



Multi-level word features based on CNN for fake news detection in cultural communication

Qingyuan Hu¹ · Qian Li¹ · Youshui Lu¹ · Yue Yang¹ · Jingxian Cheng¹

Received: 28 May 2019 / Accepted: 29 July 2019

© Springer-Verlag London Ltd., part of Springer Nature 2019, corrected publication 2022

Abstract

In recent years, due to the booming development of online social networks, fake news has been appearing in large numbers and widespread in the online world. With deceptive words, online social network users can get infected by these online fake news easily, which has brought about tremendous effects on the offline society already. An important goal in improving the trustworthiness of information in online social networks is to identify the fake news timely. However, fake news detection remains to be a challenge, primarily because the content is crafted to resemble the truth in order to deceive readers, and without fact-checking or additional information, it is often hard to determine veracity by text analysis alone. In this paper, we first proposed multi-level convolutional neural network (MCNN), which introduced the local convolutional features as well as the global semantics features, to effectively capture semantic information from article texts which can be used to classify the news as fake or not. We then employed a method of calculating the weight of sensitive words (TFW), which has shown their stronger importance with their fake or true labels. Finally, we develop MCNN-TFW, a multiple-level convolutional neural network-based fake news detection system, which is combined to perform fake news detection in that MCNN extracts article representation and WS calculates the weight of sensitive words for each news. Extensive experiments have been done on fake news detection in cultural communication to compare MCNN-TFW with several state-of-the-art models, and the experimental results have demonstrated the effectiveness of the proposed model.

Keywords Multi-level CNN · Local convolutional · Global semantics · Cultural communication

1 Introduction

In recent years, the topic of fake news has experienced a resurgence of interest in society. The increased attention stems largely from growing concerns around the widespread impact of fake news on public opinion and events. In January 2017, a

spokesman for the German government stated that they “are dealing with a phenomenon of a dimension that they have not seen before,” referring to the proliferation of fake news on social media. Although social media has increased the ease with which real-time information disseminates, its popularity has exacerbated the problem of fake news by expediting the speed and scope at which false information can be spread. Fuller et al. [1] noted that with the massive growth of online communication, the potential for people to deceive through computer-mediated communication has also grown and such deception can have disastrous and far-reaching results on many areas of our lives. Figure 1 shows the social network of the users who share fake news and true news [2–4].

Over the past decade, we have witnessed the development of fake news detection technologies, mainly grouping into cue and feature-based methods [5–7], which can be employed to distinguish fake news contents from true news contents by designing a set of linguistic cues that are informative of the content veracity, and linguistic analysis based methods [8, 9], which can be applied to distinguish fake from true news by

✉ Qingyuan Hu
qingyuanhu@foxmail.com

Qian Li
qianlixjtu@163.com

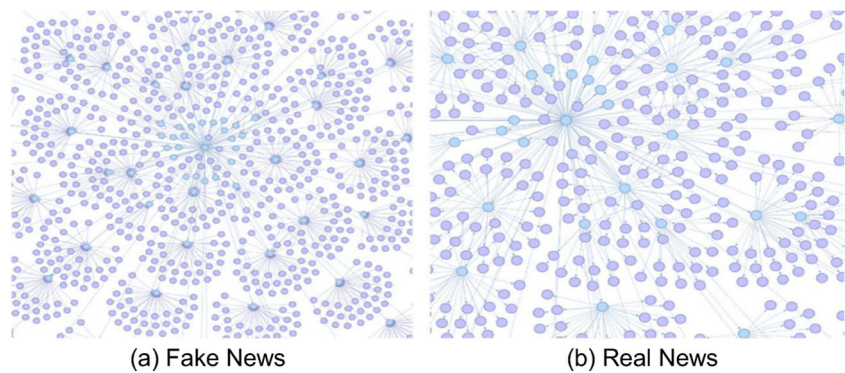
Youshui Lu
lucienlu@stu.xjtu.edu.cn

Yue Yang
yang_yue2015@stu.xjtu.edu.cn

Jingxian Cheng
chengjx3@126.com

¹ Xi'an Jiaotong University, 28 Xianning West Road, Xi'an, China

Fig. 1 a, b The social network of the users who share fake news and true news



exploiting differences in writing style, language, and sentiment. Such methods do not require task-specific, hand-engineered cue sets and rely on automatically extracting linguistic features from the text. Unfortunately, variations in linguistic cues implies that a new cue set must be designed for a new situation, making it hard to generalize cue and feature engineering methods across topics and domains. Linguistic analysis methods, although better than cue-based methods, still do not fully extract and exploit the rich semantic and syntactic information in the content.

Neural network is an attracted machine learning model that can learn the nonlinear mapping from data, especially for deep model. They have been employed for automatic detection of fake news [10–12] and show an impressed practical performance. However, even with sophisticated feature extraction of deep learning methods, fake news detection remains to be a challenge, primarily because the content is crafted to resemble the truth in order to deceive readers, and without fact-checking or additional information, it is often hard to determine veracity by text analysis alone. To tackle these challenges, we proposed MCNN-TFW which can provide deeper semantic analysis and understanding of the news article text and its veracity through the relationship between the article text content and the corresponding weight of sensitive words it invokes.

The multi-level convolutional neural network (MCNN) is designed to condense word-level information into sentences and the process the sentence-level representations with a convolutional neural network, to effectively capture semantic information from new texts which can be used to classify the article as fake or not. The method of calculating the weight of sensitive words (TFW) is denoted their importance to true and false news. The MCNN-TFW where MCNN captures semantic information from article text by representing it at the sentence and word levels, and WS calculate the weight of sensitive words in order to assist fake news detection.

We designed and implemented MCNN-TFW with 1200 lines of Python code. After applying MCNN-TFW to datasets with culture as subject, we found that it exhibits impressive performance of fake news detection. In summary, our main contributions include the following:

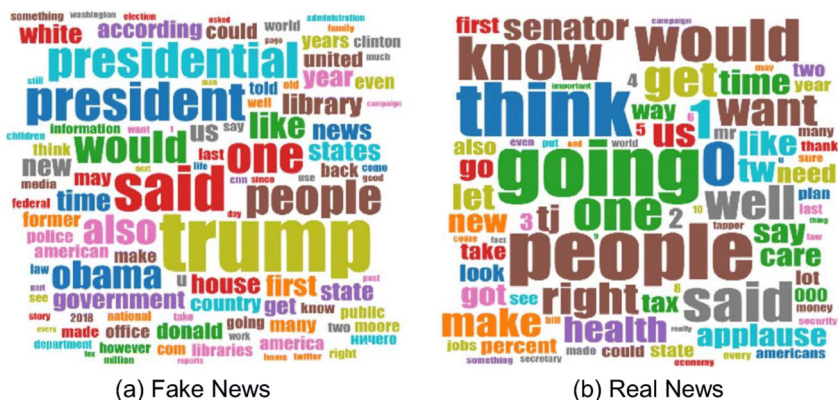
1. The MCNN employed the local convolutional features as well as the global semantics features for context feature learning.
2. We proposed a method of calculating the TFW, which has shown their stronger importance with their fake or true labels.
3. We designed and implemented a novel type of MCNN-TFW, a multiple-level convolutional neural network-based fake news detection system, which can be trained end to end.
4. We conducted extensive experiments to evaluate the performance of the MCNN-TFW system. The results show that it exhibits impressive performance of fake news detection.

The rest of this paper is organized as follows. Section 2 discusses related work on previous fake news detection. The framework of MCNN-TFW is described in detail in Section 3. The experimental results which show the performance of our framework are in Section 4. We conclude this paper in Section 5.

2 Related work

In order to detect fake news from article text, earlier fake news detection works were mainly based on manually designed features extracted from news articles or information generated during the news propagation process. Figure 2 shows the word cloud of fake news content and true news content [2, 4]. Though intuitive, manual feature engineering is labor intensive, not comprehensive, and hard to generalize. Recent research has focused on deep learning content-based methods. Deep learning methods alleviate the shortcomings of linguistic analysis based methods by automatic feature extraction, being able to extract both simple features and more complex features that are difficult to specify. Deep learning-based methods have demonstrated significant advances in text classification and analysis [10, 12, 13] and are powerful methods for feature extraction and classification with their ability to

Fig. 2 a, b Word cloud of fake news content and true news content



capture complex patterns relevant to the task. In this section, we will summarize these techniques.

2.1 Linguistic analysis

The most effective linguistic analysis method applied to fake news detection is the n-gram approach [8, 9, 14]. n-grams are sequences of n contiguous words in a text, constituting words (unigrams) and phrases (bigrams, trigrams) and are widely used in language modeling and text analysis. Apart from word-based features such as n-grams, syntactic features such as part-of-speech (POS) tags are also exploited to capture linguistic characteristics of texts. Ott et al. [9] examined whether this variation in POS tag distribution also exists with respect to text veracity. They trained a SVM classifier using relative POS tag frequencies of texts as features on a dataset containing fake reviews. Ott et al. obtained better classification performance with the n-grams approach, but nevertheless found that the POS tag approach is a strong baseline outperforming the best human judge. Later work has considered deeper syntactic features derived from probabilistic context-free grammars (PCFG) trees [15, 16]. Feng et al. [17] examined the use of PCFG to encode deeper syntactic features for deception detection. In particular, they proposed four variants when encoding production rules as features.

Even with word based n-gram features combined with deeper syntactic features from PCFG trees, linguistic analysis methods, although better than cue-based methods [5–7], still do not fully extract and exploit the rich semantic and syntactic information in the content. The n-gram approach is simple and cannot model more complex contextual dependencies in the text. Syntactic features used alone are less powerful than word based n-grams, and a naive combination of the two cannot capture their complex interdependence.

2.2 Convolutional neural networks

Convolutional neural networks (CNN) [18] are generally used in natural language processing tasks such as

semantic parsing [12, 19–22] and text classification [13]. Wang [23] proposed a convolutional neural network to classify short political statements as fake or not using the text features of the statements and available metadata. Qian et al. [24] also similarly demonstrated improved CNN performance over linguistic analysis-based methods such as LIWC, POS, and n-gram approach when classifying a collection of news articles as fake or true. In addition, to handle longer article texts, Qian et al. suggested a variant of the CNN architecture called two-level convolutional neural network (TCNN) which first takes an average of word embedding vectors for words in a sentence to generate sentence representations and then represents articles as a sequence of sentence representations provided as input to the convolutional and pooling layers. Qian et al. found the TCNN variant to be more effective than CNN in classifying the articles.

2.3 Other variants

Recurrent neural network (RNN) [25, 26] based architectures are also proposed for fake news detection. RNNs process the word embeddings in the text sequentially, one word/token at a time, utilizing at each step the information from the current word to update its hidden state which has aggregated information about the previous words. The final hidden state is generally taken as the feature representation extracted by the RNN for the given input sequence. A specific variant called long short-term memory (LSTM) [27–30], which alleviates some of the training difficulties in RNN, is often used due to its ability to effectively capture long range dependencies in the text and has been applied to fake news detection, similar to the use of convolutional neural networks, in several works [31–34], whereas in another variant, LSTM has been applied to both, article headline and article text (body), in an attempt to classify the level of disagreement between the two for deception detection [35].

3 Methodology of MCNN-TFW

In this section, we introduce the architecture of the proposed MCNN-TFW fake news detection model. MCNN-TFW is composed of two parts: (1) MCNN represents each article in the local convolutional features as well as the global semantics features and is able to apply detection on fake news articles. (2) TFW is employed to construct sensitive vocabulary set and calculate the weight of sensitive words. The architecture of the proposed model is shown in Fig. 3.

3.1 Notations

We consider the setting where we have a set of news articles D , and each article is denoted as d_i . Each article d_i is composed of a sequence of sentences $x_1, x_2 \dots x_{n_i}$, where n_i is the number of sentences in the article d_i . In the proposed model, the final feature vector extracted for each article d_i for classification is marked as y' . For each article, there will be several related weight of sensitive words. A given weight to article d_i is marked as $WS(d_i)$. For each article d_i , the target is marked as y_i . $y_i = 1$ means this article is real news, and $y_i = 0$ means this article is fake news.

3.2 Multi-level convolutional neural network

Due to the sequential nature of sentences, recurrent neural networks are widely employed to produce textual features. In our proposed framework, as shown in Fig. 4, we employ CNN for sentence encoding instead of RNN. The hierarchical representations of CNNs make local semantic learning in convolutional layers possible; it may better reflect the characteristics information such as sensitive words. Specifically, instead of only forcing a consistency in the semantic space of global features, we can also add the consistency constraints on

the local convolutional features. This additional constraint encourages the model to consider the regional semantics into consideration to focus more on sensitive words. Eventually, this design is expected to produce more robust and better global semantics for fake news detection.

Therefore, our model consists of two learning objectives. The first learning objective, also defined as global objective, is to learn the semantic embeddings using the feature representations of the whole sentences. We introduce the second objective function, local objective, which is to flatten the convolutional features into a vector.

Firstly, following the design in [36], the word embeddings are initialized using pre-trained word2vec model on Google News corpus [37], where each word is embedded into a feature space with dimension k . The article representation is derived from the sentence representations by concatenation of each sentence representation. Let $x_i \in \mathbb{R}^k$ be the k -dimensional word vector corresponding to the i th word in the sentence. The article d_i , containing n_i sentences, is represented as:

$$x_{1:n_i} = x_1 \oplus x_2 \oplus \dots \oplus x_{n_i} \tag{1}$$

where \oplus is a concatenation operator. In general, let $x_{i:i+j}$ refers to the concatenation of words $x_i, x_{i+1}, \dots, x_{i+j}$. Note that each sentence is represented on a word level; the news article now is represented on a sentence level as shown in Eq. (1).

A convolution operation then applies a filter $w \in \mathbb{R}^{hk}$ to a window of h sentences moving through the article to extract semantic information features from the article. A feature c_i is generated from a window of words $x_{i:i+h-1}$ by

$$r_i = f(w \cdot x_{i:i+h-1} + b) \tag{2}$$

Here $b \in \mathbb{R}$ is a bias term and f is an activation function such as ReLU. This filter is applied to each possible window of

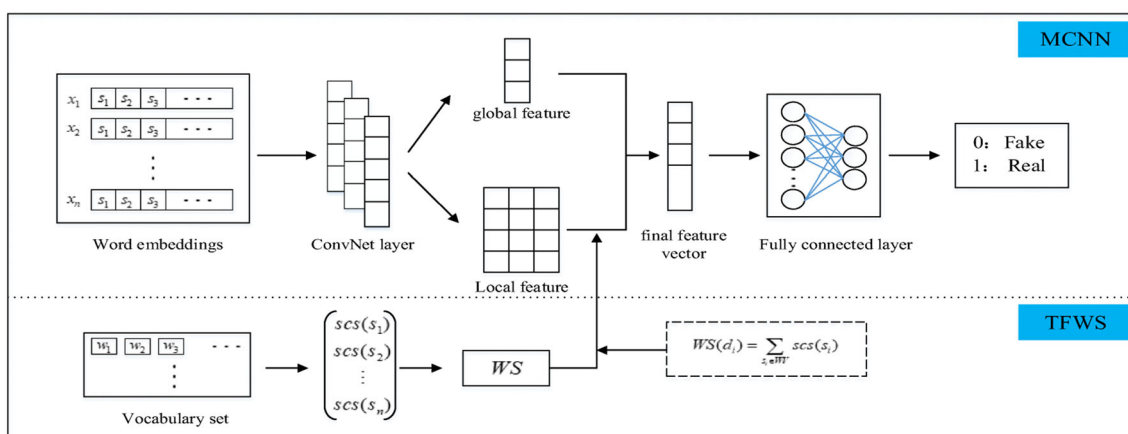


Fig. 3 The overview architecture of MCNN-TFW: (1) multi-level convolutional neural Network (Section 3.2) and (2) method of calculating the weight of sensitive words (Section 3.3)

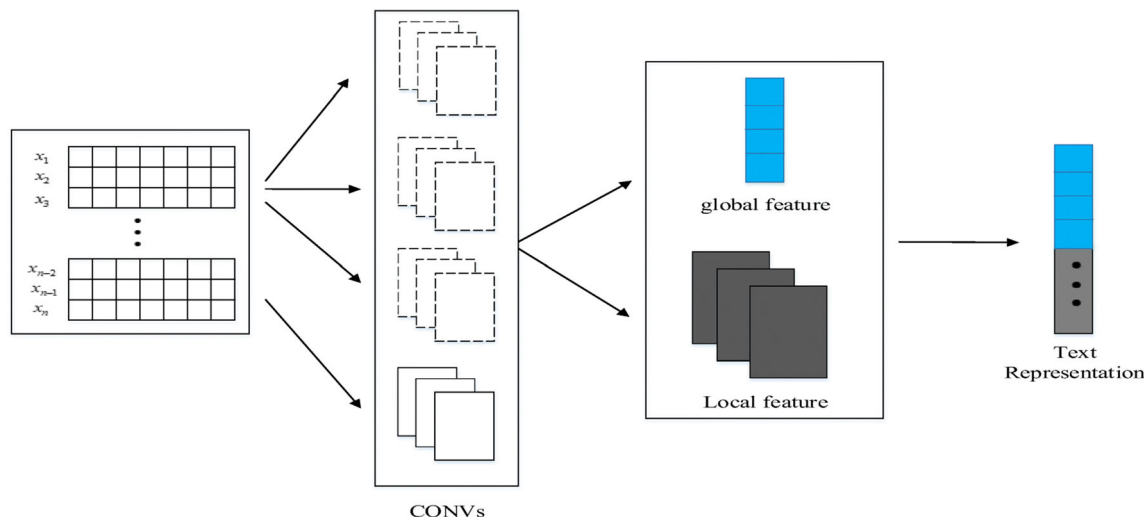


Fig. 4 The proposed convolutional neural network architecture for text representation learning. It consists of several convolutional layers with different kernel sizes and hierarchical text representation learning

words in the sentence $\{x_{1:h}, x_{2:h+1}, \dots, x_{n-h+1:n}\}$ to produce a feature map for each article

$$\text{Loc} = [r_1, r_2, \dots, r_{n-h+1}] \tag{3}$$

where $\text{Loc} \in \mathbb{R}^{n-h+1}$ is the local convolutional features. After that, we apply a max-over-time pooling operation over the feature map and take the maximum value $\text{Glo} = \max \{\text{oc}\}$ as the feature corresponding to this particular filter. Here Glo is the global semantics feature. Filters which have different lengths or have a same length but with different parameters are applied in order to capture features of different lengths and meaning.

Finally, we will combine the local convolutional features and the global semantics features in a certain way to form a final feature vector (Section 2.2) which are used as the input to a fully connected layer and a softmax output.

3.3 TFW: method of calculating the weight of sensitive words

Textual information of fake news can reveal important signals for their credibility detection; a set of frequently used words can also be extracted from the fake news. These extracted words have shown their stronger correlations with their fake or true labels.

Therefore, we first crawled frequent words (named sensitive words) from the dataset we constructed (Section 4.1); let WV denotes the complete vocabulary set. Then we calculated the weight of sensitive words which is denoted their importance to real and false news. Given that several sensitive words are used in fake and real news, the measurement would be biased if only the coefficient of a sensitive word is calculated with its frequency of occurrence in a fake news. We propose a TFW

measure of the sensitivity coefficient of sensitive words that exploits the idea of TF-IDF [38–40]. To achieve this, 5618 fake news and 5290 real news are downloaded from above dataset, real news in eight categories, such as art, artifacts, and historical sites. We use six terms of sensitive words s_i to understand its distribution in our fake and real news.

1. $nc(s_i)$: fake news count of s_i . It denotes the number of fake news using s_i in the fake news dataset.
2. $rc(s_i, c)$: real news count of s_i . It denotes the number of real news using s_i in category c .
3. $nrt(s_i)$: ratio of $nc(s_i)$ to the total number of fake news in the fake news dataset which is represented as p . $nrt(s_i)$ can be obtained with $nrt(s_i) = \frac{nc(s_i)}{p}$, where $p = 5618$ in our work.
4. $rct(s_i, c)$: ratio of $rc(s_i, c)$ to the total number of real news in category c which is represented as $q(c)$. $rct(s_i, c)$ can be obtained with $rct(s_i, c) = \frac{rc(s_i, c)}{q(c)}$.
5. $nrk(s_i)$: rank number of $nrk(s_i)$ among all the sensitive words.
6. $rrk(s_i, c)$: rank number of $rrk(s_i, c)$ among all the sensitive words in category c .

Through analysis of the above dataset by using these six terms, we can draw the following three conclusions:

1. Several sensitive words are used frequently in the fake and real news.
2. Several sensitive words are used more frequently in the fake news than in the real news.
3. The $rct(s_i, c)$ and $rrk(s_i, c)$ differ in the different categories.

In text mining literature, TF-IDF is a numerical statistic intended to reflect how discriminating a term is to a

document in a corpus. By utilizing the idea of TF-IDF for reference, we make the scs of a sensitive words be in positive correlation with its nrt and in negative correlation with its rrt . For sensitive words s_i of a news that belongs to a specific category c , its sensitivity coefficient $scs(s_i)$ is calculated with Eq. (4).

$$scs(s_i) = nrt(s_i) \times \log \frac{1}{rrt(s_i, c)} \quad (4)$$

The formula shows that the sensitivity coefficients calculated by the TFW measure can reflect the importance of sensitive words in different categories. However, there are some datasets that have no category information. Therefore, we calculate the sensitivity coefficients of sensitive words for such news as:

$$scs(s_i) = nrt(s_i) \times \log \frac{1}{rrt(s_i)} \quad (5)$$

$rrt(s_i)$ denotes the percent of news in all real news using the sensitive words s_i and it is obtained using Eq. (6), in which C denotes the set of all the real categories.

$$rrt(s_i) = \frac{1 + \sum_{c \in C} rc(s_i, c)}{\sum_{c \in C} q(c)} \quad (6)$$

3.4 Fake news detection user MCNN-TFW

MCNN is able to extract features including local convolutional features and the global semantics features from the article text and use that for predicting whether the article is fake or not, whereas TFW is able to calculate the weight of sensitive words that exist in each news.

In this section, we first combine the local convolutional features and the global semantics features based on weight of sensitive words for each news to form a final feature vector. Based on the pre-extracted vocabulary sets WV , given a news $d_i \in D$, we calculate the weight of sensitive words for each news as:

$$WS(d_i) = \sum_{s_i \in WV} scs(s_i) \quad (7)$$

The final feature vector is denoted as:

$$VLG = Glo.WS(d_i) + Loc \quad (8)$$

Then, the final feature vector is fed into a feedforward softmax classifier for classification as shown in Fig. 1 to predict whether the news is fake or real. During the learning process, the loss for each n sized batch sample is evaluated as the sum of the cross-entropy between the neural net's prediction and the true label, and the loss function is described as follows: where Y' denotes the label of input vector, and Y denotes the predicted label result [41–48].

4 Evaluation

In this section, to empirically validate our developed system MCNN-TFW, we first introduce the study setup of our experiments and then address the following four research questions.

- RQ 1: does MCNN-TFW have a higher performance?
- RQ 2: does MCNN-TFW outperform the baseline approaches in term of accuracy?
- RQ 3: what is the role of MCNN in MCNN-TFW?
- RQ 4: can MCNN-TFW work efficiently and be scalable for a large number of article?
- RQ 5: whether the method we proposed has general applicability?

4.1 Study setup

4.1.1 Datasets

We utilize five available fake news datasets in this study, which are from the benchmark dataset commonly used in today's methods. The first dataset is collected by Wang et al. [23] from the LIAR. For the next two datasets, we utilize two available online fake news datasets provided by Weibo [49] and Twitter15 [50]. The last two datasets are provided by NewsFN (https://github.com/GeorgeMcIntire/fake_real_news_dataset) and KaggleFN (<https://www.kaggle.com/mrisdal/fake-news>). In this paper, our focus is the detection

Table 1 The dataset information used in all the experiments

Dataset	No. of samples	Fake	Real	Train	Test	Dataset source
Dataset I	4180	2100	2080	60%	40%	Weibo/Twitter15/NewsFN
Dataset II	6728	3518	3210	60%	40%	LIAR/KaggleFN

Table 2 Descriptions of the used metrics

Term	Abbr	Description
True positive	TP	Number of apps correctly classified as malicious
True negative	TN	Number of apps correctly classified as benign
False positive	FP	Number of apps mistakenly classified as malicious
False negative	FN	Number of apps mistakenly classified as benign
True positive rate	TPR	$TP/(TP + FN)$
False positive rate	FPR	$FP/(FP + TN)$
Accuracy	ACC	$(TP + TN)/(TP + TN + FP + FN)$
Recall	r	$TP/(TP + FN)$
F-measure	F1	$2rp/(r + p)$
ROC area	AUC	Area under ROC curve

of fake news in cultural communication. Therefore, we combine all datasets and from which crawling culture as subject of news to from two new datasets. Dataset I uses 4180 news from the Weibo, Twitter15, and NewsFN. Dataset II uses 6728 news from LIAR and KaggleFN, including 3518 fake news and 3210 real news. Table 1 lists the information about these two datasets.

4.1.2 Experimental setting

In the experiments, we set the word embedding dimension to be 128 and filter size to 3, 4, 5. For each filter size, 64 filters are initialized randomly and trained. The whole network is trained using the Adam optimization algorithm with a learning rate of 0.001 and dropout rate of 0.5. The mini-batch size is 64 and both local and global loss are computed within each mini-batch.

We build and train the model using TensorFlow and use tenfold cross-validation for evaluation of the model. We used an Ubuntu 14.04 machine with Intel Core i7-5820 k CPU, GeForce GTX TITAN X GPU and 104 GB RAM to deploy our proposed framework. In the experiments, the GPU was utilized to accelerate machine learning algorithms. Table 2 lists the metrics used to evaluate MCNN-TFW.

Table 3 Performance of MCNN-TFW fake news detection on dataset I and dataset II

Dataset	TPR (%)	ACC(%)	F1 (%)
Dataset I	90.24	91.67	91.02
Dataset II	91.44	92.08	91.68

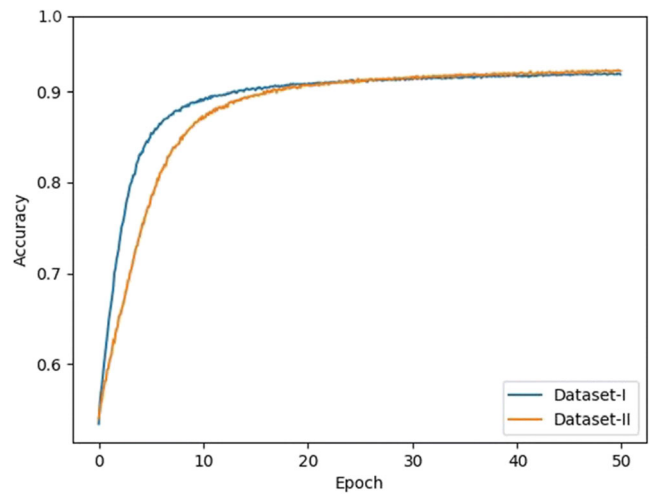


Fig. 5 Classification accuracy of MCNN-TFW on two datasets

4.2 Accuracy of MCNN-TFW

4.2.1 RQ 1: does MCNN-TFW have a higher performance?

We show the performance of MCNN-TFW fake news detection in cultural communication on dataset I and dataset II in Table 3. The accuracy of the classifier is 91.67% on dataset I, and the accuracy of the classifier is 92.08% on dataset II. This shows that the MCNN-TFW can classify fake news with high precision. Figure 5 shows the classification accuracy of MCNN-TFW on dataset I and dataset II.

As is shown in Fig. 6, We present the results in terms of the performance of detection, on varying percentage (20–80%) of data samples used as training data to evaluate the variation and stability in performance for the evaluated methods. Overall MCNN-TFW can classify fake news with high performance, even when the training data is limited.

We can infer that the main reason is that MCNN first introduced the local convolutional features as well as the global semantics features, to effectively capture semantic information from article texts. Then, we employed a method of calculating the weight of sensitive words (TFW), which has shown their stronger importance with their fake or true labels. Finally, the MCNN-TFW where MCNN captures semantic information from article text by representing it at the sentence and word levels and WS calculate the weight of sensitive words in order to assist fake news detection.

Answer to RQ 1 MCNN-TFW can classify fake news with high precision. The accuracy of the classifier is 91.67% on dataset I, and the accuracy of the classifier is 92.08% on dataset II.

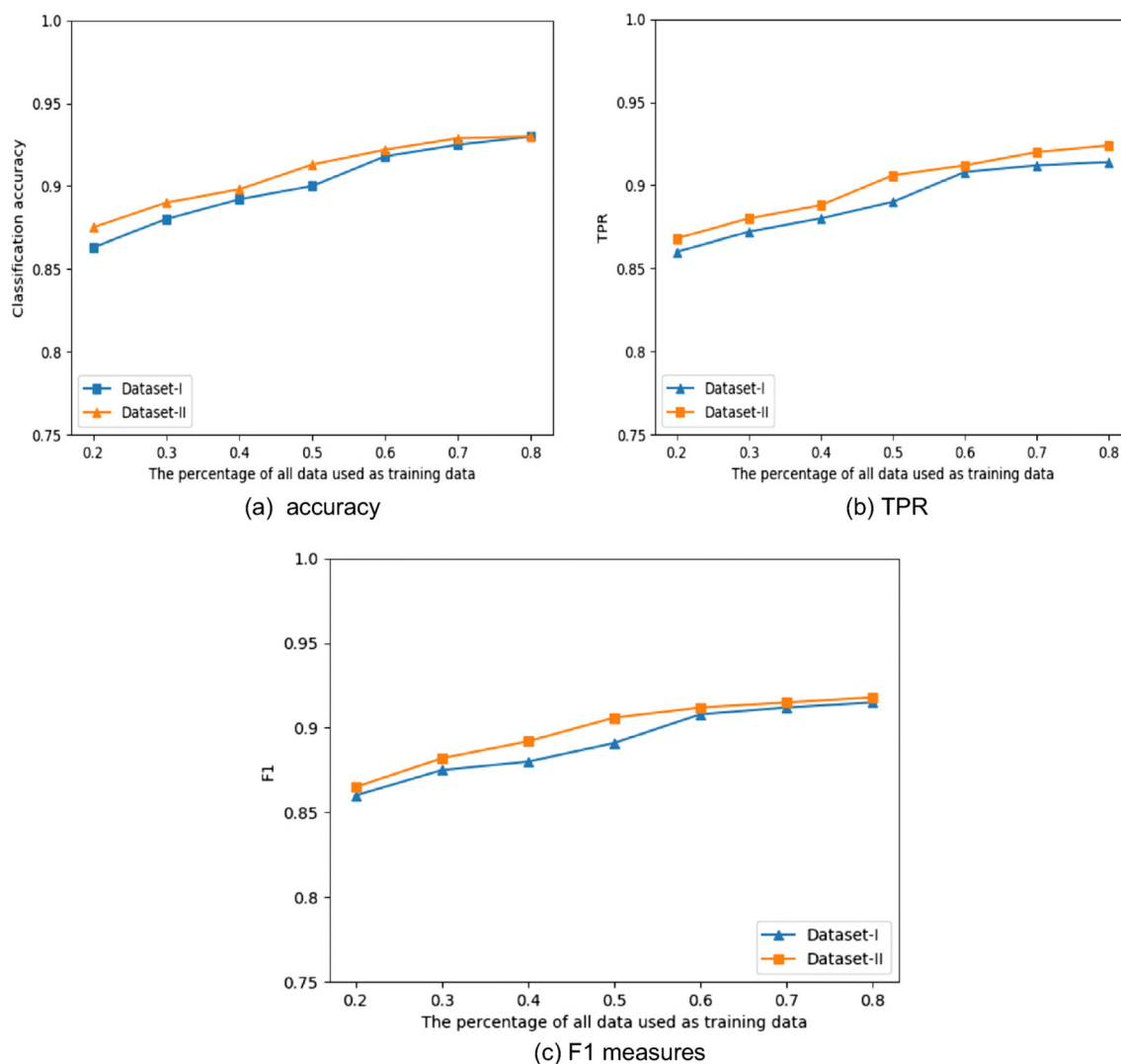


Fig. 6 a–c Detection performances for MCNN-TFW on dataset I and dataset II. The x -axis represents the percentage of all data used as training data

4.3 Comparison of MCNN-TFW with other advanced defense methods

4.3.1 RQ 2: does MCNN-TFW outperform the baseline approaches in term of accuracy?

To show the performance of our framework compared with state-of-art detection systems, we investigated the similar approaches that have been previously proposed. In this section, we compare the accuracy of MCNN-TFW with the three baseline approaches that are briefly introduced as below:

1. LIWC. Based on the work of Ott et al. [8], the first baseline we propose is based on using LIWC (linguistic inquiry and word count) features for text analysis. LIWC is a widely used lexicon in social science studies proposed by Pennebaker et al. [51–53].

2. CNN. Convolutional neural networks have achieved state-of-the-art in text classification tasks and based on the work of Wang [23] which demonstrates superior performance of CNN over recurrent neural architectures like the bidirectional LSTM (long short-term memory) for fake news detection, we choose CNN for comparison.
3. RST. We extract a set of RST relations using the implementation of the method proposed by Ji et al. [54]. Then, we vectorize the relations and employ SVM for classification. This baseline proposed by Rubin et al. [55] takes into account the hierarchical structure of documents via RST [56].

Experimental results show the detection performances for MCNNTFW and three baseline approaches on dataset I and dataset II. We use accuracy as the metric of performance evaluation given. Table 4 shows the comparison

Table 4 Classification accuracy of MCNN-TFW and three baseline approaches on dataset I and dataset II

Models	Dataset I (%)	Dataset II (%)
RST	68.03	70.08
LIWC	70.87	72.26
CNN	86.03	87.23
MCNN-TFW	91.67	92.08

results on dataset I and dataset II, and we make the following observations:

1. The poor performance of RST is because of the following reasons: (a) using RST without an annotated corpus is not very effective and (b) RST relations are extracted using auxiliary stools optimized for other corpora which cannot be applied effectively to the fake news corpus in hand. Note that annotating RST for our corpus is extremely unscalable and time consuming.
2. CNN achieves a better performance than LIWC. In line with the previous study, this shows that for fake news detection, taking into account the text is represented at the word level and fed to the CNN that extracts semantic representation of the detection of fake news do is more effective than employing the existing pre-defined dictionaries as LIWC does.
3. MCNN-TFW outperforms CNN with the proposed multi-level representation. Single-layer CNN built over word-level article representations can only utilize combinations of several nearby words. However, by first condensing word-level information into each sentence, then deriving

sentence-level representation for the news article, higher level semantic information can be extracted more effectively.

Answer to RQ 2 MCNN-TFW outperforms the other methods compared against including RST, LIWC, and CNN due to its ability to effectively capture semantic information from the article text content. Moreover, on the one hand, MCNN introduced the local convolutional features as well as the global semantics features. On the other hand, TFW further improves the performance of MCNN and pushed the accuracy even higher.

4.4 Evaluation of MCNN in MCNN-TFW

4.4.1 RQ 3: what is the effectiveness and significance of MCNN in MCNN-TFW?

In this section, based on the dataset of dataset II above, we further validate the effectiveness and significance of our proposed MCNN in detecting fake news. We compare MCNN-TFW with a classifier built directly using MCNN. The latter does not use the TFW we proposed. Instead, it uses the MCNN model to detect fake news in Section 3.2. We trained a MCNN classifier to achieve an accuracy of 90.21%, which is close to the state of the art. With TFW, the accuracy of ARNDroid is increased to 92.08%. This small reduction is negligible. We evaluated the performance of MCNN-TFW and MCNN on dataset II. The experimental results are shown in Fig. 7.

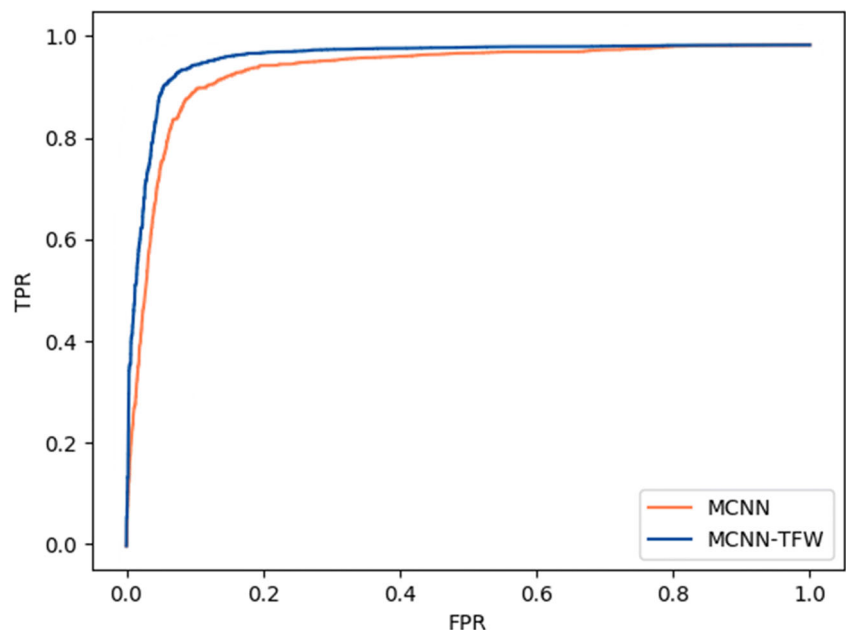
Fig. 7 Detection performances for MCNN-TFW and MCNN

Figure 7 shows the detection performance of the tenfold cross-validations for MCNN-TFW and MCNN on dataset II. The ROC curve indicates that both MCNN-TFW and MCNN achieve a high value for TPR and a low value for FPR. In particular, the AUC values of MCNN-TFW and MCNN are near 0.96 and 0.94, respectively, which indicates that they have similar detection effects. However, it is worth noting that for ARNDroid, the detection performance yields a TPR of 0.914 at an FPR of 0.008.

Answer to RQ 3 MCNN represents each article in the local convolutional features as well as the global semantics features and is able to apply detection on fake news articles.

4.5 Efficiency and scalability of MCNN-TFW

4.5.1 RQ 4: can MCNN-TFW work efficiently and be scalable for a large number of article?

The number of both real and fake news is growing very quickly making it increasingly important that fake news analysis scales so that such fake news does not remain undetected long enough to do major damage or even any damage. Therefore, we systematically evaluate the performance of our developed system MCNN-TFW, including scalability and detection effectiveness.

We first compute the runtime overhead of the three main phases of MCNN-TFW, WV (the complete vocabulary set) construction, MCNN training, and testing. In particular, once the MCNN are trained, it is almost instantaneous to use it to detect fake news. Therefore, for the evaluation of MCNN-TFW efficiency, we mainly measure the WV construction and MCNN training running time on dataset I and dataset II.

Table 5 describes the proposed MCNN-TFW construction WV and training MCNN execution time (in minutes). It is worth noting that the WV construction step can be processed in parallel on multiple servers, so the total time overhead can be greatly reduced if hardware conditions permit when there is a lot of data.

Finally, we show the detection stability of MCNN-TFW, with different sizes of sample sets in Fig. 8. From the results, we can conclude that our developed system MCNN-TFW is feasible in practical use for fake news detection.

Answer to RQ 4 The low runtime overhead allows MCNN-TFW to work efficiently and be scalable to a large number of news.

4.6 General applicability of MCNN-TFW

4.6.1 RQ 5: whether the method we proposed has general applicability?

In the following experiments, we examine the generality of our proposed method by applying it to the Weibo and NewsFN fake news detection tasks. Similar to the experiments implemented on fake news in cultural communication, we also implement two groups of experiments, one for presenting the classification accuracy on each dataset and another for comparing between different baseline technologies.

As shown in Fig. 9, we show the performance of MCNN-TFW fake news detection on Weibo and NewsFN. The trend of accuracy is consistent with that found in the experiments (Section 4.2). The classification accuracy on Weibo is 88.82%, and the accuracy on NewsFN is 90.10%.

As is shown in Table 6, we show measures of classification accuracy of our proposed MCNN-TFW and the above three baseline methods on the Weibo and NewsFN datasets. The experimental results show the following:

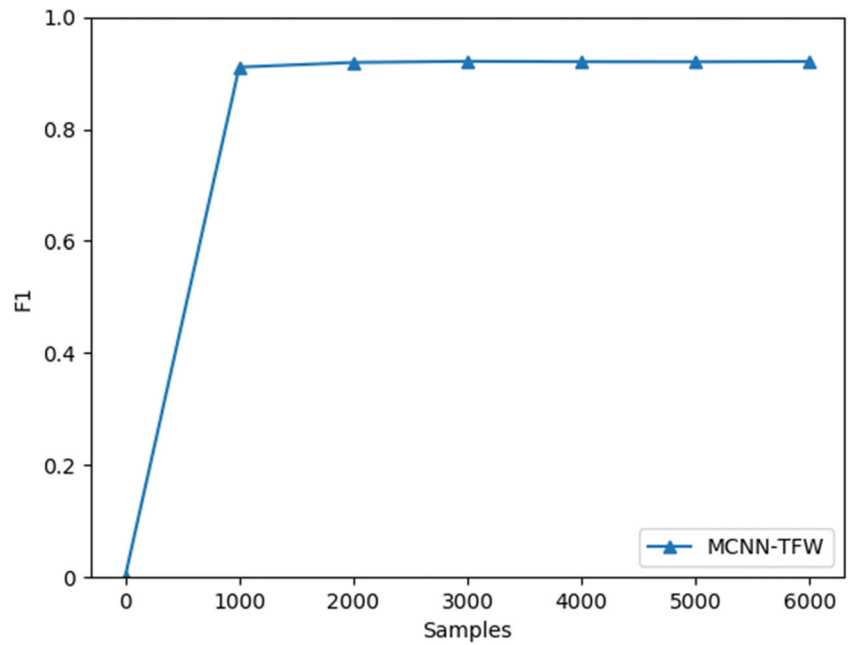
1. MCNN-TFW has higher on the accuracy than the other three comparison methods, which indicates that the performance of MCNN-TFW fake news detection is significantly better than other models.
2. The RST model exhibits poor performance on both Weibo and NewsFN datasets.
3. Compared with the LIWC model, the accuracy of CNN is significantly better, which has increased by 18.18% and 19.30% on Weibo and NewsFN datasets respectively.

Answer to RQ 5 MCNN-TFW has higher on the accuracy than the other three comparison methods on Weibo and NewsFN datasets, which demonstrates that our proposed method has the general applicability.

Table 5 WV construction and MCNN training runtimes in minutes

Dataset	WV construction	MCNN training	Total
Dataset I	5	4	9
Dataset II	5	6	11

Fig. 8 Stability evaluation of MCNN-TFW



5 Conclusion

we first proposed multi-level convolutional neural network (MCNN), which introduced the local convolutional features as well as the global semantics features, to effectively capture semantic information from article texts which can be used to classify the news as fake or not. We then employed a method of calculating the weight of sensitive words (TFW), which has shown their stronger importance with their fake or true labels. Finally, we develop MCNN-TFW, a multiple-level convolutional neural

network-based fake news detection system, which is combined to perform fake news detection in that MCNN extracts article representation and TFW calculates the weight of sensitive words for such news.

Our extensive evaluation results show that MCNN-TFW outperforms the state-of-the-art approaches in terms of accuracy and efficiency. Our proposed method on the detection of fake news in cultural communication paradigm can also be nearly applied to other fake news detection tasks. Future work will include studying the performance of our approach in a wider range of applications.

Fig. 9 Classification accuracy on NewsFN and Weibo datasets

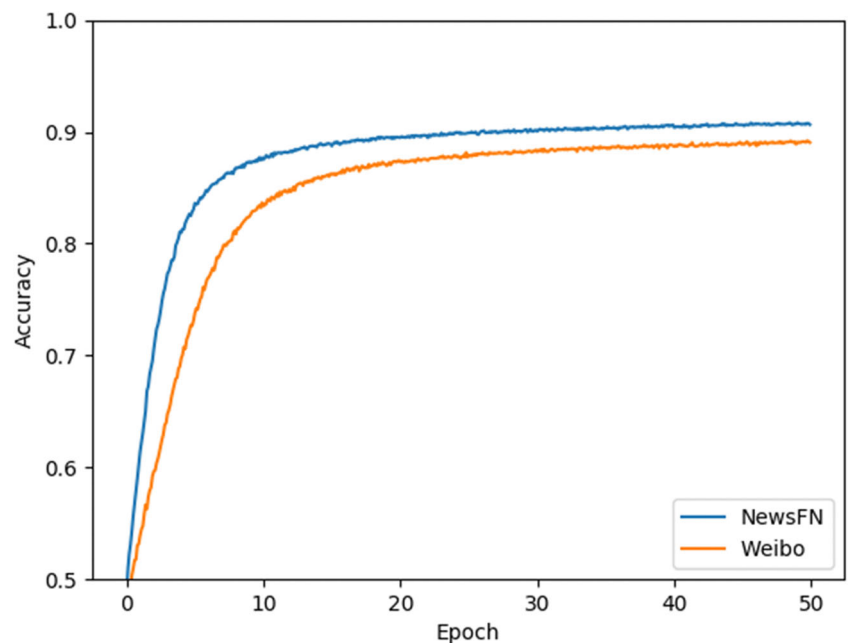


Table 6 Classification accuracy of MCNN-TFW and three baseline approaches on Weibo and NewsFN datasets

Models	Weibo (%)	NewsFN (%)
RST	62.08	64.18
LIWC	64.27	65.26
CNN	84.05	84.56
MCNN-TFW	88.82	90.10

Funding This work is supported by the National key R&D Program of China under Grant No. 2018YFB0203901, 2016YFB1000604, and 2018YFB1402700 and the Key Research and Development Program of Shaanxi Province (No. 2018ZDXM-GY-036).

References

- Fuller CM, Biros DP, Wilson RL (2009) Decision support for determining veracity via linguistic-based cues. *Decis Support Syst* 46(3):695–703
- Morstatter F, Kumar S, Liu H, Maciejewski R (2013) Understanding twitter data with tweetexplorer. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1482–1485)
- Shu K, Mahudeswaran D, Wang S, Lee D, Liu H (2018). Fakenewsnet: A data repository with news content, social context and spatio-temporal information for studying fake news on social media. *arXiv preprint arXiv:1809.01286*
- Shu K, Mahudeswaran D, Liu H (2019). FakeNewsTracker: a tool for fake news collection, detection, and visualization. *Computational and Mathematical Organization Theory* 25(1):60–71
- Bogaard G, Meijer EH, Vrij A, Merckelbach H (2016) Scientific Content Analysis (SCAN) cannot distinguish between truthful and fabricated accounts of a negative event. *Front Psychol* 7(2016):243
- Nahari G, Vrij A, Fisher RP (2012) Does the truth come out in the writing? Scan as a lie detection tool. *Law Human Behavior* 36(1):68
- Castillo C, Mendoza M, Poblete B (2011) Information credibility on twitter. In *Proceedings of the 20th international conference on World Wide Web. ACM*, 675–684. 134:A635–A646, Dec. 1965
- Ott M, Cardie C, Hancock JT (2013) Negative deceptive opinion spam. In *HLT-NAACL*. 497–501
- Ott M, Choi Y, Cardie C, Hancock JT (2011) Finding deceptive opinion spam by any stretch of the imagination. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics*, pp 309–319
- Blunsom P, Grefenstette P, Kalchbrenner N (2014) A convolutional neural network for modelling sentences. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*
- Kimura M, Saito K, Motoda H (2009) Efficient estimation of influence functions for SIS model on social networks. In *IJCAI*. 2046–2051
- Yih W-t, He X, Meek C (2014) Semantic parsing for single-relation question answering. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2:643–648
- Kim Y (2014) Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*
- Mihalcea R, Strapparava C (2009) The lie detector: explorations in the automatic recognition of deceptive language. In: *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers. Association for Computational Linguistics*, pp 309–312
- Johnson M (1998) PCFG models of linguistic tree representations. *Comput Linguist* 24(4):613–632
- Dou H, Qi Y, Wei W, Song H (2016) A two-time-scale load balancing framework for minimizing electricity bills of internet data centers [J]. *Pers Ubiquit Comput* 20(5):681–693
- Feng S, Banerjee R, Choi Y (2012) Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2. Association for Computational Linguistics*, pp 171–175
- LeCun Y, Bottou L, Bengio Y, Patrick H (1998) Gradient-based learning applied to document recognition. *Proc IEEE* 86(11):2278–2324
- Qi S, Zheng Y, Li M, Liu Y, Qiu J (2016) Scalable industry data access control in RFID-enabled supply chain. *IEEE/ACM Trans Netw (ToN)* 24(6):3551–3564, 3,376
- Qi S, Zheng Y, Li M, Lu L, Liu Y (2016) Secure and private RFID-enabled third-party supply chain systems. *IEEE Trans Comput (TC)* 65(11):3413–3426 2,916
- Qi S, Zheng Y Crypt-DAC: cryptographically enforced dynamic access control in the Cloud, *IEEE Transactions on Dependable and Secure Computing*, 29 March 2019. <https://doi.org/10.1109/TDSC.2019.2908164>
- Xi M, Qi Y, Wu K, Zhao J, Li M (2011) Using potential to guide mobile nodes in wireless sensor networks. *Ad Hoc & Sensor Wireless Networks* 12(3–4):229–251
- Wang WY (2017) “Liar, Liar pants on fire”: a new benchmark dataset for fake news detection. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Association for Computational Linguistics*, pp 422–426
- Qian F, Gong C, Sharma K, Liu Y (2018) Neural user response generator: fake news detection with collective user intelligence. In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18. International Joint Conferences on Artificial Intelligence Organization*, pp 3834–3840
- Rumelhart DE, Hinton GE, Williams RJ et al (1988) Learning representations by back-propagating errors. *Cogn Model* 3(1988):1
- Yan J, Qi Y, Rao Q (2018) Detecting malware with an ensemble method based on deep neural network. <https://doi.org/10.1155/2018/7247095>, UNSP 7247095
- Chen P, Qi Y, Li X, Hou D, Lyu MR-T (2016) ARF-Predictor: effective prediction of aging-related failure using entropy. <https://doi.org/10.1109/TDSC.2016.2604381>
- Wang X, Qi Y, Wang Z et al Design and implementation of SecPod: a framework for virtualization-based security systems. <https://doi.org/10.1109/TDSC.2017.2675991>
- Hochreiter S, Jürgen S (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780

30. Wang P, Qi Y, Liu X (2014) Power-aware optimization for heterogeneous multi-tier clusters, pages 2005–2015
31. Sun Z, Song H, Wang H, Fan X Energy balance-based steerable arguments coverage method in WSNs. *IEEE Access* 2017 Mar 20, Issue: 99. 6: 33766–33773. <https://doi.org/10.1109/ACCESS.2017.2682845>
32. Rashkin H, Choi E, Jang JY, Volkova S, Choi Y (2017) Truth of varying shades: analyzing language in fake news and political fact-checking. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp 2921–2927
33. Volkova S, Shaffer K, Jang JY, Hodas N (2017) Separating facts from fiction: linguistic models to classify suspicious and trusted news posts on twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vol. 2. 647–653
34. Zheng P, Qi Y, Zhou Y, Chen P, Zhan J, L yu MR-T (2014) An automatic framework for detecting and characterizing the performance degradation of software systems. 63:927–943
35. Chopra S, Jain S, Sholar JM (2017) Towards automatic identification of fake news: Headline-article stance detection with LSTM attention models
36. Kim Y (2014) Convolutional neural networks for sentence classification. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751
37. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119
38. Wu HC, Luk RWP, Wong KF, Kwok KL (Jun. 2008) Interpreting TF-IDF term weights as making relevance decisions. *ACM Trans Inf Syst* 26(3):13
39. Aizawa A (2003) An information-theoretic perspective of TF-IDF measures. *Inf Process Manage* 39(1):45–65
40. Qiao Y-n, Yong Q, Di H (2001) Tensor field model for higher-order information retrieval. 84(12):2303–2313
41. Wei W, Yong Q (2011) Information potential fields navigation in wireless ad-hoc sensor networks. *Sensors* 11(5):4794–4807
42. Xu Q, Wang L, Hei XH, Shen P, Shi W, Shan L (2014) GI/Geom/1 queue based on communication model for mesh networks. *Int J Commun Syst* 27(11):3013–3029
43. Yang XL, Shen PY et al (2012) Holes detection in anisotropic sensor networks: topological methods [J]. *Int J Distrib Sens Netw* 8(10):135054
44. Song H, Li W, Shen P, Vasilakos A (2017) Gradient-driven parking navigation using a continuous information potential field based on wireless sensor network. *Inf Sci* 408(C):100–114. <https://doi.org/10.1016/j.ins.2017.04.042>
45. Qiang Y, Zhang J (2013) A bijection between lattice-valued filters and lattice-valued congruences in residuated lattices. *Math Probl Eng* 36(8):4218–4229
46. Yang XL, Zhou B, Feng J, Shen PY (2012) Combined energy minimization for image reconstruction from few views. *Math Probl Eng* 2012
47. Srivastava HM, Zhang Y, Wang L, Shen P, Zhang J (2014) A local fractional integral inequality on fractal space analogous to Anderson’s inequality[C]//*Abstract and Applied Analysis*. Hindawi Publishing Corporation, 46(8): 5218–5229, Article number: 797561, <https://doi.org/10.1155/2014/797561>, WOS: 000339756400001
48. Zhang J, Song H, Wan Y (2018) Big data analytics enabled by feature extraction based on partial independence. *Neurocomputing* 288:3–10
49. Ma J, Gao W, Mitra P, Kwon S, Jansen BJ, Wong K-F, Cha M (2016) Detecting rumors from microblogs with recurrent neural networks. In: *IJCAI*, pp 3818–3824
50. Ma J, Gao W, Wong K-F (2017) Detect rumors in microblog posts using propagation structure via kernel learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. pp 708–717
51. Pennebaker JW, Boyd RL, Jordan K, Blackburn K (2015) The development and psychometric properties of LIWC2015. Technical Report
52. Ren Y, Zhang Y (2016) Deceptive opinion spam detection using neural network. In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 140–150
53. Zhang J, WeiWei DP, Woźniak M, Kośmider L, Damaševičius R (2019) A neuro-heuristic approach for recognition of lung diseases from X-ray images. Author links open overlay panel. *Exp Syst Appl* 126:218–232
54. Ji Y, Eisenstein J (2014) Representation learning for text-level discourse parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 13–24
55. Rubin VL, Lukoianova T (2015) Truth and deception at the rhetorical structure level. *J Assoc Inf Sci Technol* 66(5):905–917
56. Wang X, Qi Y, Wang Z et al (2019) Design and implementation of SecPod: a framework for virtualization-based security systems. 16(1):44–57

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.