology network as follows:

$$V_{Tf} = \left[V_{Tf\_1}, \ldots, V_{Tf\_k} \ldots, V_{Tf\_12}\right],$$
$$V_T = W_{ts} * \left(V_{Tf} \times T\right) \tag{10}$$

where $*$ is hadamard product, $\times$ represents the out product between V_{Tf} and T.

Thereafter, all the vectorization representations of the post content, metadata and topology network will be fed into the BCMM-GRU mechanism for subsequent processing.

3.3. BCMM-GRU mechanism

As mentioned above, traditional methods need more data and a longer time span for ERD, which may be contrary to the original intent of early detection of rumours. In view of this, we propose a novel BCMM-GRU mechanism who requires less data and a shorter time span for ERD by performing full mapping of the data twice on a three-layer GRU (see Fig. 3). We segment a one-hour dataset into three stages, i.e., data of one hour, 40 min and 20 min respectively (taking post content as an example). Dividing too many stages will increase the time complexity of the experiment and reduce the efficiency of the experiment. The three-stage division can take into account both experimental efficiency and performance based on GRU mechanism. This is why we only divided into three stages. The details are as follows:

(1) For the one-hour dataset, we send the vectorization representation input1, i.e., V_{C_60} into the first-layer GRU for processing (shown in purple in Fig. 3), where the initial vector dimension is equal to the first-layer GRU and the output is $Layer_{out\_1}$, namely first stage.

(2) For data of 40 min, we segment the data of the first 40 min from the one-hour dataset and feed the vectorization representation input2, i.e., V_{C_40} into the second-layer GRU for processing (shown in blue in Fig. 3). The dimension of V_{C_40} is two thirds of the first layer and is equal to the second-layer. At the same time, $Layer_{out\_1}$ makes a full mapping to the second layer, and the second-layer output is $Layer_{out\_2}$, namely second stage.

(3) Similar to the first and second stages, we separate the data of the first 20 min from the 40-minute segment and feed the vectorization representation input3, i.e., V_{C_20} into the third-layer GRU for processing (shown in red in Fig. 3). The dimension of V_{C_20} is half that of the second layer and is equal to the third-layer. $Layer_{out\_2}$ makes a full mapping to the third layer and the third-layer output is $Layer_{out\_3}$, namely third stage.

Finally, we combine $Layer_{out\_3}$ from the post content, topology network and metadata respectively into a joint representation, which is then integrated with the weight vector calculated by the stance recognition for element-wise multiplication to detect rumour and non-rumour. The equations are as follows:

$$Layer_{out\_1} = \sigma\left(W_{o1}h_1\right), \tag{11}$$

where σ is the logistic sigmoid function, W_{o1} is the output weight of the first-layer GRU, h_1 is the output of the first-layer GRU. h_1, i.e., the GRU computation [48,49], is shown as follows:

$$r^t = \sigma\left(W_r\left[h^{t-1}, V_{C\_60}\right] + b_r\right)$$
$$z^t = \sigma\left(W_z\left[h^{t-1}, V_{C\_60}\right] + b_z\right), \tag{12}$$

where r^t is a set of reset gates. When switched off (r^t close to 0), the reset gate effectively makes the unit act as if it is reading the first symbol of V_{C_60}, allowing it to forget the previously computed state. z^t is the update gate that decides how much the unit updates its activation, or V_{C_60}. h^{t-1} is the output of the last unit at time

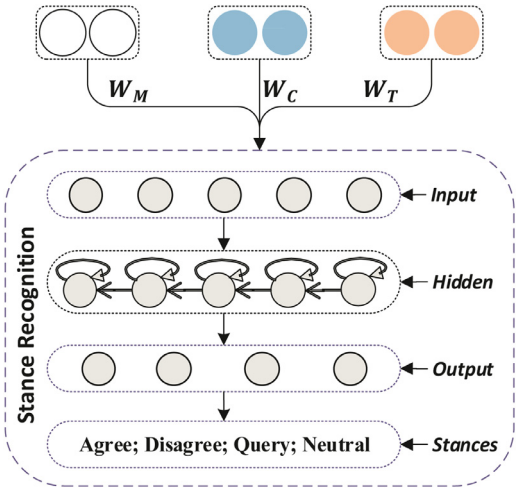

Fig. 4. The diagram of stance recognition.

$t-1$. W_r and W_z represent the weights of the reset gate and update gate respectively while b_r and b_z represent the corresponding bias term.

$$\tilde{h}^t = \tanh\left(W_{\tilde{h}}\left[r^t \odot h^{t-1}, V_{C\_60}\right] + b_{\tilde{h}}\right)$$
$$h_1 = \left(1 - z^t\right) \odot h^{t-1} + z^t \odot \tilde{h}^t, \tag{13}$$

where $\tilde{h}^t$ is the candidate activation which is similar to the traditional recurrent unit, and its corresponding bias term is $b_{\tilde{h}}$.

From Eqs. (11)–(13) and Fig. 3, the computational equations of $Layer_{out\_2}$ and $Layer_{out\_3}$ are as follows:

$$Layer_{out\_2} = \sigma\left(W_{o2} h_2\right)$$
$$Layer_{out\_3} = \sigma\left(W_o^C h^C\right), \tag{14}$$

where W_o^C is the weight of $Layer_{out\_3}$ based on the post content, h^C is the output of the third stage based on the post content.

From Eq. (14) and Fig. 3, the corresponding equation for joint representation $y^{'}$ is as follows:

$$y^{'} = Layer^C_{out\_3} \oplus Layer^T_{out\_3} \oplus Layer^M_{out\_3}, \tag{15}$$

where $Layer^C_{out\_3}$, $Layer^T_{out\_3}$ and $Layer^M_{out\_3}$ correspond to the outputs of post content, topology network and post metadata, respectively. Thereafter, we combine $y^{'}$ and W_S (the weight generated from the stance recognition detailed in Section 3.4) to classify the post into rumour or non-rumour.

3.4. Stance recognition

The user comments or stances to the post are a direct response to the current post. In view of this, we consider the posts' stances (namely "crowd wisdom") to study its influence on the performance of ERD. The overview of the stance recognition is illustrated in Fig. 4.

Based on Figs. 1 and 4, we make the following settings to classify all the posts of an event into four categories, i.e., "disagree", "agree", "query" and "neutral" generated from Ma and Luo [7] and Zubiaga et al. [4], 45]. The community of "disagree" is classified into the rumour category (R_C), the community of "agree", "query" and "neutral" is classified into non-rumour category (NR_C). Then, we can infer the stance weight W_S, which can adjust the joint representation $y^{'}$ for ERD as follows:

$$W_S = \frac{R_C}{R_C + NR_C} \tag{16}$$

Based on Section 3.3 and Fig. 4, the computation of W_S and stance recognition is as follows:

$$y_{n_{in}} = \sum_{1}^{n_{in}} \mathfrak{F}_{in}\left(W_{n_{in}} \mathfrak{M}_{n_{in}} + b_{in}\right), \tag{17}$$

where $y_{n_{in}}$ is output of the n_{in}th neuron in the input layer, computed by the corresponding joint vectorization representation $\mathfrak{M}_{n_{in}}$, where $\mathfrak{M}_{n_{in}} = W_C Layer^C_{out\_3} \oplus W_T Layer^T_{out\_3} \oplus W_M Layer^M_{out\_3}$. n_{in} is the number of neurons in the input layer, $W_{n_{in}}$ represents the weight of the n_{in}th neuron in the input layer, b_{in} is the bias term of input layer, $\mathfrak{F}_{in}(\cdot)$ is the function of computation in the input layer. From Eq. (17), the corresponding equations of the hidden and output layers are as follows:

$$y_{n_h} = \sum_{1}^{n_h} \sum_{1}^{n_{in}} \mathfrak{F}_h\left(W_{n_h} y_{n_{in}} + b_h\right), \tag{18}$$

where y_{n_h} is the output of the n_{in}th neuron in the hidden layer, computed by the corresponding neuron output $y_{n_{in}}$ from the input layer, n_h is the number of neurons in the hidden layer. W_{n_h} represents the weight of the n_hth neuron in the hidden layer, b_h is the bias term of the hidden layer, $\mathfrak{F}_h(\cdot)$ is the function of computation in the hidden layer.

$$y_{n_o} = \sum_{1}^{n_o} \sum_{1}^{n_h} \mathfrak{F}_o\left(W_{n_o} y_{n_h} + b_o\right), 0 \le y_{n_o} \le 1, n_o = 4, \tag{19}$$

where y_{n_o} is the output of the n_oth neuron in the output layer, computed by the corresponding neuron output y_{n_h} from the hidden layer, n_o is the number of neurons in the output layer. W_{n_o} represents the weight of the n_oth neuron in the output layer, b_o is the bias term of the hidden layer, $\mathfrak{F}_o(\cdot)$ is the function of computation in the hidden layer. In addition, we set the number of neuron as $n_o = 4$ to match the number of stances.

Thereafter, based on the result of y_{n_o}, the corresponding four stances (i.e., agree, disagree, query, neutral) and $y_{n_o}(agree) + y_{n_o}(disagree) + y_{n_o}(query) + y_{n_o}(neutral) = 1$, we will select the score that is the highest and is greater than 0.6, otherwise, iteration continues until convergence within 50 iterations. Finally, based on Eqs. (15)–(19), we can give the rumour detection equation y as follows:

$$y = \sigma\left(W_S * y^{'}\right), \tag{20}$$

where $0 < y \le 1$. We define the threshold such that if $y \ge 0.6$, then the current post is a rumour, otherwise it is a non-rumour.

4. Experiments

In this section, we present datasets to evaluate our proposed method and its variants as well as the experimental settings.

4.1. Datasets

Our experiment focuses on four datasets corresponding to emergency events that are crawled from Sina Weibo[1] (seven events), Twitter[2] (six events) and Mixed media (ten events, e.g., Toutiao[3]), as well as part of the dataset in the public PHEME[4] (five events), which are social platforms popular in China or worldwide. Table 4 presents the distribution of all the datasets used in our experiments. Avg_{pm} and Avg_{pe} are the averages of posts

[1] https://open.weibo.com/wiki/API%E6%96%87%E6%A1%A3/en
[2] https://dev.twitter.com/docs
[3] https://open.mp.toutiao.com/#/resource?_k=ct5glh
[4] https://zenodo.org/record/3269768, and we also consider the same dataset applied in [45].

Table 4
Data distribution of four datasets at different stages.

Datasets	Stages	Posts	Rumours	Non-rumours	Replies	Avg_{pm}[a]	Avg_{pe}
Sina Weibo (7 events)	60 min	1565	440	1125	19235	26.1	223.6
	40 min	785	280	505	8049	19.6	112.1
	20 min	346	212	134	4274	17.3	49.4
Twitter (6 events)	60 min	1492	421	1071	18313	24.9	248.7
	40 min	611	269	342	7639	15.3	101.8
	20 min	346	212	134	3970	17.3	57.7
PHEME (part) (5 events)	60 min	591	167	424	7248	9.9	118.2
	40 min	343	146	197	3967	8.6	68.6
	20 min	224	137	87	1735	11.2	44.8
Mixed (10 events)	60 min	2858	805	2053	35135	47.6	285.8
	40 min	1122	401	721	15116	28.1	112.2
	20 min	545	334	211	7307	27.3	54.5

[a] All values are rounded to one decimal place, and the same is true for Avg_{pe}

per minute and per event corresponding to different stages (i.e., 60-minute, 40-minute and 20-minute), respectively. These posts in these events have all been annotated by five journalists or linguists who will give each post a score ($0 < score \leq 1$). Then, we set the average of these scores as *avg_scr*. If $avg\_scr \geq 0.7$, the current post is annotated as rumour label, otherwise it is non-rumour label. In addition, each post contains some replies/comments which represent the personal stances to the current post and these stances can be annotated as the "disagree", "agree", "query" or "neutral".

As we can see from Table 4, one important characteristic of events in each dataset is the uneven distribution in the different stages, i.e., the number of posts shows an upward trend over time. Specially, for Twitter dataset and PHEME (part) dataset, there is an interesting phenomenon that the Avg_{pm} of the 20-minute stage is significantly greater than that in 40-minute stage, which is different from the other datasets. On the contrary, for Avg_{pe}, the number of posts shows an upward trend on each dataset over time. The reason for this may be that, in about 40 min, the popularity of events in Twitter has decreased. In addition, for the number of rumours, its value of 20-minute is greater than that of non-rumours in each dataset without any exceptions. We think the reason for this situation is that in the early stage of the event, the people who are engaged are unable to judge the authenticity of the event because they have little information and thus they would be more credulous of the event and spread (or repost) it. The imbalanced distribution of posts aptly represents the real-world scenario but it poses an increased challenge to ERD. As such, if our proposed method works well for these datasets, it will have great practical significance.

4.2. Settings

To evaluate our proposed method, we adopt four performance metrics, i.e., accuracy (Acc.), precision (P), recall (R) and F1-measure (F1). All experimental results are the average of cross validation folds.

Traditional rumour detection or machine learning methods [7,8] have interest in exploiting conventional K-fold Cross Validation (CV) with different ratios (e.g., training (4):testing (1)) to evaluate the performance of their proposed methods. Thus, we also adopt this CV and set it as 6-fold to evaluate our method. We employ categorical cross-entropy to support our loss function and Adam optimisation algorithm to train the method. We also set the initial vectorization dimension as 200 is less computationally intensive than setting at 500 especially when the performances are similar. In addition, the number of epochs, learning rate and regularisation strength are set by the Tree of Parzen Estimators (TPE) algorithm [50].

4.3. Baselines and Ablation evaluation

Baselines: In order to evaluate our proposed method, we exploit the following state-of-the-art rumour detection techniques as the baselines.

- Ma and Luo [7]: We set 5-fold CV to compare the performance with our proposed method on the PHEME (part) and Mixed dataset with the train/test ratio set as 4:1 and three performance metrics. In addition, this experiment will be evaluated on the full data of 60-minute dataset so similar to Chen et al. [8].
- Chen et al. [8]: Since its model is nested with multiple layers of neurons, we set up 3-fold CV and set the train/test ratio as 4:1 on PHEME (part) and Mixed dataset with three performance metrics to compare it with our method. This setting not only saves the time of the experiment but also yields the most promising result in the current situation.

Ablation evaluation: In addition, we include our proposed method and its variants as comparisons to demonstrate the importance of the individual components for ERD.

- F-BCMM: This is our full settings with BCMM to compare with its variants and the baselines.
- Fc-BCMM: Only post text content is employed as input to evaluate the method. The topology-based and metadata-based features and the related equations are ignored. Similarly, the computation of stance (i.e., W_S) will only consider the post content.
- Fct-BCMM: We consider both the post content and topology network as inputs to validate the method but ignore the metadata-based features.[5]
- Fcm-BCMM: This variant adopts the post content and metadata as inputs to evaluate the method by removing the topology network and related topology-based features.
- F-GRU-Stage1: This variant retains all the components without considering the BCMM. It only employs stage1 (i.e., the full 60-minute dataset) and ignores stage2 and stage3.
- F-GRU-Stage3: This variant is similar to F-GRU-Stage1 in that it only considers stage3 (i.e., only use 20-minute data segment from the 60-minute dataset).

5. Results and analysis

Based on the methodology and settings, the performance of our proposed method is compared to its variants and the baselines. The results are shown in Table 5.

[5] We will not consider topology network alone, because it is dependent on post content, i.e., if there is no post content, topology network cannot be constructed, so is Fcm-BCMM.

Table 5
Comparisons of ERD performance between the baselines/variants and F-BCMM on four datasets. The bold numbers represent the optimal results for each evaluation metric.

Methods	Sina Weibo Dataset				Twitter Dataset			
	Acc.(%)	P (%)	R (%)	F1 (%)	Acc.(%)	P (%)	R (%)	F1 (%)
F-BCMM	**79.98**	79.86	**79.95**	**79.90**	**80.09**	**80.34**	80.01	**80.18**
Fc-BCMM	77.16	76.65	77.4	77.02	77.35	77.20	77.54	77.37
Fct-BCMM	79.97	**80.01**	78.44	79.22	78.25	78.23	**80.02**	79.11
Fcm-BCMM	77.98	77.52	78.53	78.02	78.17	78.07	78.67	78.37
F-GRU-Stage1	76.52	76.14	76.12	76.13	76.71	76.69	76.26	76.47
F-GRU-Stage3	74.87	74.08	74.02	74.05	75.05	74.61	74.15	74.38
Methods	PHEME (part) Dataset				Mixed Dataset			
	Acc.(%)	P (%)	R (%)	F1 (%)	Acc.(%)	P (%)	R (%)	F1 (%)
F-BCMM	**79.96**	**79.91**	79.87	**79.89**	**80.01**	79.85	**79.96**	**79.91**
Fc-BCMM	77.14	76.71	77.34	77.02	77.21	76.88	77.31	77.09
Fct-BCMM	78.04	77.74	78.38	78.06	79.11	**79.86**	79.24	79.55
Fcm-BCMM	77.96	77.58	**79.97**	78.76	78.03	77.75	78.43	78.09
F-GRU-Stage1	76.50	76.20	76.06	76.13	76.57	76.37	76.02	76.19
F-GRU-Stage3	74.85	74.14	73.96	74.05	74.92	74.31	73.92	74.11
Ma and Luo [7]	-	74.09	74.56	74.32	-	74.25	74.60	74.42
Chen et al. [8]	-	76.16	76.09	76.12	-	76.38	76.01	76.19

In general, F-BCMM's performance is superior to its variants and the baselines in terms of metric F1 on the four datasets. The performance of all these methods on Twitter dataset is superior to that on other datasets. In particular, the metric F1 of F-BCMM achieves 80.18% and 79.91% on Twitter dataset and Mixed Dataset respectively, which are superior to the other methods or other datasets. This could be because the Avg_{pm} shows a downward trend between 20 min and 40 min, and it has the least growth over the entire process relative to the other datasets. Most people disagree with this post and think it is a rumour, so there will be relatively few posts about the event. In addition, the metric F1 of F-BCMM on Twitter dataset is better than that on PHEME (part) dataset. The size of the dataset (i.e., Twitter has more posts than PHEME (part)) could be the main reason that limits the performance, which testifies to the fact that machine learning methods improve their performance with increasing size of the dataset. In other words, F-BCMM basically presents the best results on all datasets which means it can achieve good results regardless of the size of the dataset.

For [7], they merely considers comments and words to represent vector semantic which could be the reason that it achieves the lowest performance in the metric F1 compared to our method on the two chosen datasets. They employed PHEME as their dataset, which is the reason that we also perform the experiments on this dataset. To evaluate the performance of our methods, we also choose to perform comparison experiment on the Mixed dataset with most posts. For [8], they applied multi-layer CNN to detect rumour on the dataset of longer time span, which achieves a better result. However, their method cannot sustain the advantage on dataset with shorter time span. In addition, the performance of [7] and [8] is slightly worse than F-GRU-Stage1 on the two chosen datasets, yet their results are better than F-GRU-Stage3. This also confirms its advantage in dataset of long time span, and also highlights its disadvantage in dataset of short time span. On the other hand, it also shows that our method performs well be it dataset of long or short time span.

Aside from F-BCMM, for Fct-BCMM, its performance in the metric F1 is superior to the other models in most datasets except for PHEME (part) dataset. In particular, its metric P achieves 80.01% and 79.86% on the dataset of Sina Weibo and Mixed respectively, which are superior to F-BCMM on the corresponding datasets. This could be due to these two datasets containing the most posts since the more posts there are, the more perfect is the topology network, which will improve the performance of ERD and also confirm the importance of topology network. For Fcm-BCMM, its performance is only slightly worse than Fct-BCMM on most datasets, yet it is superior to other variants of F-BCMM. In terms of the metric R, Fcm-BCMM presents the best result with 79.97% on the PHEME (part) dataset, which is superior to the other methods. This performance improvement comes from the ability of the metadata to further enrich or enhance the vectorization representation of the post content, which will further improve the performance of the experiment. In addition, the performances of Fct-BCMM and Fcm-BCMM are superior to Fc-BCMM, which also shows that it is right to consider the post metadata and the topology network.

For F-GRU-Stage1, its performance is not the worst, but it is at the lower level compared to the other variants. This could be the reason that it ignores the BCMM mechanism. Its performance is superior to F-GRU-Stage3 and [7] on all datasets. Hence, it also shows the advantage of considering the post content, the topology network and the post metadata simultaneously. This leads to a situation is which F-GRU-Stage1 is slightly better at 0.01% compared to [8] in terms of metric F1 on PHEME (part) dataset while it equals [8] on the Mixed dataset. For F-GRU-Stage3, it presents the worst result on all datasets and this could be that its data size is the smallest which compromises the performance of topology structure and thus weakens the semantic representation of the posts and metadata.

In general, our proposed method and its variants achieve the optimal or 2nd highest optimal results, which testifies that BCMM mechanism is useful in improving the performance of ERD.

Thereafter, we investigate the performance of F-BCMM and its variants on Twitter dataset adulterated with different percentage of random noise data. We employ the same settings as those in the experiments with clean data. As we can see from Table 6, mixing noise data into the experiments will decrease the performance of ERD. All evaluation metrics decrease for F-BCMM and its variants with increasing percentage of noise data. In particular, F-BCMM exhibits excellent performance by maintaining good performance even when there is a 10% noise interference. Its performance only degrades by 5.60% in the metric F1 when noise is increased which shows the robustness of F-BCMM.

We notice that F-GRU-Stage1 and F-GRU-Stage3 are more sensitive to noise, clocking 11.77% and 12.53% reduction in the metric F1 when noise is increased from 1% to 10%, respectively. For both of them, the degradation in performance is caused by the reduction in clean data and the absence of the BCMM module further weakens the method's ability to distinguish between clean and noise

Table 6
Performance of F-BCMM variants and itself on Twitter dataset adulterated with different percentage of noise data.

Percent	F-BCMM				Fc-BCMM			
	Acc.(%)	P (%)	R (%)	F1 (%)	Acc.(%)	P (%)	R (%)	F1 (%)
1%	79.21	79.18	79.88	79.53	75.26	75.31	75.46	75.38
3%	79.07	79.05	78.99	79.02	73.91	73.89	74.06	73.97
5%	78.06	78.03	78.63	78.33	72.01	72.06	72.02	72.04
7%	76.94	76.91	77.01	76.96	70.06	70.16	70.22	70.19
9%	74.03	74.82	75.64	75.23	66.99	66.25	67.73	66.98
10%	73.38	73.21	74.66	73.93	65.19	65.14	65.69	65.41
Percent	Fct-BCMM				Fcm-BCMM			
	Acc.(%)	P (%)	R (%)	F1 (%)	Acc.(%)	P (%)	R (%)	F1 (%)
1%	77.17	76.99	77.60	77.29	75.69	75.75	76.52	76.13
3%	75.96	75.75	75.44	75.59	74.76	74.64	74.45	74.54
5%	74.02	73.91	73.93	73.92	72.35	72.31	72.12	72.21
7%	72.89	71.67	71.01	71.34	70.86	70.51	70.62	70.56
9%	70.76	69.61	69.08	69.34	68.16	68.06	67.99	68.02
10%	69.58	68.03	68.79	68.41	67.01	67.07	67.36	67.21
Percent	F-GRU-Stage1				F-GRU-Stage3			
	Acc.(%)	P (%)	R (%)	F1 (%)	Acc.(%)	P (%)	R (%)	F1 (%)
1%	75.65	75.62	75.44	75.53	73.92	74.02	72.86	73.44
3%	73.32	73.43	73.53	73.48	71.26	71.92	70.91	71.41
5%	71.55	71.51	70.96	71.23	68.91	68.85	67.58	68.21
7%	68.06	68.96	67.99	68.47	65.76	65.81	65.66	65.73
9%	65.02	65.92	64.02	64.96	62.22	62.05	62.07	62.06
10%	64.66	64.31	63.21	63.76	60.92	60.86	60.96	60.91

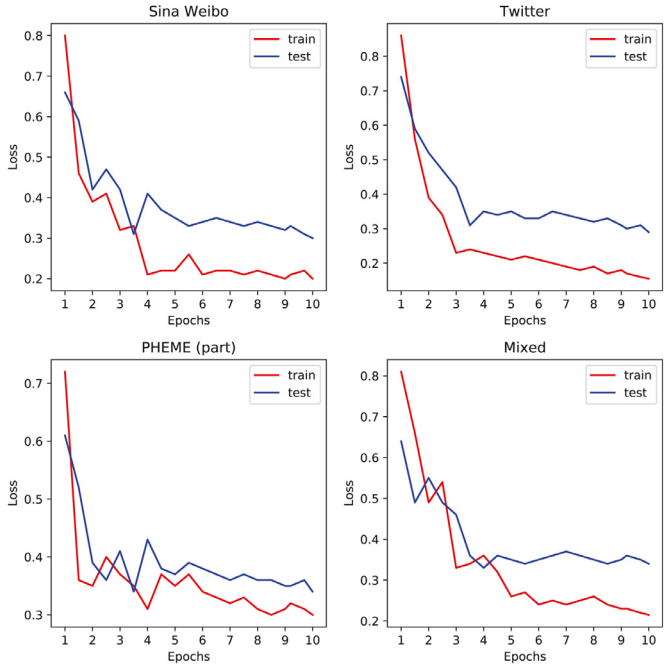

Fig. 5. Training loss with 10 epochs over time during the training of F-BCMM in 6-fold CV.

data. The amount of clean data for F-GRU-Stage3 is lower than F-GRU-Stage1, which could account for the higher degradation in F-GRU-Stage3 than F-GRU-Stage1.

For Fc-BCMM, Fct-BCMM and Fcm-BCMM, they respectively achieve 9.97%, 8.88%, and 8.92% reduction in the metric F1 when noise is increased. Although, they present unsatisfied performance compared to F-BCMM, they are still better than F-GRU-Stage1 and F-GRU-Stage3 on the noise datasets. This shows that the topology and metadata also can strengthen the robustness of our proposed method.

In order to avoid overfitting, we set the number of epoch as 10 to test our method. The results of training loss with 10 epochs over time during the training of F-BCMM in 6-fold CV are shown in Fig. 5.

From our observation, the proposed method shows a continuous downward trend in the first several epochs. Thereafter, it will maintain a relatively stable state and exhibits a tendency of overfitting after the 10th epoch. Training on Twitter dataset shows the best performance where the training loss and testing loss reach a stable state after the 3rd and 4th epochs respectively. For Mixed dataset, it reaches steady state after the 6th and 4th epochs in terms of training loss and testing loss respectively. For Sina Weibo dataset and PHEME (part) dataset, their results are unstable in the early epochs and only stabilize after the 6th and 7th epochs in terms of training loss and testing loss respectively. Hence, the results of Sina Weibo and PHEME (part) are slightly worse than those on Twitter. This is consistent with what is shown in Table 5, i.e., in terms of metric Acc. and metric F1, the results of Twitter are superior to other datasets.

6. Conclusion

In this paper, we have presented a framework for early rumour detection by constructing a BCMM mechanism to enhance posts semantic representation from a short time span, namely F-BCMM. We evaluate the metrics of Accuracy, Precision, Recall and F1-Score on our proposed method and compare it against some its variants and the baselines. The results show that F-BCMM can detect the rumour more effectively and accurately than the rest in the early break out stage of events. F-BCMM achieves superior performance in metric Acc. of 80.09% and metric F1 of 80.18%. It triumphs its variants and some baselines on four datasets with up to 3.77% improvement in the metric F1 on PHEME (part) dataset compared to the baseline.

ERD is often associated with some emergencies, hot issues, etc. which are critical today where OSM permeates all sectors of society and its viral nature can disseminate rumours at astronomical speeds and can sow social unrest and instability. It is therefore crucial that rumours propagating over OSM can be detected as fast

and as accurate as possible in the early stage. Herein lies the role of F-BCMM in providing an accurate and practical framework for ERD leading to the fast and accurate debunking of OSM rumours at an early stage. Researchers working in this area will be able to benefit from F-BCMM to research on the influence of backward compression mapping mechanism on vectorization representation, e.g., short text analysis, short document retrieval, and the rumour detection within a short time span (or small dataset).

One potential limitation of F-BCMM is that we just consider the single-layer social network topology architecture and not the topology of a multi-layer social network (i.e., the relationship of multiple social network communities, even the relationship of online and offline networks). The multi-layer topology will however result in a very significant computational burden and complexity in constructing the model. How to use multi-layer network to improve the experimental performance under the premise of reducing the amount of computation is for further study.

In view of the superior performance of the F-BCM, the future work is to expand the application of the model in other situations while further improving the accuracy of the experiment, and even further shortening the time span to achieve real-time rumour detection.

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.

Acknowledgments

This work was partially supported by the Fundamental Research Funds for the Central Universities [grant number NW2020001]; and the China Scholarship Council [grant number 201906830054].

References

[1] Y. Luo, J. Ma, The influence of positive news on rumor spreading in social networks with scale-free characteristics, Int. J. Mod. Phys. C 29 (09) (2018) 1850078, doi:10.1142/S012918311850078X.
[2] T. Uricchio, L. Ballan, L. Seidenari, A.D. Bimbo, Automatic image annotation via label transfer in the semantic space, Pattern Recognit. 71 (2017) 144–157, doi:10.1016/j.patcog.2017.05.019.
[3] A.I.E. Hosni, K. Li, S. Ahmad, Minimizing rumor influence in multiplex online social networks based on human individual and social behaviors, Inf. Sci. 512 (2020) 1458–1480, doi:10.1016/j.ins.2019.10.063.
[4] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, R. Procter, Detection and resolution of rumours in social media: a survey, ACM Comput. Surv. 51 (2) (2018), doi:10.1145/3161603.
[5] G. Giasemidis, N. Kaplis, I. Agrafiotis, J.R.C. Nurse, A semi-supervised approach to message stance classification, IEEE Trans. Knowl. Data Eng. 32 (1) (2020) 1–11.
[6] M. Lukasik, K. Bontcheva, T. Cohn, A. Zubiaga, M. Liakata, R. Procter, Gaussian processes for rumour stance classification in social media, ACM Trans. Inf. Syst. 37 (2) (2019) 20:1–20:24, doi:10.1145/3295823.
[7] J. Ma, Y. Luo, The classification of rumour standpoints in online social network based on combinatorial classifiers, J. Inf. Sci. 46 (2) (2020) 191–204, doi:10.1177/0165551519828619.
[8] W. Chen, Y. Zhang, C.K. Yeo, C.T. Lau, B.S. Lee, Unsupervised rumor detection based on users' behaviors using neural networks, Pattern Recognit. Lett. 105 (2018) 226–233, doi:10.1016/j.patrec.2017.10.014.
[9] E. Chatzilari, S. Nikolopoulos, I. Patras, I. Kompatsiaris, Leveraging social media for scalable object detection, Pattern Recognit. 45 (8) (2012) 2962–2979, doi:10.1016/j.patcog.2012.02.006.
[10] Y. Zhou, C. Wu, Q. Zhu, Y. Xiang, S.W. Loke, Rumor source detection in networks based on the SEIR model, IEEE Access 7 (2019) 45240–45258.
[11] A. Lozano-Diez, R. Zazo, D.T. Toledano, J. Gonzalez-Rodriguez, An analysis of the influence of deep neural network (DNN) topology in bottleneck feature based language recognition, PLOS ONE 12 (2017) 1–22, doi:10.1371/journal.pone.0182580.
[12] C. Wu, R. Gao, D. Zhang, S. Han, Y. Zhang, PRWHMDA: human microbe-disease association prediction by random walk on the heterogeneous network with PSO, Int. J. Biol. Sci. 14 (8) (2018) 849–857.
[13] Z. Liao, L. Liu, Y. Chen, A novel link prediction method for opportunistic networks based on random walk and a deep belief network, IEEE Access 8 (2020) 16236–16247.
[14] M. Li, H. Gao, F. Zuo, H. Li, A continuous random walk model with explicit coherence regularization for image segmentation, IEEE Trans. Image Process. 28 (4) (2019) 1759–1772, doi:10.1109/TIP.2018.2881907.
[15] T. Leaver, T. Highfield, Visualising the ends of identity: pre-birth and post-death on instagram, Inf. Commun. Soc. 21 (1) (2018) 30–45, doi:10.1080/1369118X.2016.1259343.
[16] G. Fanti, P. Kairouz, S. Oh, K. Ramchandran, P. Viswanath, Hiding the rumor source, IEEE Trans. Inf. Theory 63 (10) (2017) 6679–6713.
[17] M. Fazil, M. Abulaish, A hybrid approach for detecting automated spammers in twitter, IEEE Trans. Inf. ForensicsSecur. 13 (11) (2018) 2707–2719.
[18] M.A. Abebe, J. Tekli, F. Getahun, R. Chbeir, G. Tekli, Generic metadata representation framework for social-based event detection, description, and linkage, Knowl.-Based Syst. 188 (2020) 104817, doi:10.1016/j.knosys.2019.06.025.
[19] M.S. Park, Understanding characteristics of semantic associations in health consumer generated knowledge representation in social media, J. Assoc. Inf. Sci. Technol. 70 (11) (2019) 1210–1222, doi:10.1002/asi.24198.
[20] X. Wang, B. Fang, H. Zhang, X. Wang, Predicting the security threats on the spreading of rumor, false information of facebook content based on the principle of sociology, Comput. Commun. 150 (2020) 455–462, doi:10.1016/j.comcom.2019.11.042.
[21] J. Joo, Z.C. Steinert-Threlkeld, J. Luo, Social and political event analysis based on rich media, in: Proceedings of the 26th ACM International Conference on Multimedia, in: MM'18, Association for Computing Machinery, New York, NY, USA, 2018, pp. 2093–2095, doi:10.1145/3240508.3241470.
[22] R. Zafarani, X. Zhou, K. Shu, H. Liu, Fake news research: theories, detection strategies, and open problems, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, in: KDD'19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 3207–3208, doi:10.1145/3292500.3332287.
[23] A. Bondielli, F. Marcelloni, A survey on fake news and rumour detection techniques, Inf. Sci. 497 (2019) 38–55, doi:10.1016/j.ins.2019.05.035.
[24] A.U. Rehman, A.K. Malik, B. Raza, W. Ali, A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis, Multimed. Tools Appl. 78 (18) (2019) 26597–26613, doi:10.1007/s11042-019-07788-7.
[25] J. Zhao, X. Mao, L. Chen, Speech emotion recognition using deep 1D & 2D CNN lstm networks, Biomed. Signal Process. Control 47 (2019) 312–323, doi:10.1016/j.bspc.2018.08.035.
[26] Y. Luo, J. Ma, C. Li, Entity name recognition of cross-border e-commerce commodity titles based on TWs-LSTM, Electron. Commer. Res. 20 (2) (2020) 405–426, doi:10.1007/s10660-019-09371-6.
[27] G. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, IEEE Trans. Syst. Man Cybern.Part B (Cybernetics) 42 (2) (2012) 513–529.
[28] N.N. Thuy, S. Wongthanavasu, A new approach for reduction of attributes based on stripped quotient sets, Pattern Recognit. 97 (2020) 106999, doi:10.1016/j.patcog.2019.106999.
[29] C.-C. Chang, B.-H. Liao, Active learning based on minimization of the expected path-length of random walks on the learned manifold structure, Pattern Recognit. 71 (2017) 337–348, doi:10.1016/j.patcog.2017.06.001.
[30] H. Zhang, Q. Fang, S. Qian, C. Xu, Multi-modal knowledge-aware event memory network for social media rumor detection, in: Proceedings of the 27th ACM International Conference on Multimedia, in: MM '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 1942–1951, doi:10.1145/3343031.3350850.
[31] H. Yang, Z. Xu, L. Liu, M. Guo, Y. Zhang, Dynamic slide window-based feature scoring and extraction for on-line rumor detection with CNN, in: ICC 2019 - 2019 IEEE International Conference on Communications (ICC), 2019, pp. 1–6.
[32] S. Zheng, H. Zhang, Y. Qi, B. Zhang, Modal-dependent retrieval based on mid-level semantic enhancement space, IEEE Access 7 (2019) 49906–49917.
[33] C. Cai, L. Li, D. Zengi, Behavior enhanced deep bot detection in social media, in: 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), 2017, pp. 128–130.
[34] N.T. Tam, M. Weidlich, B. Zheng, H. Yin, N.Q.V. Hung, B. Stantic, From anomaly detection to rumour detection using data streams of social platforms, Proc. VLDB Endow. 12 (9) (2019) 1016–1029, doi:10.14778/3329772.3329778.
[35] G. Wang, Y. Chi, Y. Liu, Y. Wang, Studies on a multidimensional public opinion network model and its topic detection algorithm, Inf. Process. Manage. 56 (3) (2019) 584–608, doi:10.1016/j.ipm.2018.11.010.
[36] E. Borowski, Y. Chen, H. Mahmassani, Social media effects on sustainable mobility opinion diffusion: model framework and implications for behavior change, Travel Behav. Soc. 19 (2020) 170–183, doi:10.1016/j.tbs.2020.01.003.
[37] C.A. Lavender, A.J. Shapiro, F.S. Day, D.C. Fargo, ORSO (online resource for social omics): a data-driven social network connecting scientists to genomics datasets, PLOS Comput. Biol. 16 (1) (2020) 1–12, doi:10.1371/journal.pcbi.1007571.
[38] Z.Z. Alp, S.G. Öğüdücü, Identifying topical influencers on twitter based on user behavior and network topology, Knowl.-Based Syst. 141 (2018) 211–221, doi:10.1016/j.knosys.2017.11.021.
[39] A.A. Morgan-Lopez, A.E. Kim, R.F. Chew, P. Ruddle, Predicting age groups of twitter users based on language and metadata features, PLOS ONE 12 (8) (2017) 1–12, doi:10.1371/journal.pone.0183537.
[40] S. Kudugunta, E. Ferrara, Deep neural networks for bot detection, Inf. Sci. 467 (2018) 312–322, doi:10.1016/j.ins.2018.08.019.
[41] A. Morales, N. Gandhi, M.S. Chan, S. Lohmann, T. Sanchez, K.A. Brady, L. Ungar, D. Albarracn, C. Zhai, Multi-attribute topic feature construction for social me-

dia-based prediction, in: 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 1073–1078.
[42] A. Pandya, M. Oussalah, P. Monachesi, P. Kostakos, On the use of distributed semantics of tweet metadata for user age prediction, Fut. Gener. Comput. Syst. 102 (2020) 437–452, doi:10.1016/j.future.2019.08.018.
[43] Y. Albalawi, N.S. Nikolov, J. Buckley, Trustworthy health-related tweets on social media in saudi arabia: tweet metadata analysis, J. Med. Internet Res. 21 (10) (2019) e14731, doi:10.2196/14731.
[44] M. Lukasik, P.K. Srijith, D. Vu, K. Bontcheva, A. Zubiaga, T. Cohn, Hawkes processes for continuous time sequence classification: an application to rumour stance classification in twitter, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Berlin, Germany, 2016, pp. 393–398, doi:10.18653/v1/P16-2064.
[45] A. Zubiaga, E. Kochkina, M. Liakata, R. Procter, M. Lukasik, K. Bontcheva, T. Cohn, I. Augenstein, Discourse-aware rumour stance classification in social media using sequential classifiers, Inf. Process. Manage. 54 (2) (2018) 273–290, doi:10.1016/j.ipm.2017.11.009.
[46] Z. Zhang, D. Chen, Z. Wang, H. Li, L. Bai, E.R. Hancock, Depth-based subgraph convolutional auto-encoder for network representation learning, Pattern Recognit. 90 (2019) 363–376, doi:10.1016/j.patcog.2019.01.045.
[47] J. Lin, Q. Kong, W. Mao, L. Wang, A topic enhanced approach to detecting multiple standpoints in web texts, Inf. Sci. 501 (2019) 483–494, doi:10.1016/j.ins.2019.05.068.
[48] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder–decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, 2014, pp. 1724–1734, doi:10.3115/v1/D14-1179.
[49] J. Chung, C. Gülcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, CoRR (2014) abs/1412.3555.
[50] J.S. Bergstra, R. Bardenet, Y. Bengio, B. Kégl, Algorithms for hyper-parameter optimization, in: J. Shawe-Taylor, R.S. Zemel, P.L. Bartlett, F. Pereira, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 24, Curran Associates, Inc., 2011, pp. 2546–2554.

Yongcong Luo is a doctoral candidate in the Department of management science and engineering at Nanjing University of Aeronautics and Astronautics (NUAA). He is currently a visiting student at Nanyang Technological University (NTU). His research interests include data analysis, machine learning, and social media.

Jing Ma is a professor in the Department of Management Science and Engineering at NUAA. She received the Ph.D. degree in management science and engineering from NUAA in 2008. Her research interests include machine learning, electronic commerce, complex network, and data mining.

Chai Kiat Yeo received the B.Eng. (Hons.) and M.Sc. degrees both in electrical engineering, from the National University of Singapore and the Ph.D. degree from the School of Electrical and Electronics Engineering, NTU, Singapore. She was an Assistant Principal Engineer with Singapore Technologies Electronics and Engineering Limited prior to joining NTU in 1993. She is an Associate Professor and was the Deputy Director of Centre for Multimedia and Network Technology (CeMNet) and the Associate Chair (Academic) with the School of Computer Science and Engineering, NTU. She is currently the Deputy Director and Programme Lead of Singtel Cognitive and Artificial Intelligence Lab for Enterprises@NTU. Her current research interests include anomaly detection, machine learning, artificial intelligence, predictive operational analytics, ad hoc and mobile networks.