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.
SemSeq4FD: Integrating global semantic relationship and local sequential order to enhance text representation for fake news detection

Yuhang Wang, Li Wang, Yanjie Yang, Tao Lian

PII: S0957-4174(20)30846-0
DOI: https://doi.org/10.1016/j.eswa.2020.114090
Reference: ESWA 114090

To appear in: Expert Systems With Applications

Received date: 12 July 2020
Revised date: 17 September 2020
Accepted date: 2 October 2020

Please cite this article as: Y. Wang, L. Wang, Y. Yang et al., SemSeq4FD: Integrating global semantic relationship and local sequential order to enhance text representation for fake news detection. Expert Systems With Applications (2020), doi: https://doi.org/10.1016/j.eswa.2020.114090.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2020 Published by Elsevier Ltd.
SemSeq4FD: Integrating Global Semantic Relationship and Local Sequential Order to Enhance Text Representation for Fake News Detection

Author names: Yuhang Wang, Li Wang, Yanjie Yang, Tao Lian

Authors’ affiliations: Data Science College, Taiyuan University of Technology, Jinchong, Shanxi, 030600, China

Authors’ e-mails: a1098815289@qq.com (Yuhang Wang); bwangli@tyut.edu.cn (Li Wang); c1298960582@qq.com (Yanjie Yang); dlianta@tyut.edu.cn (Tao Lian)

Corresponding author name: Li Wang

Postal address: Data Science College, Taiyuan University of Technology, Jinchong, Shanxi, 030600, China

E-mail address: wangli@tyut.edu.cn

Telephone number: (+86) 13934041125
Highlights

SemSeq4FD: Integrating Global Semantic Relationship and Local Sequential Order to Enhance Text Representation for Fake News Detection

Yuhang Wang, Li Wang, Yanjie Yang, Tao Lian

- An enhanced text representation model for content-based early fake news detection.
- Global semantic relations between sentences modeled with graph convolution network.
- Local sentence representation extracted by applying 1D convolution on local context.
- Document representation obtained by incorporating the sentence order through LSTM.
- Generalization ability verified on cross-source and cross-domain datasets.
SemSeq4FD: Integrating Global Semantic Relationship and Local Sequential Order to Enhance Text Representation for Fake News Detection

Yuhang Wang, Li Wang*, Yanjie Yang and Tao Lian

Data Science College, Taiyuan University of Technology, Jinzhong, Shanxi, China

ARTICLE INFO

Keywords:
Fake news detection
Enhanced text representation
Global semantic relationship
Local sequential order
Graph neural network

ABSTRACT

The wide spread of fake news has caused huge losses to both governments and the public. Many existing works on fake news detection utilized spreading information like propagators profiles and the propagation structure. However, such methods face the difficulty of data collection and cannot detect fake news at the early stage. An alternative approach is to detect fake news solely based on its content. Early content-based methods rely on manually designed linguistic features. Such shallow features are domain-dependent, and cannot easily be generalized to cross-domain data. Recently, many natural language processing tasks resort to deep learning methods to learn word, sentence, and document representations. In this paper, we propose a novel graph-based neural network model named SemSeq4FD for early fake news detection based on enhanced text representations. In SemSeq4FD, we model the global pair-wise semantic relations between sentences as a complete graph, and learn the global sentence representations via a graph convolutional network with self-attention mechanism. Considering the importance of local context in conveying the sentence meaning, we employ a 1D convolutional network to learn the local sentence representations. The two representations are combined to form the enhanced sentence representations. Then a LSTM-based network is used to model the sequence of enhanced sentence representations, yielding the final document representation for fake news detection. Experiments conducted on four real-world datasets in English and Chinese, including cross-source and cross-domain datasets, demonstrate that our model can outperform the state-of-the-art methods.

*Corresponding author

wangli@tyut.edu.cn (L. Wang)
ORCID(s): 0000-0002-7385-1426 (L. Wang)
1. Introduction

With the fast advances of Internet techniques, online news platforms have become a popular means for people to obtain and share information. However, due to the convenience and low cost of information dissemination, a large amount of fake news has emerged, causing many detrimental effects on politics, economy and social security. For instance, the fake news that Barack Obama was injured in an explosion wipes out 130 billion dollars in stock (Rapoza, 26 February 2017). After the outbreak of COVID-19 epidemic, various fake news has been spreading over social media (Kouzy, Abi Jaoude, Kraitem, El Alam, Karam, Adib, Zarka, Traboulsi, Akl & Baldour, 2020), which causes panic and misunderstanding. Some people deliberately imitate real news to create fake news. For ordinary people, it is hard to distinguish fake news from the real ones. Therefore, it is vital to develop automatic fake news detection methods.

Existing works can be classified into three types: propagation structure-based (Liu, Yu, Wu & Wang, 2018; Liu & Wu, 2018), user information-based (Shu, Wang & Liu, 2018), and news content-based methods (Zhou, Jain, Phoha & Zafarani, 2019). Propagation structure-based methods extract news transmission features from social network, which contain the information about how people retweet or reply this news in news dissemination process. The user information-based methods focus on the crowds which participate in news life cycle, including the people who publish the news, forward news and comment on news. A collection of features from user profile have been utilized, such as description, gender, followers, friends, location and verified type, which is known to be an essential information to obtain credibility feature (Yang, Shu, Wang, Gu, Wu & Liu, 2019). Nevertheless, the approaches based on propagation structure and user information are limited by the data missing, data noise problem as well as the difficulty of data collection. Researchers must follow the route of fake news transmission and unceasingly capture its related behaviors. In contrast, using news content is a more straightforward and convenient way for detecting fake news than other methods mentioned above, especially in the early stage, and avoids the need to collect propagators’ information and the propagation structure. In this paper, we seek to effectively judge the veracity of news solely based on its content. Most content-based methods detect fake news by extracting linguistic features. Pérez-Rosas, Kleinberg, Lefèvre & Mihalcea (2018) employed vocabulary, syntax, and semantic features to distinct fake news. Zhou et al. (2019) proposed a theory-driven early detection model, which extracts features like Bow and POS tags from vocabulary and discourse level. However, these approaches only rely on feature engineering and have shortages of high-level representations and content structure utilization. Hence, we adopt the text representation learning approach. To obtain enhanced text representations for fake news detection, we especially take into account the content structure—both global semantic relationship and local sequential order among sentences in a news document. On the one hand, modeling global semantic relations among sentences in the entire document is helpful for fully understanding the news (Vaibhav, Mandyam & Hovy, 2019). Key sentences located far off in the document may have close semantic relations and these sentences convey the main idea together. On the other hand, local sequential order between consecutive sentences also makes a difference. It may have certain logic, such as causal, contrastive, and adversative relations. Switching the order may result in different meanings. Besides, the global sequential order is also important for expressing the information of entire document. Based on the above motivations, we build an end-to-end model named SemSeq4FD for early fake news detection based on enhanced text representations. We first utilize the LSTM network to encode individual sentences in a news document through word vectors constituting them. Then, the graph convolutional network with self-attention is exploited to capture global semantic relations among far-off sentences in a news document and stress the importance of different sentences through the attention mechanism. And the 1D convolutional neural network is adopted to model local sequential order between consecutive sentences. The representations learned from these networks are fused to form the enhanced sentence representation. Finally, we feed the enhanced sentence representations into the LSTM-based network sequentially, and obtain the informative document representation by max-pooling, which is further used for fake news detection. We focus on cross-source and cross-domain datasets to verify the effectiveness of our model. And we only rely on text content information to achieve early accurate classification of fake news. The main contributions of our work can be summarized as follows:

- We propose a content-based fake news detection model, named as SemSeq4FD, to take into account both global semantic relationship and local sequential order jointly.
- We leverage self-attention graph convolutional network to obtain global semantic relevance among sentences within document, and we apply 1D convolutional neural network to learn the local sequential order relationship.
We concatenate the features learned by each network to express the enhanced sentence representation.

- We introduce the global sequential order and utilize LSTM-based network to generate document representation based on the enhanced sentence representation.

- We conduct extensive experiments on four real-world datasets in English and Chinese, including cross-source and cross-domain datasets, demonstrating that our model can outperform the state-of-the-art methods.

Traditional models have poor adaptability in the cross-domain tasks. They often evaluate models with test sets that share the same source and domain knowledge as the training set. Classifiers trained in this way have data dependence and are easily affected by noise. Different media sources have different content generators and focus on diverse topics or fields. In this work, we utilize the structure features to learn the generalizable representations of each news. We show that our proposed model could detect fake news in cross-source and cross-domain datasets.

The rest of this paper is organized as follows. In Section 2, we briefly review the related work on fake news detection task. Section 3 describes the details of our proposed model. In Section 4 we conduct a series of experiments on four real-world datasets in English and Chinese, which contain cross-source and cross-domain datasets to evaluate the effectiveness of the proposed SemSeq4FD model. Finally, we conclude this paper and shed light on our feature work.

2. Related Work

Most of the previous work used information other than news content to identify fake news, such as reply news or comments associated with the news article (Wu, Yang & Zhu, 2015; Ma, Gao, Mitra, Kwon, Jansen, Wong & Cha, 2016; Liu & Wu, 2018), context information (Ma, Gao, Wei, Lu & Wong, 2015), time patterns, sources (Shin, Jian, Driscoll & Bar, 2018), user profiles (Yang et al., 2019; Rath, Gao, Ma & Srivastava, 2017; Shu, Wang & Liu, 2018), and a combination of these features (Ruchansky, Seo & Liu, 2017; Vosoughi, Mohsenvand & Roy, 2017) etc. Despite the success of aforementioned works, the additional data they need will undoubtedly increase the difficulty and workload of fake news detection task.

Content-based research can directly judge fake news without auxiliary information, which is more conducive to discovering fake news early. Typical approaches relying on content consist of two types: Linguistic features-based methods and structure features-based methods.

2.1. Linguistic Features-based Methods

Linguistic features-based methods generally detect fake news from words, sentences and documents level. It can be roughly divided into two types: machine learning and deep learning. Horne & Adali (2017) proposed a SVM-based method with 3 broad categories features: stylistic, complexity, and psychological. This study aimed at figuring out the content stylistic differences between fake and real news. Similarly, Pérez-Rosas et al. (2018) extracted handcrafted features from news, and built combined feature sets to train linear SVM model. Recently, Ozbay & Alatas (2020) presented a two-step method for fake news detection, which first conducts text mining and then applied twenty-three supervised artificial intelligent classification methods to the news datasets. Besides, the metaheuristic algorithms could also be used for fake news detection problem. They are general-purpose solution search methods and are able to give optimum solution to the problem at an acceptable space cost. Ozbay & Alatas (2019) utilized the metaheuristic algorithms (such as Grey Wolf Optimizer and Salp Swarm Algorithm) as search methods for fake news detection and obtained promising results. The above methods could indeed solve the fake news detection problem to a certain extent, but they require manual pre-processing and artificially designed feature extraction, which are cumbersome and labor-intensive.

To improve the efficiency and detect fake news automatically, Wang (2017) developed a deep learning-based method, which exploits CNN and BiLSTM to detect fake news in word level. Volkova, Shaffer, Jang & Hodas (2017) evaluated news authority within a fusion of linguistic cues and news word embedding by CNN and LSTM. These methods only focus on local word features. Some researchers have studied fake news recognition at the sentence level and document level based on deep learning. Yu, Liu, Wu, Wang & Tan (2017) proposed a CNN model to learn high level features for each group of posts via paragraph embeddings. Ahn & Jeong (2019) utilized the pre-training BERT model to judge fake news on sentence-level.

Compared with algorithms in word level, methods based on sentence and document level have better detection effect. However, there are still several limitations. Firstly, these methods cannot solve the problem of long-distance
dependence of documents. The positions of semantic related sentences may not be close in the document. The existence of long-distance dependent structures makes it difficult for model to capture global semantic information of the entire document. Secondly, these algorithms have poor generalizability and are prone to overfitting. Fake news spread in various sources of the network. These articles with different source and domain have distinct language features and clues. Models must classify news without relying on this specific information. These algorithms mentioned above usually cannot adapt to unseen news from diverse source and domain due to the overfitting problem.

2.2. Structure Features-based Methods

Structure-based fake news detection methods include tree structure and graph structure. Recently, Uppal, Sachdeva & Sharma (2020) created a discourse dependency tree structure to implement automatic deception detection. Nevertheless, graph structure has stronger ability to express information than other structures. Since graph convolutional neural networks (GCN) (Kipf & Welling, 2017) apply deep neural networks to graph structure data and perform well, some studies use GCN to define fake news detection problems as node classification tasks (Wei, Xu & Mao, 2019) and entire graph classification tasks (Vaibhav et al., 2019). Constructing a graph structure has two methods depending on the modeling object. One is to use external information such as the propagation structure and the user information, and the other is to simply use text content. Most of the existing studies compose graph using external data. Wei et al. (2019) established a hierarchical multi-task learning framework, which uses the forwarding structure of tweets as the composition relationship among texts and utilizes the stances of tweets to help identify fake news based on graph convolutional neural network. In addition, Hu, Ding, Qi, Wang & Liao (2019) exploited users profile information as news multi-relationship to build graph structures and proposed a multi-depth GCN model that can detect fake news by aggregating multi-hop neighbor information.

Few studies have considered the sentence relations in a news document. Vaibhav et al. (2019) proposed a graph neural network model for fake news detection, which models the semantic relations among all pairs of sentences in a piece of news. However, they ignore the sentence order within a document, which make a difference for comprehending the meaning of each sentence and conveying the main thought of the news. We consider both global semantic relations between far-off sentences and local sequential order between consecutive sentences for fake news detection based on its content.

3. Proposed Method

Consider a news document consisting of \( n \) sentences \( D = \{ S_i \}_{i=1}^{n} \), where each sentence \( S_i \) is a set that contains \( T_i \) words, i.e., \( S_i = \{ w_{i1}, w_{i2}, \ldots, w_{iT_i} \} \). We define the fake news detection as the binary classification problem. Each news document is associated with a label \( y \in \{ 0, 1 \} \) indicating whether it is fake or not. Thus, fake news detection task can be seen as to learn a function \( f : D \rightarrow \mathcal{Y} \). We seek to perform fake news detection based solely on its content.
Figure 1 depicts the overall framework of the proposed model SemSeq4FD. It consists of three modules. (1) Sentence Encoding: Each sentence in the news document is encoded by using a LSTM network, whose inputs are the sequence of word vectors constituting the sentence, yielding the primary sentence representation. (2) Sentence Representation: The sentence representations are enhanced by taking into account both global semantic relations between far-off sentences and local sequential order between consecutive sentences. The graph convolutional network with self-attention is exploited to capture global semantic relations among far-off sentences. We utilize the 1D convolutional neural network to model local sequential order between local context. The representations learned from the two networks mentioned above are concatenated to form the enhanced sentence representation. (3) Document Representation: The document representation is obtained by feeding the enhanced sentence representations into a LSTM network one by one and max-pooling the hidden states, which are finally fed into a softmax classification layer.

3.1. Sentence Encoding

We use the LSTM encoder to map variable-length sequence of word embeddings into fixed-length sentence embedding, which takes the word sequential order into account. Specifically, consider a sentence $S_i$ composed of $T_i$ words. We first get individual word vectors for each word in the sentence. They are fed into a LSTM network one by one from word $t_i^1$ to $t_i^{T_i}$. The hidden state in the last step $h_{T_i}^i \in \mathbb{R}^m$ can be treated as the sentence vector $s^i$, which is given by

$$h_{T_i}^i = \text{LSTM} \left( w_{t_1}^i, w_{t_2}^i, \ldots, w_{t_{T_i}}^i \right).$$

We obtain the sentence vector $s^i$ encoded by LSTM encoder, i.e., $s^i = h_{T_i}^i$. A sequence of sequential and fixed-length representations for sentences are generated by this way, i.e., $s^1, s^2, \ldots, s^n$. In the following, we refer to these primary sentence representations as the primary feature matrix of sentences $X^{(0)} \in \mathbb{R}^{n \times m}$.

3.2. Sentence Representation

In this section, we use two neural networks to enhance the sentence representations. Firstly, we utilize a graph convolutional network with self-attention to learn sentence representations that incorporate global semantic relations among any pair of sentences in the news document. Secondly, we model the local sequential order between consecutive sentences by using a 1D convolutional neural network.

3.2.1. SA-GCN: Graph Convolutional Network with Self-Attention

We define the semantic relevance graph as $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of graph nodes and $\mathcal{E}$ is the set of graph edges. Each node in set $\mathcal{V}$ is represented by the sentence vector. Here we transform the edge set $\mathcal{E}$ to a adjacency matrix $A \in \mathbb{R}^{m \times m}$ of the complete graph which is fully connected and takes the form of all 1 with 0 on the diagonal. This structure solves the problem of long-distance dependencies between sentences, because each sentence can establish contact with all the other sentences.

Semantic Information Propagation: We use GCN (Kipf & Welling, 2017) based on the semantic relevance graph $G$ to model the semantic data and enhance the sentence representation. We define the feature matrix composed of all sentence feature vectors as $X^{(0)} \in \mathbb{R}^{n \times m}$, and the adjacency matrix as $A \in \mathbb{R}^{m \times m}$. The graph convolution operation is given by:

$$X^{l} = \tanh \left( AX^{(l-1)}W_s \right)$$

In the above equation, $A$ is the adjacency matrix. $X^{(l-1)} \in \mathbb{R}^{n \times m}$ is the sentence representations in the $(l-1)$-th layer. $W_s \in \mathbb{R}^{m \times m}$ is the weight matrix shared across all sentences, where $m'$ is the dimensionality of output node embeddings. This operation learns representations by aggregating the information of neighbor sentences. The output in the highest layer is the enhanced sentence representations $X^{(l)} \in \mathbb{R}^{n \times m'}$ in the $l$-th layer.

Important Sentence Attention: Consider that not all sentences contribute equally to the expression of fake news comprehension. However, each sentence has different importance. Some sentences only play a transitional role, and some sentences contain the core content. In order to allow SemSeq4FD model to focus on important sentences of the article, we leverage the self-attention mechanism (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser & Polosukhin, 2017) to learn the weights to measure the importance of each sentence. Formally, the scaled dot-product
attention is adopted:

$$X_s = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V$$

where, $X_s$ contains the importance of each sentence to the entire document. The formula uses the self-attention mentioned in (Vaswani et al., 2017) and sets $Q = K = V = X^{(0)}$. $d$ is a scaling factor that prevents the molecular dot product value from being too large, and which value is the dimension of sentence. $X_s \in \mathbb{R}^{n \times m'}$ denotes the output of the module SA-GCN, that is, the enhanced sentence representations, which fuses information from other sentences in the document in a weighted manner.

### 3.2.2. Text-CNN: 1D Convolutional Neural Network

We use the sentence level 1D convolutional neural network (Kim, 2014) to capture the local order between consecutive sentences. The input is the primary sentence representation generated by the first module. That is, $X^{(0)} \in \mathbb{R}^{n \times m} = [s^1; s^2; \ldots; s^n]$. Then we define a filter $w \in \mathbb{R}^{k \times m}$, in which $k$ is the number of sentences in a window. A total of $m'$ convolution filters are used. By moving the sliding window of size $k$ from the first to the last sentence and applying all the filters to each window, we can get the feature map $Xa \in \mathbb{R}^{n \times m'}$, which enhances the representation of each sentence by incorporating sentences before and after it when $k = 3$. The PAD used here is the padding operation, which avoids data loss during convolution. We set it to 1 so that the first and last sentences can also participate in the convolution process.

**Fusion Strategy** We leverage concatenation operation to integrate two types of sentence-level representations generated by the above networks, which contain global semantic relationship and local sequential order respectively, i.e., $X_s$ and $Xa$. We get the joint representation $Se \in \mathbb{R}^{n \times 2m'}$ as the enhanced sentence representation.

$$Se = Xs \oplus Xa$$

where $\oplus$ is the concatenation operation.

The pseudocode of the algorithm for learning enhanced sentence representation is described in Algorithm 1.

### 3.3. Document Representation

After going through the Sentence Representation Module, we can get the enhanced sentence representations for each sentence. In this section, we use these enhanced sentence-level representations to generate the entire document-level representation for fake news classification.

Since LSTM network is able to learn contextualized representations, it can fully capture the global sequential order information of sentences in the document, so we use LSTM to obtain the document representation. Meanwhile, as LSTM has a forget gate $f$, an input gate $i$ and an output gate $o$, it can retain historical information and prevent valuable information loss during the learning process. Suppose the network has $n$ time steps. At the $t$-th time step, LSTM updates its unit state as follows.

$$f_t = \sigma(W_f [h_{t-1} \oplus Se_t] + b_f)$$

$$i_t = \sigma(W_i [h_{t-1} \oplus Se_t] + b_i)$$

$$\tilde{c}_t = \tanh(W_c [h_{t-1} \oplus Se_t] + b_c)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$o_t = \sigma(W_o [h_{t-1} \oplus Se_t] + b_o)$$

$$c_t = c_t \odot o_t$$

$$s_t = c_t \odot o_t$$

$$h_t = c_t$$

Wang et al.: Preprint submitted to Elsevier
Algorithm 1: The algorithm for enhanced sentence representation learning

Input: Semantic relevance graph $G = (\mathcal{V}, \mathcal{E})$

The primary sentence embeddings $s_i$ for node $i, i \in \mathcal{V}, \mathcal{V} = \{1, 2, ..., n\}$

Output: The enhanced sentence representations $\mathbf{S_e}$

/* Graph Convolutional Network */
1 Get the primary feature matrix of sentences $\mathbf{X}(0) = [s_1; s_2; ...; s_n]$

foreach node $i \in \mathcal{V}$ do
  2 Find the neighbor nodes set $\text{neighbor}_i$
  foreach node $j \in \text{neighbor}_i$ do
    3 // Aggregate the neighbor node embedding
    4 Update node $i$ embedding $s_i$ by Equation (2)
  end
end

6 The feature matrix $\mathbf{X}(1) = [s_1; s_2; ...; s_n]$ learned by Graph Convolutional Network

/* Self-Attention Mechanism */
7 Calculate Self-Attention value with $\mathbf{X}(1)$ by Equation (3)

/* Text-CNN */
8 foreach convolution filters $w_i, i = 1, \ldots, m'$ do
  9 Applied $w_i$ to each possible window of sentences, window size is $k \times m$
  foreach sentence $j, j = 1, \ldots, n$ do
    10 Generate each sentence representation $\mathbf{X_a}_i$ with contextual information by convolution operation
    // Move window to cover the next $k$ sentences
  end
end

13 The feature matrix of sentence representations named $\mathbf{X_a}$ learned by Text-CNN

/* Fusion Strategy */
14 Concatenate the two feature matrices $\mathbf{X_s}$ and $\mathbf{X_a}$ to get the complete enhanced sentence representation $\mathbf{S_e}$ by Equation (4)

return $\mathbf{S_e}$

$h_t = o_t \odot \tanh (c_t) \quad (10)$

$W_f, W_i, W_c, W_o \in \mathbb{R}^{m' \times 3m'}$ are weight matrices. $b_f, b_i, b_c, b_o \in \mathbb{R}^{m'}$ are bias vectors. $\sigma$ is a sigmoid function. $\odot$ is the element-wise multiplication. $h_{t-1} \in \mathbb{R}^{m'}$ is the output of $t-1$ time step and $h_t \in \mathbb{R}^{m'}$ is the output of the current state. In the whole process, the forget gate $f_t$ first selectively filters out the information stored at the previous state. Secondly, the input gate $i_t$ decides which new information to be updated. Then, $c_t$ is the unit status updated by forgetting historical information and adding new information $\tilde{c}_t$. It replaces the old state value with the new state value. Finally, the output gate $o_t$ determine how much information will be output. The current state output $h_t$ is obtained by filtering information through $o_t$.

We combine hidden vectors trained at each time step and apply a max-pooling layer to each dimension.

$z = \max [h_1; h_2; \ldots; h_n] \quad (11)$

where $z$ represents the final representation of a document, which is then fed into a softmax layer.

$f = \text{softmax} (W_z z + b_f) \quad (12)$

$\hat{y} = [\hat{y}_0, \hat{y}_1]$ is the predicted probability. We minimize the loss function calculated by the cross-entropy criterion.

$L = -y \log (\hat{y}_1) - (1 - y) \log (1 - \hat{y}_0) \quad (13)$
Algorithm 2: The algorithm for enhanced document representation learning

Input: The enhanced sentences representations $S_e$ learned by Sentence Representation Module
Output: The document representation $z$

/* Document Representation Module */

// Utilized LSTM to consider the historical sentence information learned from each time step and generate the final document representation.

foreach time step $t$, $t = 1, \ldots, n$ do

Decide which old information to be forgotten by Equation (5)
Decide which new information to be updated by Equation (6)

Update the unit status and obtained the current state output $h_t$ by Equation (7) to (10)

end

Combine hidden vectors trained at each time step $[h_1; h_2; \ldots; h_n]$  

Generate the enhanced document representation $z$ by Equation (11)

return $z$

$\theta$ is the parameter of the entire model. $y \in \{0, 1\}$ is the ground-truth label.

The pseudocode of the algorithm for learning enhanced document representation is described in Algorithm 2.

4. Experiments

To test the effectiveness of our method, we investigate the following research questions:

RQ1 Can our model enhance fake news classification performance both in cross-source and in cross-domain in English and Chinese datasets?

RQ2 Does the way we encode local sequential order information contribute to the classification performance?

RQ3 What is the effect of LSTM, which is used to model the global sequential order information in the process of learning entire document-level representations for improving the fake news detection performance?

RQ4 Whether the model can show stable prediction ability on small samples?

RQ5 How does our model perform on the Net Reclassification Improvement metric?

4.1. Datasets

We perform fake news detection experiments on four real-world datasets, including two English datasets: LUN and SLN and two Chinese datasets: Weibo and RCED. Each dataset has a collection of news from news publishers or social media. We employ these datasets for the following reasons: (1) We believe that news content in two languages fake news corpora differs greatly in terms of characteristics, expression and rhetorical modes due to the variance of social cultural and cognitive systems. We need to test the effectiveness of our model on different language datasets. (2) Meanwhile, fake news detection methods are generally limited to the domain or source they have been trained on, and it is difficult for them to generalize to other testing data from various publication source and knowledge domain contain completely different lexical features. We hope to verify the generalization capabilities and effectiveness of our model with multiple publication sources and various knowledge domain datasets. (3) We want to avert or reduce the data-dependence of our model as much as possible, so that the model can have theoretical and practical value.

The details of datasets are listed as follows:

4.1.1. English datasets

We use two well-known fake news datasets: LUN (Rashkin, Choi, Jang, Volkova & Choi, 2017)\(^1\) and SLN (Rubin, Conroy, Chen & Cornwell, 2016)\(^2\). Specific statistics are shown in Table 1. The LUN-train dataset contains approximately 24k news from the Onion and the Gigaword news excluding APW\(^3\), WPB\(^4\) sources, and the LUN-test has 1.5k

\(^1\)The LUN dataset can be obtained from https://homes.cs.washington.edu/~hrashkin/fact_checking_files/newsfiles.tar.gz
\(^2\)The SLN dataset can be obtained from http://victoriarubin.fims.uwo.ca/news-verification/data-to-go/
\(^3\)“APW” is the abbreviation of “Associated Press Worldstream”
\(^4\)“WPB” is the abbreviation of “Washington Post/Bloomberg Newswire service”
Table 1
Descriptive statistics of the English datasets

| Veracity | LUN-train | LUN-test | SLN |
|----------|-----------|----------|-----|
| Real     | 9,995     | 750      | 180 |
| Fake     | 14,047    | 750      | 180 |
| Total    | 24,042    | 1500     | 360 |

Table 2
Descriptive statistics of the Weibo dataset

| Veracity | Training(6785) | Test(515) |
|----------|----------------|-----------|
|          | Health | Economic | Technology | Entertainment | Society | Military | Political | Education |
| Real     | 562    | 127      | 12         | 207           | 2328    | 17       | 127       | 86        |
| Fake     | 1614   | 166      | 25         | 390           | 1354    | 33       | 88        | 164       |
| Total    | 2176   | 293      | 37         | 597           | 3682    | 50       | 215       | 250       |

news from the Gigaword news (only APW, WPB sources), the Borowitz Report and the Clickhole sources. LUN-test is completely different from the source of LUN-train. We call such datasets with different publication sources as the cross-source datasets. The SLN dataset contains 360 news from the Toronto Star, the NY Times, the Onion and the Beaverton sources. It has only one news source common with the LUN-train dataset. In (Vaibhav et al., 2019), the SLN dataset is considered to be an out of domain test set. We follow this paper and refer to the SLN dataset as the cross-domain dataset. Following (Vaibhav et al., 2019), we also use LUN-train as the training dataset. To verify the cross-source capability of our model, we employ LUN-test as a cross-source testing dataset. And to test the model’s cross-domain generalization capability, we set the SLN as a cross-domain testing dataset.

4.1.2. Chinese datasets
Our Weibo dataset used in experiments is available on the "Internet fake news detection during the epidemic" competition held by CCF Task Force on Big Data 5. This dataset contains 3 kinds of news across 8 domains, including health, economic, technology, entertainment, society, military, political and education. Because our goal is to evaluate whether the news is true or false, we only choose the real and fake news to do the binary classification task. After data cleaning, the dataset consists of 7300 news articles in all, with 3466 labeled real and 3834 labeled fake. Tabel 2 shows the statistics of Weibo dataset. We hope that our model could be trained on dataset from several known domains, but can effectively predict fake news from other knowledge domains that the training set has not seen. To validate this cross-domain adaptability of the model, we design cross-domain experiments on the Weibo dataset and call this dataset as a cross-domain dataset. We can see that the amount of data in different fields varies greatly in Weibo dataset. In order to prove the cross-domain nature of our model and maintain a balanced amount of data, we utilize the data-intensive and easily accessible domain as training datasets (health, economic, technology, entertainment, society) and others as testing datasets (military, political and education).

5 The Weibo dataset could be obtained from https://www.datafountain.cn/competitions/422
6 The RCED dataset could be obtained from https://github.com/thunlp/Chinese_Rumor_Dataset and https://www.dropbox.com/s/7ewzdrbelpmrnxu/rumdetect2017.zip?dl=0

To verify the effectiveness of our model in both cross-domain and in-domain datasets, we employ an in-domain dataset named RCED 6. In order to obtain a larger dataset, we merged the two datasets compiled by (Song, Yang, Chen, Tu, Liu & Sun, 2019) and (Ma et al., 2016) as our RCED dataset. We filter out the news that contains few sentences and we only focus on documents with more than two sentences. Meanwhile, this dataset employs the fake and non-fake data from Sina Weibo and captures both original messages and all their reply messages as well. And we solely use the original news content of this dataset. The detailed statistics are shown in Tabel 3. Unlike the Weibo dataset, RCED is not a cross-domain dataset. We use it to validate the performance of models on in-domain short stories.
Table 3
Descriptive statistics of the RCED dataset

| Veracity | Train | Test |
|----------|-------|------|
| Real     | 1650  | 184  |
| Fake     | 1009  | 112  |
| Total    | 2659  | 296  |

Table 4
Descriptive statistics of document length on datasets

| Veracity | LUN-train | LUN-test | SLN   | Weibo | RCED |
|----------|-----------|----------|-------|-------|------|
| Avg. length | 18.81     | 23.02    | 30.25 | 5.1   | 4.12 |
| Max. length  | 333       | 370      | 200   | 95    | 107  |

4.1.3. Data Preprocessing

Same as the paper (Vaibhav et al., 2019), in all four datasets, we first hold out 10% of the whole dataset for test. Then, we randomly divide the rest of dataset into training and validation subsets with proportion of 80% and 20%.

For SLN and LUN-test datasets in English, the words are separated by white space. The sentences are separated by English full stop. And for Chinese datasets like Weibo and RCED, we cut the words by Jieba library. The sentences are segmented by Chinese full stop. We select the instances that have more than 2 sentences. Short news that does not meet the criteria is filtered out. We also count the average and maximum number of the sentences contained in per news document of each dataset as shown in Table 4. We will prove the effectiveness of the model regardless of the length of the news.

4.2. Comparison Methods

We compared our proposed model SemSeq4FD to 7 state-of-the-art baseline models which can be divided into three categories: machine learning models, non-graph deep learning network models and graph-based deep learning network models.

The baseline models based on machine learning include:

- **SVM** (Scholkopf & Smola, 2001): A Support Vector Machines (SVM) model with linear kernel are used to classify fake news. It detects fake news based on linguistic features extracted from news content.

  The detailed experimental process for SVM is as follows: We first collect the ngram vocabulary features of each document after a number of data pre-processing like removing stop words and tokenization. Then we employ the efficient text representation algorithm Term Frequency-Inverse Document Frequency (TF-IDF) to obtain the term frequency values on these ngrams features (Zhang, Zhao & LeCun, 2015). Finally, each document could be represented as a vector using the term frequency values and the vector would be fed into the SVM model for classification.

- **Logistic Regression** (Kleinbaum, Dietz, Gail, Klein & Klein, 2002): A Logistic Regression model are utilized to model article content and detect fake news. We vectorize the document using the same TF-IDF method as described above, then classify it using a stable and effective logistic regression method.

  **Hyperparameters**: For both SVM and Logistic Regression, the codes were written with scikit-lean and the ngram feature we extracted are unigram, bigram and trigram. The maximum number of features is 500. Other values were scikit-learn defaults.

The baseline models based on non-graph deep learning network are as follows:

- **CNN** (Kim, 2014): A Convolutional Neural Network are utilized to solve the fake news detection problem. We use a 1-d convolution layer with a filter of size 3 over the word embeddings of the whole documents, followed by a max-pooling layer and a fully connected layer.
• BERT (Devlin, Chang, Lee & Toutanova, 2019): We use the Google-BERT pre-trained model to vectorize each sentence in the document and classify the document with a LSTM network using these sentences representations. As each language has individual pre-trained BERT model, for English datasets, we utilize the bert-base-uncased pre-trained model to represent the sentence. And for Chinese datasets, we first use the Chinese word segment tool Jieba to cut the words and then vectorize sentences with the bert-base-chinese pre-trained model before they are fed into the LSTM network.

Hyperparameters: For CNN, we set the filter size to be 3, the dimension of word embeddings is 100 and the output channel is 100. For BERT, we implement the experiment with 32.0 GB memory. The maximum number of sentences in a document is 50. The dimension of sentence embeddings after pre-trained is 768. The batch size is 2.

The baseline models based on graph deep learning network include:

• GCN (Vaibhav et al., 2019): A graph convolutional network (GCN) (Kipf & Welling, 2017) has been applied to detect fake news. We first encode each sentence in document with a LSTM encoder. Then we apply the fully connected graph convolutional network proposed by (Vaibhav et al., 2019) to learn high-level sentence representation. These representations are converted to document representation through a max-pooling layer and the document representation is fed into the fully connected layer to get its prediction.

• GAT (Vaibhav et al., 2019): A Graph Attention Network (GAT) (Veličković, Cucurull, Casanova, Romero, Liò & Bengio, 2018) has been introduced to detect fake news. We apply a graph attention network to the fully connected graph before the max-pooling layer.

• GAT2H (Vaibhav et al., 2019): A Graph Attention Network with two attention heads (GAT2H) is a graph attention network extended by employing two-heads attention. We apply the GAT2H to the same fully connected graph and then concatenate each head’s output representations horizontally and fed it into the max-pooling layer.

Hyperparameters: For GAT and GAT2H, the slope of LeakyReLU is 0.2. By default, other parameters of GAT, GAT2H and GCN have the same values as our SemSeq4FD method. The details are provided in the Section 4.3.1.

4.3. Experimental Setup

4.3.1. Implementation Details and Hyperparameters

The experimental environment is as follows: Intel i7 2.20 GHz processor, 8.0 GB memory, GTX-1050 ti GPUs. And all codes are implemented in Python (3.6.4). We implement all machine learning based baseline models with scikit-learn libraries (0.22.1). And we implement all deep learning based baseline models, as well as our model, with Pytorch libraries (1.1.0).

To make a fair comparison, all results on all datasets have been averaged over several trials. We use the Adam optimizer as optimization. The learning rate starts at 0.001 and reduced by 2 times if the validation accuracy doesn’t increase for 3 epochs. The parameters are updated using stochastic gradient descent.

For hyperparameter settings, Table 5 presents a list of hyperparameters of our proposed model. The abstracts of these parameters are as follows:

• Max Sent Len: the threshold to control the maximum length of each sentence.

• Max Sents in a Doc: the threshold to control the maximum number of sentences in a document.

• Emb Dimension: the dimension of word embedding.

• Hidden Dimension: the dimension of primary sentence embedding.

• Node Emb Dimension: the dimension of enhanced sentence representation.

• Output Dimension: the dimension of document representation.

• K: the 1-D CNNs filter size in the SemSeq4FD model.
Table 5  
The details of the parameters

| Parameter          | Value |
|--------------------|-------|
| Max Sent Len       | 104   |
| Max Sents in a Doc | 50    |
| Emb Dimension      | 100   |
| Hidden Dimension   | 100   |
| Node Emb Dimension | 32    |
| Output Dimension   | 32    |
| K                  | 3     |
| Dropout            | 0.2   |
| Batch Size         | 32    |
| Max Epochs         | 10    |

- dropout: the dropout rate which we set to prevent neural networks from overfitting.
- Batch Size: the size of each batch.
- Max Epochs: the threshold to control the maximum epoch in training step.

4.3.2. Evaluation Metrics

This paper uses five different evaluation metrics to compare the performance of the proposed model, including four general metrics: Accuracy, Precision, Recall and F1 score and one special metric: NRI.

**General Metrics**: The ultimate goal of our model is to detect whether a news document is fake or real. For this binary classification problem, news documents in testing datasets can usually be divided into four groups: TP (True Positive), FP (False Positive), TN (True Negative) and FN (False Negative) according to their ground-truth label and label predicted by models.

- **TP**: The number of fake news documents correctly predicted as fake news.
- **FP**: The number of real news documents wrongly predicted as fake news.
- **TN**: The number of real news documents correctly predicted as real news.
- **FN**: The number of fake news documents wrongly predicted as real news.

With above concepts, the explanations for Accuracy, Precision, Recall and F1 score metrics are as follows:

The **Accuracy** metric describes the proportion of all news documents predicted to be correct.

\[
\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}
\]  
(14)

The **Precision** metric usually describes the proportion of fake news documents correctly predicted as fake news in all news documents predicted as fake news. However, such calculation method just focuses on fake news documents rather than the overall documents datasets. To make a fair evaluation, we followed the previous work (Vaibhav et al., 2019) and used the macro-average method. Specifically, we first calculated the precision for both fake news documents \((P_{fake})\) and real news documents \((P_{real})\) respectively and then averaged them as our Precision score (Precision) :

\[
\text{Precision} = \frac{1}{2} \left( P_{fake} + P_{real} \right) = \frac{1}{2} \left( \frac{TP}{TP + FP} + \frac{TN}{TN + FN} \right)
\]  
(15)

Similarity, the **Recall** and **F1 score** can be calculated as follows:

\[
\text{Recall} = \frac{1}{2} \left( R_{fake} + R_{real} \right) = \frac{1}{2} \left( \frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right)
\]  
(16)

Wang et al.: Preprint submitted to Elsevier  Page 12 of 21
Table 6
Main results of English datasets.
The best model performance values and the best baseline model values are emphasized by **bolding** and **underlining**, respectively.

| Method       | SLN (cross-domain) | LUN-test (cross-source) |
|--------------|---------------------|-------------------------|
|              | Accuracy | Precision | Recall | F1   | Accuracy | Precision | Recall | F1   |
| SVM          | 0.8333   | 0.8337    | 0.8333 | 0.8332| 0.7886   | 0.8105    | 0.7886 | 0.7848|
| Logistic     | 0.8388   | 0.8388    | 0.8388 | 0.8388| 0.7893   | 0.8083    | 0.7893 | 0.7860|
| CNN          | 0.6452   | 0.6466    | 0.6452 | 0.6440| 0.9094   | 0.9112    | 0.9088 | 0.9086|
| BERT         | 0.7583   | 0.7662    | 0.7583 | 0.7565| 0.8346   | 0.8356    | 0.8346 | 0.8345|
| GCN          | 0.8640   | 0.8670    | 0.8640 | 0.8638| 0.9224   | 0.9248    | 0.9222 | 0.9222|
| GAT          | 0.8538   | 0.8567    | 0.8538 | 0.8535| 0.9255   | 0.9261    | 0.9251 | 0.9251|
| GAT2H        | 0.8584   | 0.8600    | 0.8584 | 0.8580| 0.9178   | 0.9212    | 0.9178 | 0.9176|
| SemSeq4FD    | 0.8842   | 0.8904    | 0.8842 | 0.8838| 0.9378   | 0.9390    | 0.9378 | 0.9378|
| Improvement  | 2.338%   | 2.699%    | 2.338% | 2.315%| 1.329%   | 1.174%    | 1.373% | 1.373%|

Table 7
Main results of Chinese datasets.
The best model performance values and the best baseline model values are emphasized by **bolding** and **underlining**, respectively.

| Method       | Weibo (cross-domain) | RCED (in-domain) |
|--------------|-----------------------|------------------|
|              | Accuracy | Precision | Recall | F1 | Accuracy | Precision | Recall | F1 |
| SVM          | 0.7398   | 0.7905    | 0.7607 | 0.7363| 0.8479   | 0.8404    | 0.8340 | 0.8369|
| Logistic     | 0.7378   | 0.7939    | 0.7598 | 0.7337| 0.8378   | 0.8401    | 0.8101 | 0.8205|
| CNN          | 0.7262   | 0.7611    | 0.7438 | 0.7242| 0.7459   | 0.8298    | 0.6884 | 0.6766|
| BERT         | 0.8058   | **0.8387** | 0.8224 | 0.8049| 0.8885   | 0.8792    | **0.8876** | **0.8828** |
| GCN          | 0.7669   | 0.7899    | 0.7810 | 0.7664| **0.8911** | **0.8883** | 0.8793 | 0.8828|
| GAT          | 0.6718   | 0.7626    | 0.7005 | 0.6589| 0.8763   | 0.8758    | 0.8600 | 0.8661|
| GAT2H        | 0.7669   | 0.8063    | 0.7852 | 0.7651| 0.8803   | 0.8840    | 0.8611 | 0.8659|
| SemSeq4FD    | 0.8174   | 0.8273    | **0.8271** | **0.8174** | 0.9034   | 0.9069    | 0.8873 | 0.8950|
| Improvement  | 1.440%   | -         | 0.571% | 1.553% | 1.380% | 2.094% | - | 1.382%

\[ F_1 = \frac{1}{2} \left( F_1^{fake} + F_1^{real} \right) = \frac{1}{2} \left( \frac{2P_{fake}^R_{fake}}{P_{fake} + R_{fake}} + \frac{2P_{real}^R_{real}}{P_{real} + R_{real}} \right) \]  

(17)

Special Metric: We introduced a novel performance evaluation metric named NRI to the fake news detection task. "NRI" is the abbreviation of "Net Reclassification Improvement". It is commonly used in healthcare domain, to quantify how well a new model reclassifies subjects as compared to an old model. The detailed information about NRI metric and its experimental results are shown in Section 4.7.

4.4. Performance Comparison (RQ1)

In order to answer RQ1, we compare our model against several related models covering the machine learning methods as well as the deep learning methods on four datasets as exhibited in Table 6 and Table 7. From the experimental results, we have the following observation conclusions:
Experimental results on four fake news datasets demonstrate that our proposed method is significantly better than all the comparison models and can detect fake news effectively with high accuracy. We also underline the best result of each metrics among baseline models in both tables. For English datasets, SemSeq4FD boosts F1 score by around 2.315% and 1.373% on average in SLN and LUN-test datasets. And for Chinese datasets, our model obtains 1.553% and 1.382% improvements in Weibo and RCED datasets.

We can see that the models based on the graph neural networks with graph structure information affect the performance evidently, which illustrate the effectiveness of graph convolution operation. Hence, it is essential to consider the graph structure among article sentences in learning their enhanced representations. In addition, SVM and Logistic with TF-IDF method outperform CNN model in most of the datasets, which indicates that the word frequency feature could help detect fake news.

Our model has been proved to work well in both English and Chinese languages.

- For English datasets
  
  We use LUN-test as our cross-source dataset and SLN as our cross-domain dataset. Both of them are considered to be out of domain test sets in (Vaibhav et al., 2019). However, in our view, the LUN-test comes from different sources of publishers, which have different writing styles. We call it as a cross-source dataset. Experiments show that our model has good generalization ability on cross-source and cross-domain English datasets.

- For Chinese datasets
  
  To verify that our proposed model could detect fake news in cross-domain Chinese datasets, we use Weibo as a cross-domain dataset and RCED as a in-domain dataset. Our model outperforms all of the baseline models on both datasets. This shows that our model has cross-domain characteristics and can really predict cross-domain data accurately. This is very helpful for practical applications, because we can use news from social, entertainment and other fields with large amounts of data as training set to predict the news in multiple fields where people have difficulty getting large data. We also obtain the results of validation set that split from the Weibo training set, on which CNN, GCN, and our models can achieve accuracy of 0.9749, 0.9764, and 0.9829, respectively. This indicates that in the same domain, models will have a fake high accuracy on the validation set, and for test sets which have different domain, they will perform poorly.

(4) Table 4 compares document lengths by counting the number of sentences contained in each document. We can see that Table 6 and Table 7 demonstrate the performance of our model with different news length. It is obvious that news length largely affects classification results and it significantly limits the model’s ability to enhance structured representations (e.g. Compared with the English dataset with long documents, the results of Chinese dataset with short documents have a lower improvement). Besides, our model is slightly below the BERT model in just one metrics (precision) on each Chinese dataset. Although the BERT baseline model works well on several NLP tasks, it seems not good at handling fake news detection tasks with cross-domain and cross-source datasets in our experiments. Meanwhile, it runs for a long time and requires high computational power and memory of the device in the training step. We still believe that our model has advantages.

We also compare the F1 standard deviation of the baseline models based on the graph neural networks and our model in SLN and LUN-test. Table 8 shows that our model is more stable than these baselines.

4.5. Ablation Analysis (RQ2 & RQ3)

To further illustrate the effectiveness of employing local sequential order through Text-CNN and global sequential order through LSTM network in learning entire document-level representations, we design the ablation experiments and explore the two component of SemSeq4FD.

More concretely, we investigate the effects of these components by deriving two variants. We define the variants of SemSeq4FD as follows:

- w/o CNN: w/o CNN is a variant of SemSeq4FD, which removes the Text-CNN and does not consider the local contextual information between consecutive sentences. The enhanced sentence representations are learned by the SA-GCN module and then they are fed into the LSTM network without concatenate operation.
Table 8
Standard deviation of F1 scores

|          | SLN  | LUN-test |
|----------|------|----------|
|          | F1-SD| F1-SD    |
| SemSeq4FD| 0.0053| 0.0041   |
| GCN      | 0.0068| 0.0050   |
| GAT      | 0.0242| 0.0068   |
| GAT2H    | 0.0129| 0.0057   |

F1-SD denotes the standard deviation of F1 scores.

- **w/o LSTM**: w/o LSTM is a variant of SemSeq4FD, which excludes the LSTM network in the Document Representation module and does not specifically model the global sequential order. As an alternative, the document representation $\mathbf{z}$ is obtained by applying the max-pooling layer on the enhanced sentence representations learned by the Sentence Representation module.

The experimental results of F1 score on all datasets are shown in Figure 2. From these column charts, we could draw the corresponding conclusions:

![Figure 2: Ablation results on four datasets.](image)

SemSeq4FD is the proposed model. "w/o CNN" is a variant of SemSeq4FD, which removes the Text-CNN in the Sentence Representation module. "w/o LSTM" is a variant of SemSeq4FD, which excludes the LSTM network in the Document Representation module. Ablation experiments of SemSeq4FD demonstrated the effectiveness of modeling both local sequential order and global sequential order information.

1. Our proposed model achieves excellent performance on each dataset and outperforms its variants that without CNN and LSTM. This indicates that simultaneously consider both local sequential order and global sequential information is indispensable for model.

2. Compared with w/o CNN, w/o LSTM declined further in all datasets, which implies the LSTM network indeed plays a vital role in fake news detection. It seems that combining the enhanced sentence representations through LSTM sequentially to learn the global sequential order is crucial for our model.

4.6. Few-Sample Stability Evaluation (RQ4)

There are not always a large number of samples in actual work. To answer RQ4, we design experiments to explore the model’s adaptability to small samples in SLN and LUN-test. We divide the training data into 20% to 80% and
compare the experimental results of the baseline models based on the graph neural networks and SemSeq4FD. Figure 3 clearly illustrate the variety of the F1 metrics from 20% to 80% fraction of training data in two datasets.

![Figure 3: Few-Sample stability evaluation results on two datasets.](image)

By comparing SemSeq4FD, GCN, GAT and GAT2H which are all common graph-based algorithms, it is a fact that as the fraction of training data increases, our model SemSeq4FD shows good stability in predictive ability and outperforms the baseline models. By contrast, other methods fluctuate and they go up and down. And even the amount of data is small our model still maintains a high F1 value. This obviously shows that the SemSeq4FD model can capture vital information and fully utilize the limited data in the early detection of the spreading fake news at the real situation.

4.7. Net Reclassification Improvement Analysis (RQ5)

Previous experiments directly compare misclassification rates, such as accuracy. However, some instances are misclassified in the baseline model, but are corrected in our model. The general metrics we mentioned above (accuracy, precision, etc.) can only compare differences between the ground-truth label and the predicted label of the models on the overall test set. And there are limitations in explaining how well a model correctly reclassifies instances as compared to another model. Thus, in order to give an intuitive illustration of our model, we further analyzed the results of all four datasets by using the NRI metric (Leening, Vedder, Witteman, Pencina & Steyerberg, 2014).

In the testing datasets, all instances with positive label would be classified as positive group and all instances with negative label could be classified as negative group. For a better explanation of NRI, we introduce the confusion matrix as shown in Table 9, which involves the following concepts:
Table 9
The confusion matrix of NRI index.

| Pos-group (\(N_{pos}\)) | New Model | Neg | Pos |
|--------------------------|-----------|-----|-----|
| Old Model                |           |     |     |
| Neg                      | --        | \(\text{New}^+_{pos}\) | \(\text{New}^-_{neg}\) |
| Pos                      | \(\text{New}^-_{pos}\) | --   |     |

- \(N_{pos}\): The number of instances in positive group.
- \(N_{neg}\): The number of instances in negative group.
- \(\text{New}^+_{pos}\): The number of positive instances that have been wrongly classified by the Old Model, but correctly reclassified by the New Model.
- \(\text{New}^-_{neg}\): The number of negative instances that have been wrongly classified by the Old Model, but correctly reclassified by the New Model.
- \(\text{New}^-_{pos}\): The number of positive instances that have been correctly classified by the Old Model, but wrongly reclassified to the negative label by the New Model.
- \(\text{New}^+_{neg}\): The number of negative instances that have been correctly classified by the Old model, but wrongly reclassified to the positive label by the New Model.

With the confusion matrix of NRI index, the NRI metric between Old Model and New Model would be explained. Formally, for the positive group which have \(N_{pos}\) instances, the NRI \(\text{pos}\) is given by:

\[
NRI_{\text{pos}} = \frac{\text{New}^+_{pos} - \text{New}^-_{pos}}{N_{pos}} \tag{18}
\]

Similarly, for the negative group which have \(N_{neg}\) instances, we could represent the NRI \(\text{neg}\) as:

\[
NRI_{\text{neg}} = \frac{\text{New}^+_{neg} - \text{New}^-_{neg}}{N_{neg}} \tag{19}
\]

And the Additive NRI can be regarded as the sum of them, which can be expressed as follows:

\[
NRI = NRI_{\text{pos}} + NRI_{\text{neg}} = \frac{\text{New}^+_{pos} - \text{New}^-_{pos}}{N_{pos}} + \frac{\text{New}^+_{neg} - \text{New}^-_{neg}}{N_{neg}} \tag{20}
\]

The NRI metric allows a more comprehensive comparison of the predictive ability of the two models than the general metrics. It ranged between -2 and 2. If NRI is greater than 0, it means that the new model improves the predictive ability, otherwise it indicates that the predictive ability has declined.

Take the SemSeq4FD model and the GAT model in the Weibo dataset as an example. We divide the news documents into fake news and real news. The real news group contains 230 real news and the fake news group contains 285 fake news. As shown in Table 10, we could obtain confusion matrix of Weibo dataset similar to we mentioned above. We can see that in the fake news group, our model SemSeq4FD correctly reclassifies 90 fake news documents that have been wrongly classified as real news by the GAT model. However, our model only wrongly reclassifies 2 fake news documents that were correctly classified by the GAT model. In the real news group, our model correctly reclassifies 1 real news document that was misclassified by the GAT model. And our model misclassifies 14 real news documents as fake news, while the GAT model have been correctly classified these real news documents. Thus the \(\text{New}^+_{pos}\), \(\text{New}^-_{pos}\)
Table 10
The confusion matrix of NRI index in Weibo dataset.

|                | Real-group(230) | SemSeq4FD |
|----------------|-----------------|-----------|
| Real           | 2               | 14        |
| Fake           |                 |           |

New\textsubscript{\text{neg}} and New\textsubscript{\text{-neg}} values of SemSeq4FD (New Model) relative to GAT (Old Model) are 90, 2, 1, 14, respectively. The NRI value of SemSeq4FD relative to GAT should be \((90 - 2) / (285) + (1 - 14) / (230) \approx 0.25\). And for GAT, its NRI value relative to SemSeq4FD is -0.25. It is generally accepted that the costs of classification errors in different categories are not equal. In the task of classifying fake news, we believe that wrongly classifying fake case as real one is obviously more dangerous. Although our SemSeq4FD model is slightly weaker than GAT model in detecting real news, it works well in detecting fake news and is more useful in practical work than GAT model.

We calculate the NRI values between each pairs of models in the graph-based baselines as exhibited from Figure 4 to Figure 7. The warm color indicates that the models on the horizontal axis (y axis) are better than models on the vertical axis (x axis), and the cold color indicates that models on the y axis have a worse ability to classify fake news than those on the x axis. We annotate each cell with the NRI value using float formatting. Compared with the baseline models, we can see that our model is better at predicting fake news, and can detect fake news more easily and accurately from fake and real mixed news than others.

![Figure 4: NRI values between each pair of graph-based models in SLN dataset.](image)

5. Discussion and Conclusion

In this paper, we present an end-to-end model for early fake news detection based solely on its content, which learns enhanced text representations from the word-level to the sentence-level, and then to the document-level. In particular, the model considers global semantic relations feature, local sequential order feature and the global sequential order feature among sentences in a news document. Given a news article, we first construct a complete graph structure to learn the global semantic relations feature via a graph convolutional network with self-attention mechanism. We also...
employ a 1D convolutional network to capture the local sequential order feature. Then a LSTM network is utilized to express global sequential order based on the enhanced representations. This is followed by a classifier to distinguish fake or real news. The experimental results on four datasets in English and Chinese, including cross-source and cross-domain datasets, show that the proposed model outperforms other state-of-the-art methods.

The experimental results demonstrate the stability and superiority of the proposed model. We found that text structure is significant to predicting the fake news. In the future, we will develop our research on the following three points: (1) The majority of text structure information can be mined and utilized. We can develop our model to discover and make full use of these complex text structures. (2) For text modeling methods, we still need more detailed and in-depth research to improve its representation effect. (3) Generally, complete news always has its text content and

Figure 5: NRI values between each pair of graph-based models in LUN-test dataset.

Figure 6: NRI values between each pair of graph-based models in Weibo dataset.
none-textual features like images or videos. The multi-modal information also plays a vital role in fake news detection task. We will utilize this type information to solve the problem of multi-views fake news recognition.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No: 61872260) and GF Innovative Research Program.

References

Ahn, Y., & Jeong, C. (2019). Natural language contents evaluation system for detecting fake news using deep learning. In 2019 16th International Joint Conference on Computer Science and Software Engineering (IJCSSE) (pp. 289–292).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In NAACL '19 (pp. 4171–4186).

Horne, B. D., & Adali, S. (2017). This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In ICWSM ‘17.

Hu, G., Ding, Y., Qi, S., Wang, X., & Liao, Q. (2019). Multi-depth graph convolutional networks for fake news detection. In CCF International Conference on Natural Language Processing and Chinese Computing (pp. 698–710). Springer.

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In ICLR 2017.

Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., & Klein, M. (2002). Logistic Regression. Springer.

Kouzy, R., Abi Jaoude, J., Kraitem, A., El Alam, M. B., Karam, B., Adib, E., Zarka, J., Traboulsi, C., Akl, E. W., & Baddour, K. (2020). Coronavirus goes viral: Quantifying the covid-19 misinformation epidemic on twitter. Cureus, 12.

Ko, J., & Lee, J. (2018). Mining significant microblogs for misinformation identification: An attention-based approach. ACM TIST '18, 9, 1–20.

Liu, Q., Yu, F., Wu, S., & Wang, L. (2016). Mining significant microblogs for misinformation identification: An attention-based approach. ACM TIST '18, 9, 1–20.

Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B. J., Wong, K.-F., & Cha, M. (2016). Detecting rumors from microblogs with recurrent neural networks. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence IJCAI16 (p. 38183824). AAAI Press.

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
Wang et al.: Preprint submitted to Elsevier
Credit Author Statement

Yuhang Wang: Methodology, Data curation, Software, Writing-Original draft, Visualization.
Li Wang: Conceptualization, Writing - Review & Editing, Funding acquisition, Supervision.
YanJie Yang: Investigation, Data curation, Software.
Tao Lian: Methodology, Writing - Review & Editing.
Declaration of interests

☒ 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.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: