Adaptive Interaction Fusion Networks for Fake News Detection

Lianwei Wu[1] and **Yuan Rao**[*2]

Abstract. The majority of existing methods for fake news detection universally focus on learning and fusing various features for detection. However, the learning of various features is independent, which leads to a lack of cross-interaction fusion between features on social media, especially between posts and comments. Generally, in fake news, there are emotional associations and semantic conflicts between posts and comments. How to represent and fuse the cross-interaction between both is a key challenge. In this paper, we propose Adaptive Interaction Fusion Networks (AIFN) to fulfill cross-interaction fusion among features for fake news detection. In AIFN, to discover semantic conflicts, we design gated adaptive interaction networks (GAIN) to capture adaptively similar semantics and conflicting semantics between posts and comments. To establish feature associations, we devise semantic-level fusion self-attention networks (SFSN) to enhance semantic correlations and fusion among features. Extensive experiments on two real-world datasets, i.e., RumourEval and PHEME, demonstrate that AIFN achieves the state-of-the-art performance and boosts accuracy by more than 2.05% and 1.90%, respectively.

1 Introduction

Owing to the low cost of information dissemination and the lack of timely and effective supervision, social media provides an ideal breeding ground for the growth of fake news (a.k.a. hoaxes, rumors, etc.). Research indicates that fake news has dominated the news cycle since the US presidential election (2016) [3, 9]. In detail, a tweeting rate for users tweeting links to websites containing news classified as fake more than four times larger than for traditional media and 25% of tweets during the election in Twitter spread fake news [4], meanwhile, 1% of users are even exposed to 80% of fake news [9]. Furthermore, fake news has greater virality than true information, which diffuses remarkably farther, faster, deeper, and more broadly than the truth [13]. Therefore, how to effectively evaluate information credibility has become a crucial problem, drawing wide attention from academic and industrial communities.

Currently, most existing studies based on deep neural networks capture various features for fake news detection, which have
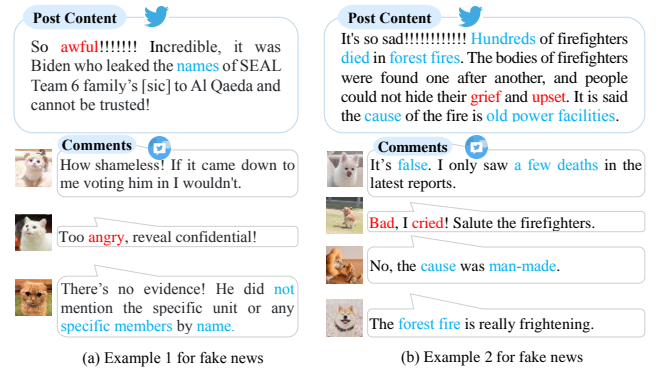

(a) Example 1 for fake news   (b) Example 2 for fake news

Figure 1. Two fake news from Twitter. Blue denotes semantic correlations between comments and posts while red means emotional correlations between comments and posts.

achieved great success. Generally, they devise different modules to capture different credibility features including news content [11, 2], diffusion patterns [16, 15], and social context (a.k.a. meta-data) [33, 31], and then use some traditional fusion strategies to integrate these features for evaluation. For instance, Wang [26] adopts CNN to capture content semantics and employs BiLSTM to learn meta-data features. Finally, the author fuses them by concatenation for fake news detection. Zhang et al. [33] obtain semantics of news and comments by different BiLSTM combining Bayesian networks then build simple multi-layer perception to fuse them for misinformation detection. Wu et al. [28] rely on adversarial networks and multi-task learning to capture differential credibility features from news content semantics and fuse them for information credibility evaluation.

However, there are still several general limitations of the above methods. **First**, feature fusion is relatively shallow. Most fusion strategies are based on concatenation, addition, or simple neural networks, which are hard to choose noteworthy features and even bring a certain amount of noise. **Secondly**, various features are typically fused only in the final evaluation stage while lacking cross-interaction fusion in the intermediate semantic learning stage. We know that cross interaction is ubiquitous on social media, especially there are feature correlations and semantic conflicts between posts and comments in fake news. Figure 1 shows two concrete examples to illustrate this phenomenon. We observe that: 1) Emotions of comments usually correspond to these of posts (in red), like 'cried' and 'grief'; and 2) Some comments have semantic conflicts with posts (in blue), like 'power facilities' and 'man-made'. Cross-interaction fusion among features can enhance core semantics and help to discover the causes of fake news. Therefore, how to represent and fuse

---

[1] 1) Lab of Social Intelligence and Complex Data Processing, Software School, Xi'an Jiaotong University, Xi'an, China; 2) Shannxi Joint Key Laboratory for Artifact Intelligence (Sub-Lab of Xi'an Jiaotong University), Xi'an, China; 3) Research Institute of Xi'an Jiaotong University, Shenzhen, China, email: stayhungry@mail.xjtu.edu.cn

[2] 1) Lab of Social Intelligence and Complex Data Processing, Software School, Xi'an Jiaotong University, Xi'an, China; 2) Shannxi Joint Key Laboratory for Artifact Intelligence (Sub-Lab of Xi'an Jiaotong University), Xi'an, China; 3) Research Institute of Xi'an Jiaotong University, Shenzhen, China, email: raoyuan@mail.xjtu.edu.cn

* Corresponding author

these interactions is a major challenge for fake news detection.

To address the above problems, we propose **A**daptive **I**nteraction **F**usion **N**etworks (henceforth, AIFN) to cross-interactively fuse various features for fake news detection. Specifically, in order to discover semantic conflicts between posts and comments, we design gated adaptive interaction networks (GAIN) to acquire adaptively similar and conflicting semantics between both, in which conflicting gate is designed for capturing differential features while refining gate aims to obtain similar semantics between posts and comments. To effectively utilize feature associations within posts and comments respectively, we explore semantic-level fusion self-attention networks (SFSN) to screen their valuable features and fuse them deeply. Experimental results show that AIFN achieves better performance than other state-of-the-art methods and gains new benchmarks. The main contributions of our work can be summarized as follows:

- Proposed gated adaptive interaction networks fulfill global feature interaction, which focuses on discovering conflicting semantics more accurately by capturing adaptively similar semantics and differential semantics (Section 4.4.3 and 4.5).
- Explored semantic-level fusion self-attention networks can achieve deep semantic fusion within posts and comments, which effectively build feature associations between both (Section 4.4.2 and Section 4.5).
- Experiments on two public, widely used fake news datasets demonstrate that our method significantly outperforms previous state-of-the-art methods (Section 4.3). We release the source code publicly for further research[3].

The rest of the paper is organized as follows. We start with an overview of related work in Section 2. Section 3 presents the details of our approach. Experiment results and discussions are given in Section 4, and Section 5 concludes our work.

2 Related Work

2.1 Fake News Detection

Existing studies for fake news detection can be roughly summarized into two categories. The first category is to extract or construct plenty of features with manual ways [5, 34]. The prominent merits of this category are low computational complexity and many effective meta-data features discovered from social context.

Instead of gaining features by labor-intensive manual design, the second category is to automatically capture credibility features based on deep neural networks. There are two ways: One is to capture linguistic features from text content, such as semantics [30, 22, 17], emotions [2], writing styles [21], and stances [27]. But the capture of stances might bring some noise and reduce some performance indicators of models. The other way emphasizes on capturing meta-data features surrounding social context such as source-based [25], post-based [29, 33], comment-based [23], user-based [24], and propagation-based [16]. In this work, combined with the pros of both categories, we capture deeply semantic and emotional features from posts and comments.

2.2 Feature Fusion

Generally, most feature fusion methods are based on concatenation [18], point-wise addition [14], and multi-layer perceptions [33], etc.

For example, Nguyen et al. [18] concatenate document features obtained by TF-IDF, semantics obtained by word2vec, and user-event graph features obtained by node2vec as the input layer of the model for fake news detection. Nevertheless, these fusion methods usually lack feature interaction. To address it, attention mechanism [24, 32] is developed to focus on the interaction of two related features, like user profiles and behaviors, posts and comments. Typically, Shu et al. [23] present a sentence-comment co-attention network to exploit both news contents and comments to jointly detect explainable check-worthy sentences and comments for fake news detection. But attention-based methods are mostly based on the interaction of local and partial features, which lack global cross-interaction of all features. To address these limitations, we propose cross-interaction fusion model integrating self-attention networks for evaluation.

3 Adaptive Interaction Fusion Networks

The architecture of AIFN with five components is illustrated in Figure 2 and each component of AIFN is described as follows:

3.1 Input Feature Descriptions

AIFN learns four types of features around posts and comments from the perspectives of word and emotion.

3.1.1 BERT Embeddings

Word embeddings of posts and comments in tweets are both applied BERT embeddings. BERT embeddings of a post containing l words are indicated as $\mathbf{X}^p = \{[\mathbf{x}_1^p; \mathbf{pos}_1^p], [\mathbf{x}_2^p; \mathbf{pos}_2^p], ..., [\mathbf{x}_l^p; \mathbf{pos}_l^p]\}$, $\mathbf{X}^p \in \mathbb{R}^{l \times (d+l)}$, where embeddings of each word are concatenated by word embeddings $\mathbf{x}_i^p \in \mathbb{R}^d$ and position embeddings $\mathbf{pos}_i^p \in \mathbb{R}^l$. $\mathbf{x}_i^p$ of word i is a d-dimensional vector obtained by pre-trained BERT model [8]. $\mathbf{pos}_i^p$ of word i is achieved by one-hot encoding. ; means concatenation operation. For BERT embeddings of comments, we first rank all comments in chronological order and then concatenate them into a sequence with k words. The way of embeddings is the same as the post, i.e., $\mathbf{X}^c = [\mathbf{x}_1^c; \mathbf{pos}_1^c], [\mathbf{x}_2^c; \mathbf{pos}_2^c], ..., [\mathbf{x}_k^c; \mathbf{pos}_k^c]$, $\mathbf{X}^c \in \mathbb{R}^{k \times (d+k)}$.

3.1.2 Emotion Embeddings

We represent D-dimensional emotion embeddings of one word via pre-trained emotion word embeddings (EWE) [1], which learns emotion-enriched word representations by projecting emotionally similar words into neighboring spaces based on a large training dataset with a rich spectrum of emotions. Here, emotion embeddings of a post $\mathbf{E}^p$ and comments $\mathbf{E}^c$ are denoted as $\mathbf{E}^p = \{\mathbf{e}_1^p, \mathbf{e}_2^p, ..., \mathbf{e}_l^p\}$ and $\mathbf{E}^c = \{\mathbf{e}_1^c, \mathbf{e}_2^c, ..., \mathbf{e}_k^c\}$, respectively, where $\mathbf{e}_i^p \in \mathbb{R}^D$, $\mathbf{E}^p \in \mathbb{R}^{l \times D}$, $\mathbf{e}_i^c \in \mathbb{R}^D$ and $\mathbf{E}^c \in \mathbb{R}^{k \times D}$.

3.2 Feature Encoding

We encode word-based and emotion-based sequence via BiLSTM, which can not only capture long-term dependencies by a persistent memory compared with RNN but also obtain contextual information by modeling word sequence from both directions of words. BiLSTM contains a forward LSTM $\overrightarrow{\mathbf{LSTM}}$ and a backward LSTM $\overleftarrow{\mathbf{LSTM}}$, which learns word sequence from the first word to the last word and learns the sequence in reverse order respectively.

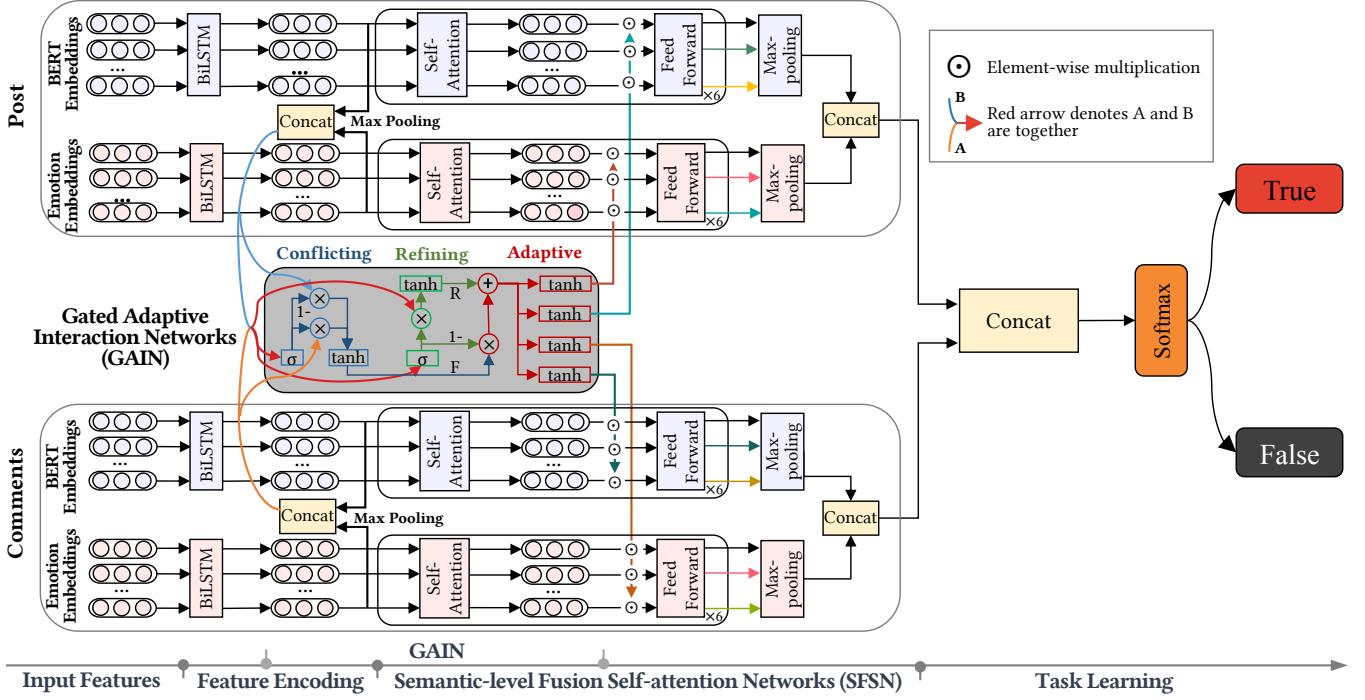[3] https://www.dropbox.com/s/9pos1pvv392adnt/interactFusions.zip?dl=0

Figure 2. Overall architecture of AIFN. The model utilizing organically four types of features focuses on interaction and fusion among features by the following components: input features, feature encoding, semantic-level fusion self-attention networks, gated adaptive interaction networks, and task learning.

$$\overrightarrow{\mathbf{h}_i} = \overrightarrow{\mathbf{LSTM}}(\overrightarrow{\mathbf{h}_{i-1}}, \mathbf{x}_i) \qquad (1)$$

$$\overleftarrow{\mathbf{h}_i} = \overleftarrow{\mathbf{LSTM}}(\overleftarrow{\mathbf{h}_{i+1}}, \mathbf{x}_i) \qquad (2)$$

$$\mathbf{h}_i = \overrightarrow{\mathbf{h}_i} \oplus \overleftarrow{\mathbf{h}_i} \qquad (3)$$

where $\overrightarrow{\mathbf{h}_i} \in \mathbb{R}^h$ and $\overleftarrow{\mathbf{h}_i} \in \mathbb{R}^h$ are hidden states of forward and backward LSTM at position i, respectively. $\mathbf{x}_i$ is the i-th input which can be replaced by word embeddings $\mathbf{X}_i^p$ and $\mathbf{X}_i^c$, or emotion embeddings $\mathbf{E}_i^p$ and $\mathbf{E}_i^c$. $\oplus$ denotes concatenation. In addition, experiments confirm BiLSTM can be replaced by BiGRU for comparable performance in AIFN.

3.3 Gated Adaptive Interaction Networks (GAIN)

We first obtain the pooled vectors $\mathbf{X}_{pool}^{pw}$, $\mathbf{X}_{pool}^{pe}$, $\mathbf{X}_{pool}^{cw}$, and $\mathbf{X}_{pool}^{ce}$ (all $\in \mathbb{R}^l$) from the encoded word-level and the encoded emotion-level features in posts and comments respectively by max-pooling operation. Then we get the integrated post vector $\mathbf{X}_{pool}^p$ and comment vector $\mathbf{X}_{pool}^c$ by concatenation.

$$\mathbf{X}_{pool}^p = \mathbf{X}_{pool}^{pw} \oplus \mathbf{X}_{pool}^{pe} \qquad (4)$$

$$\mathbf{X}_{pool}^c = \mathbf{X}_{pool}^{cw} \oplus \mathbf{X}_{pool}^{ce} \qquad (5)$$

To discover semantic conflicts between posts and comments, we devise GAIN to capture adaptively similar semantics and differential semantics.

3.3.1 Conflicting Gate

Conflicting gate is to obtain differential and conflicting features between posts and comments, i.e., $\mathbf{X}_{pool}^p$ and $\mathbf{X}_{pool}^c$, respectively.

$$\mu_f = \sigma(\mathbf{X}_{pool}^p \mathbf{W}_{f1} + \mathbf{X}_{pool}^c \mathbf{W}_{f2} + \mathbf{b}_f) \qquad (6)$$

$$\mathbf{F} = \tanh(\mathbf{X}_{pool}^p \odot \mu_f \mathbf{W}_{h1} + \mathbf{X}_{pool}^c \odot (1-\mu_f)\mathbf{W}_{h2} + \mathbf{b}_h)(7)$$

where $\mathbf{F}$ is the conflicting features obtained. All weights $\mathbf{W} \in R^{2l \times 2l}$ and biases $\mathbf{b}_f, \mathbf{b}_h \in R^{2l}$.

When μ_f approaches 0, the conflicting features in $\mathbf{X}_{pool}^p$ are more obtained. Conversely, more conflicting features in $\mathbf{X}_{pool}^c$ are obtained. This way ensures the adjustability of the conflicting gate so that all conflicting features of both have the possibility of being obtained.

3.3.2 Refining Gate

We design refining gate to capture more similar features between posts and comments $\mathbf{X}_{pool}^p$ and $\mathbf{X}_{pool}^c$:

$$\mu_r = \sigma(\mathbf{X}_{pool}^p \mathbf{W}_{r1} + \mathbf{X}_{pool}^c \mathbf{W}_{r2} + \mathbf{b}_r) \qquad (8)$$

$$\mathbf{R} = \tanh(\mathbf{X}_{pool}^p \odot \mu_r \mathbf{W}_{rp} + \mathbf{X}_{pool}^c \odot \mu_r \mathbf{W}_{rc} + \mathbf{b}_{rr}) \qquad (9)$$

where all weights $\mathbf{W} \in R^{2l \times 2l}$ and biases $\mathbf{b} \in R^{2l}$.

3.3.3 Adaptive Mechanism

The location of semantic conflicts is inseparable from the matching of highly similar semantics in sequences, in order to accurately locate the differential features in a sequence, we develop adaptive mechanism to capture adaptively the similar and conflicting semantics.

$$
\begin{aligned}
\mathbf{S} &= \mathbf{R} + (1 - \mu_r) \odot \mathbf{F} & (10) \\
\mathbf{t}^{px} &= \tanh(\mathbf{W}^{px}\mathbf{S} + \mathbf{b}^{px}) & (11) \\
\mathbf{t}^{pe} &= \tanh(\mathbf{W}^{pe}\mathbf{S} + \mathbf{b}^{pe}) & (12) \\
\mathbf{t}^{cx} &= \tanh(\mathbf{W}^{cx}\mathbf{S} + \mathbf{b}^{cx}) & (13) \\
\mathbf{t}^{ce} &= \tanh(\mathbf{W}^{ce}\mathbf{S} + \mathbf{b}^{ce}) & (14)
\end{aligned}
$$

where $\mathbf{S} \in \mathbb{R}^{d_x}$ is the adaptive features. $\mathbf{t}^{px}$, $\mathbf{t}^{pe}$, $\mathbf{t}^{cx}$ and $\mathbf{t}^{ce}$ are different dimensional global interaction vectors suitable for different types of features, i.e., word-level in posts, emotion-level in posts, word-level in comments, and emotion-level in comments, respectively.

3.4 Semantic-level Fusion Self-attention Networks (SFSN)

We design SFSN to build associations and deep semantic fusion among features. Specifically, SFSN is used to fuse deeply different types of features and the interacted features obtained by GAIN (in Section 3.3). We take word-level in posts as an example to explain the details, where $\mathbf{t}^{px}$ is the global interaction vector and $\mathbf{X}^{px} = \{\mathbf{x}_1^{px}, \mathbf{x}_2^{px}, ..., \mathbf{x}_l^{px}\}$ is outputs of BiLSTM for word-level in posts.

Self-attention Networks   We leverage the multi-head self-attention networks to learn dependencies and semantics between any two words in one sequence. The scaled dot-product attention, the core of self-attention networks, is described as

$$
\textbf{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \textbf{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}})\mathbf{V} \qquad (15)
$$

where $\mathbf{Q} \in \mathbb{R}^{l \times 2h}$, $\mathbf{K} \in \mathbb{R}^{l \times 2h}$, and $\mathbf{V} \in \mathbb{R}^{l \times 2h}$ are query, key, and value matrices respectively and $\mathbf{Q} = \mathbf{K} = \mathbf{V} = \mathbf{X}^{px}$.

To get high parallelizability of attention, multi-head attention first linearly projects queries, keys, and values j times by different linear projections and then j projections perform the scaled dot-product attention in parallel. Finally, these results of attention are concatenated and once again projected to get the new representation. Formally, the multi-head attention can be formulated as:

$$
\begin{aligned}
\text{head}_i &= \textbf{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) & (16) \\
\mathbf{O}' &= \textbf{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) & (17) \\
&= \textbf{Concat}(\text{head}_1, \text{head}_2, ..., \text{head}_h)\mathbf{W}^o & (18)
\end{aligned}
$$

where $\mathbf{W}_i^Q \in \mathbb{R}^{2h \times d_k}$, $\mathbf{W}_i^K \in \mathbb{R}^{2h \times d_k}$, $\mathbf{W}_i^V \in \mathbb{R}^{2h \times d_k}$, and $\mathbf{W}^o \in \mathbb{R}^{2h \times 2h}$ are trainable parameters and d_k is $2h/j$.

Fusion Strategy We exploit the interaction vector $\mathbf{t}^{px}$ to concentrate on the fusion of deep semantics in self-attention networks for highlighting semantic relevance of different types of features.

$$
\mathbf{O}^{px} = \mathbf{t}^{px} \odot \mathbf{O}' \qquad (19)
$$

where $\odot$ is element-wise multiplication.

Subsequently, self-attention networks pass a feed forward network (FFN) for adding non-linear features while keeping scale-invariant features, which includes a single hidden layer with an ReLU.

$$
\mathbf{O}_{ffn}^{px} = \textbf{FFN}(\mathbf{O}^{px}) = \max(0, \mathbf{O}^{px}\mathbf{W}_1 + b_1)\mathbf{W}_2 + \mathbf{b}_2 \qquad (20)
$$

where $\mathbf{W}_1$, and $\mathbf{W}_2$ (both $\in \mathbb{R}^{2h \times 2h}$), $\mathbf{b}_1$, and $\mathbf{b}_2$ (both $\in \mathbb{R}^{2h}$) are learned parameters.

Finally, we apply max-pooling to each dimension across all words for gaining a fixed-size representation $\mathbf{X}^{pw}$ of the sequence as final outputs of word-level in posts.

$$
\mathbf{X}^{pw} = \textbf{max-pooling}(\mathbf{O}_{ffn}^{px}) \qquad (21)
$$

Additionally, we leverage concatenation to integrate the learned word-level features and emotion-level features from posts and comments, respectively, i.e., $\mathbf{X}^{pw}$, $\mathbf{X}^{pe}$, $\mathbf{X}^{cw}$, and $\mathbf{X}^{ce}$.

$$
\begin{aligned}
\mathbf{X}^p &= \mathbf{X}^{pw} \oplus \mathbf{X}^{pe} & (22) \\
\mathbf{X}^c &= \mathbf{X}^{cw} \oplus \mathbf{X}^{ce} & (23)
\end{aligned}
$$

where $\mathbf{X}^p$ and $\mathbf{X}^c$ are the final integrated features from posts and comments respectively.

3.5 Task Learning

We employ concatenation operation to integrate the output features of posts and comments, i.e., $\mathbf{X}^p$ and $\mathbf{X}^c$.

$$
\mathbf{X}^{pc} = \mathbf{X}^p \oplus \mathbf{X}^c \qquad (24)
$$

Then we apply the integrated features to Softmax function for task learning. Softmax function emits the prediction of probability distribution by the following equations:

$$
\mathbf{p} = \textbf{softmax}(\mathbf{W}_d\mathbf{X}^{pc} + \mathbf{b}_d) \qquad (25)
$$

We train the model to minimize cross-entropy error for a single training instance with ground-truth label $\mathbf{y}$:

$$
\textbf{Loss} = -\sum \mathbf{y}log\mathbf{p} \qquad (26)
$$

4 Experiments

4.1 Datasets and Evaluation Metrics

For experimental evaluation, we use two real-world benchmark datasets, RumourEval [7] and PHEME [35], which respectively contains 325 and 6,425 Twitter threads discussing rumors. Both datasets include Twitter conversation threads associated with different newsworthy events, like the Ferguson unrest, where one thread consists of a source tweet conveying a rumor and a tree of comments. The credibility of each tweet can be true, false, and unverified. Since our goal is to evaluate whether one tweet is true or false, we filter out unverified tweets. Table 1 gives statistics of the two datasets.

In consideration of the imbalance label distributions, evaluation solely relying on accuracy effortlessly achieve competitive performance beyond the majority class. Therefore, besides accuracy (A), we add precision (P), recall (R) and F1-score (F1) as complementary evaluation metrics for tasks. We divide the two datasets into training, validation, and testing subsets with proportion of 70%, 10%, and 20%.

4.2 Setting

We strictly turn all hyper-parameters on the validation dataset, and we achieve the best performance via a small grid search. The hyper-parameters turned on the validation subset are shown as follows:

Table 1. Statistics of the datasets

Subset	Veracity	RumourEval		PHEME	
		#posts	#comments	#posts	#comments
Training	True	83	1,949	861	24,438
	False	70	1,504	625	17,676
	Total	153	3,453	1,468	42,114
Validation	True	10	101	95	1,154
	False	12	141	115	1,611
	Total	22	242	210	2,765
Testing	True	9	412	198	3,077
	False	12	437	219	3,265
	Total	21	849	417	6,342

- Word embedding sizes d for posts and comments are both set to 768;
- Emotion embeddings size D is 300;
- The dimensionality of LSTM hidden state h is 120;
- The initial learning rate is set to 0.001;
- The dropout rate is 0.4;
- The number of projections j is 8;
- Attention heads and blocks are set to 6 and 4, respectively;
- The minibatch size is 64.

4.3 Performance Evaluation

We compare AIFN with the following baselines:

SVM [7] are used to detect misinformation based on manually extracted features.

CNN [6] with different convolutional window sizes captures content features similar to n-grams for rumor verification.

TE [10] leverages tensor decomposition to derive concise article embeddings to create an article-by-article graph for misinformation detection.

DeClarE [20] applies posts as the queries of attention mechanism to extract evidence from comments and aggregates post sources and languages.

MTL-LSTM [12] jointly trains three tasks, i.e., rumor detection, veracity classification, and stance classification, and learns the relationships among the tasks by common LSTM for improving performance of each task.

TRNN [16] contains two tree-structured models based on RNN, which learns credibility features based on non-sequential propagation structure formed by comments for rumor detection. In this work, we adopt the top-down model with better results as the baseline.

Bayesian-DL [33] employs Bayesian to represent the uncertainty of prediction for the veracity of a claim and then encodes all the people's comments to the claim through LSTM for misinformation detection.

The experimental results are summarized in Table 2. We observe that:

- MTL-LSTM exploiting stance features shows 2.86% and 2.67% improvements in recall on RumourEval and PHEME respectively, but it also introduces noise, which makes it achieve from 4.90% to 21.38% degradation than our model in accuracy, precision, and F1. Besides, our model achieves 1.94% and 2.12% boosts than the latest baseline (Bayesian-DL) in accuracy on the two datasets respectively. These reveal the effectiveness of our model.
- SVM incorporating meta-data features gains better performance than some deep neural networks only learning content features,

like CNN and TE. It confirms the effectiveness of meta-data features for fake news detection.
- Both TRNN and Bayesian-DL by integrating comments into posts achieves eminent performance than other baselines, which reveals the effectiveness of the fusion of comments and posts. But they are worse than AIFN because AIFN not only captures semantics of comments and posts but also emotional features from the both.

4.4 Discussions

We evaluate the effectiveness of each part of AIFN and the effectiveness of internal structure of GAIN and SFSN.

4.4.1 Ablation Analysis of AIFN

Table 3 provides the experimental results of AIFN and the following simplified models:

- **BERT+BiLSTM** applies BERT to BiLSTM model.
- **BERT+BiLSTM+SFSN** adds semantic-level fusion self-attention networks (SFSN) to BERT+BiLSTM.
- **BERT+BiLSTM+SFSN+GAIN** means BERT+BiLSTM+SFSN incorporating gated adaptive interaction networks (GAIN), i.e., AIFN.
- **Glove+BiLSTM+SFSN+GAIN** uses Glove [19] as word embeddings of AIFN.

From the experimental results of Table 3, we draw the following observations:

- **Effectiveness of BERT.** Compared with BERT+BiLSTM+SFSN +GAIN and Glove+BiLSTM+SFSN+GAIN, the former achieves 2.85% and 1.65% boosts in accuracy on RumourEval and PHEME respectively. It proves the effectiveness of BERT including feature ensemble.
- **Effectiveness of SFSN.** BERT+BiLSTM+SFSN boosts the performance as compared with BERT+BiLSTM, showing 4.70% and 5.66% improvements in accuracy on the two datasets, which reveals that SFSN enhancing semantic associations and fusion is effective.
- **Effectiveness of GAIN.** Compared to BERT+BiLSTM+SFSN, BERT+BiLSTM+SFSN+GAIN notably improves the performance (2.65% and 1.49% boosts) on the two datasets with the help of conflicting semantics obtained by GAIN. It explains the effectiveness of GAIN.

4.4.2 Semantic-level Fusion Self-attention Networks Evaluation

To further evaluate the effectiveness of SFSN, we verify the performance of semantic-level fusion between four output features of GAIN and four self-attention networks i.e., four SFSN, where 'SFSN-post-word' represents only removing the fusion between the output of GAIN for word-level of posts and self-attention networks for word-level of posts. 'SFSN-post-emotion', 'SFSN-comment-word', and 'SFSN-comment-emotion' are for only removing the fusion on emotion level of posts, only removing the fusion on word level of comments, and only removing the fusion on emotion level of comments, respectively. Table 4 presents the experimental results of these methods on RumourEval and PHEME. We have the following observations:

Table 2.   Performance comparison of our proposed model against the baselines.

Dataset	Measure	SVM	CNN	TE	DeClarE	MTL-LSTM	TRNN	Bayesian-DL	Ours
RumourEval	A (%)	71.42	61.90	66.67	66.67	66.67	76.19	80.95	**82.89**
	P (%)	66.67	54.54	60.00	58.33	57.14	70.00	77.78	**78.52**
	R (%)	66.67	66.67	66.67	77.78	**88.89**	77.78	77.78	86.21
	F1 (%)	66.67	59.88	63.15	66.67	69.57	73.68	77.78	**82.19**
PHEME	A (%)	72.18	59.23	65.22	67.87	74.94	78.65	80.33	**82.45**
	P (%)	78.80	56.14	63.05	64.68	68.77	77.11	78.29	**79.12**
	R (%)	75.75	64.64	64.64	71.21	**87.87**	78.28	79.29	85.20
	F1 (%)	72.10	60.09	63.83	67.89	77.15	77.69	78.78	**82.05**

Table 3.   Comparison between AIFN and simplified models on RumourEval and PHEME.

	RumourEval				PHEME			
	A(%)	P(%)	R(%)	F1(%)	A(%)	P(%)	R(%)	F1(%)
BERT+BiLSTM	75.54	70.22	75.26	72.65	75.30	73.89	76.21	75.03
BERT+BiLSTM+SFSN	80.24	77.18	84.23	80.79	80.96	77.56	83.66	80.49
BERT+BiLSTM+SFSN+GAIN	82.89	78.52	86.21	82.19	82.45	79.12	85.20	82.05
Glove+BiLSTM+SFSN+GAIN	80.04	77.21	84.90	80.87	80.80	77.13	83.75	80.30

- The removal of word-level SFSNs presents lower performance than emotion-level SFSNs, which shows up to 1.27% and 1.04% performance degradation than emotion-level SFSNs in accuracy on RumourEval and PHEME, respectively. It reveals that semantics are more effective than emotions for fake news detection.
- Compared with comment-level SFSNs, post-level SFSNs decrease 0.52% and 0.64% performance, which indicates posts contain more effective credibility indicators.
- The removal of any SFSN will degrade the performance of the model, which shows from 2.84% to 1.91% degradation in accuracy on RumourEval and PHEME, respectively. This demonstrates the effectiveness of SFSN.

Table 4.   Comparison between different SFSNs.

		A(%)	P(%)	R(%)	F1(%)
RumourEval	SFSN-post-word	80.05	76.87	84.03	80.29
	SFSN-post-emotion	81.20	77.82	85.34	81.40
	SFSN-comment-word	80.57	77.24	84.69	80.79
	SFSN-comment-emotion	81.84	78.13	85.94	81.85
	Our Model	82.89	78.52	86.21	82.19
PHEME	SFSN-post-word	80.54	77.37	83.21	80.18
	SFSN-post-emotion	81.51	78.26	84.34	81.19
	SFSN-comment-word	80.95	77.91	83.78	80.74
	SFSN-comment-emotion	81.96	78.68	84.76	81.61
	Our Model	82.45	79.12	85.20	82.05

To validate the capture of valuable features in SFSN, we respectively map the outputs of self-attention networks and simple fusion strategies (which replaces SFSN) to the input elements of word-level and the visualized results are shown in Figure 3 where the input is the tweet in Figure 1(b). We observe that:

- Concatenation and addition can capture partial keywords, like 'firefighters' and 'died' as well as obtaining some non-keywords, like 'old' and 'another', which illustrates that simple fusion strategies easily introduce noise.
- SFSN can catch most of keywords compared with golds while it does not touch any irrelevant semantics, which confirms that SFSN is capable of effectively capturing noteworthy features.

Figure 3.   The words captured by SFSN, concatenation, and addition at word-level in posts.

4.4.3   Gated Adaptive Interaction Networks Evaluation

Table 5 provides the performance of each part of GAIN by the following simplified models: **-All** means that GAIN is replaced by concatenation; **-Conflicting** implies that only refining gate works in GAIN; **-Refining** means that the refining gate is removed; and **-Adaptive** means that adaptive mechanism is replaced with concatenation.

From Table 5, we observe that:

- Ablating any parts of GAIN can reduce the performance of model, underperforming AIFN from 0.44% to 2.65% degradation in accuracy on RumourEval and PHEME, respectively. This confirms the effectiveness of each part of GAIN;
- -Conflicting achieves the worst performance compared to the models that only ablate one part, showing at least 1.29% degradation in accuracy on the two datasets, which explains that conflicting semantics obtained by -Conflicting can more strongly improve the performance of fake news detection.

4.4.4   Error Analysis

According to the results of Table 2, our model achieves unsatisfied performance in recall compared with MTL-LSTM by -2.86% and -2.67% on RumourEval and PHEME, respectively. Two reasons can be explained for this issue: 1) MTL-LSTM builds multi-task learning, which can learn the relationship features between tasks. Our model focuses on the interaction and interaction of multiple effective features under a single task fake news detection, lacking joint

Table 5. Comparison among different parts of GAIN.

		A(%)	P(%)	R(%)	F1(%)
RumourEval	-All	80.24	77.18	84.23	80.79
	-Conflicting	81.04	77.26	85.48	81.16
	-Refining	81.32	77.74	84.56	81.01
	-Adaptive	82.45	78.03	85.89	81.77
	AIFN	82.89	78.52	86.21	82.19
PHEME	-All	80.96	77.56	83.66	80.49
	-Conflicting	81.16	77.91	84.21	80.94
	-Refining	81.52	78.53	84.78	81.54
	-Adaptive	81.89	78.87	84.95	81.80
	AIFN	82.45	79.12	85.20	82.05

training of multiple tasks. 2) Stance features may have significant advantages in recall rate, but have no obvious performance boosts and even bring negative effects on other evaluation metrics. In order to improve the performance of our model effectively and equitably, we do not leverage stance features to interact with other credibility indicators.
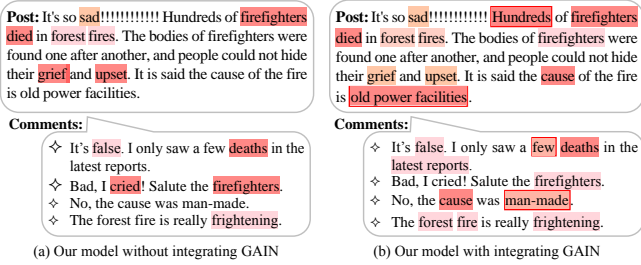

(a) Our model without integrating GAIN   (b) Our model with integrating GAIN

Figure 4. The visualization of fusion interaction of AIFN.

4.5 Case Study

To express intuitively what features AIFN has learned from posts and comments, we design two groups of experiments to visualize the outputs of semantic-level fusion self-attention networks (SFSN). The first group is SFSN integrating GAIN (our model with integrating GAIN) and the second is that SFSN integrates the outputs of concatenation which replaces GAIN (our model without integrating GAIN). Specifically, we firstly employ max-pooling operation to pool the outputs of four self-attention networks and then map them into the corresponding elements in the input layer respectively, and finally obtain interesting patterns to visualize in Figure 4. We observe that:

- From Figure 4 (a), our model without integrating GAIN not only obtains some keywords, like 'firefighters' and 'died' but also highlights semantic associations between posts and comments, such as 'died' to 'deaths', 'grief, upset' to 'cried', and 'firefighters', which gets benefit from SFSN screening and feature correlation.
- From Figure 4 (b), the model with GAIN captures similar semantics between posts and comments, like 'died' and 'cause', and also focuses on some conflicting words, like 'hundred', 'old power facilities' in posts and 'few', 'man-made' in comments. It reveals GAIN effectively finds conflicting semantics about fake news from comments.

5 Conclusion

In this paper, we studied the problem of cross-interaction fusion in fake news detection on social media and proposed adaptive interaction fusion networks considering the extracted features from posts and comments. Specifically, we discovered conflicting semantics between posts and comments by gated adaptive interaction networks and developed semantic-level fusion self-attention networks to explore feature associations for capturing noteworthy features and fusing deeply them. The experimental results based on two real-world datasets demonstrated that our method significantly outperformed previous state-of-the-art models. In the future, we will capture semantic conflicts by considering the hierarchical interaction structure of comments for fake news detection.

ACKNOWLEDGEMENTS

REFERENCES

[1] Ameeta Agrawal, Aijun An, and Manos Papagelis, 'Learning emotion-enriched word representations', in *COLING*, pp. 950–961, (2018).

[2] Oluwaseun Ajao, Deepayan Bhowmik, and Shahrzad Zargari, 'Sentiment aware fake news detection on online social networks', in *ICASSP*, pp. 2507–2511, (2019).

[3] Hunt Allcott and Matthew Gentzkow, 'Social media and fake news in the 2016 election', *Journal of economic perspectives*, **31**(2), 211–36, (2017).

[4] Alexandre Bovet and Hernán A Makse, 'Influence of fake news in twitter during the 2016 us presidential election', *Nature communications*, **10**(1), 7, (2019).

[5] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete, 'Information credibility on twitter', in *The Web Conference*, pp. 675–684, (2011).

[6] Yi-Chin Chen, Zhao-Yang Liu, and Hung-Yu Kao, 'Ikm at semeval-2017 task 8: Convolutional neural networks for stance detection and rumor verification', in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 465–469, (2017).

[7] Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga, 'Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours', in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 69–76, (2017).

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, 'Bert: Pre-training of deep bidirectional transformers for language understanding', in *NAACL*, pp. 4171–4186, (2019).

[9] Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer, 'Fake news on twitter during the 2016 us presidential election', *Science*, **363**(6425), 374–378, (2019).

[10] Gisel Bastidas Guacho, Sara Abdali, Neil Shah, and Evangelos E Papalexakis, 'Semi-supervised content-based detection of misinformation via tensor embeddings', in *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 322–325. IEEE, (2018).

[11] Dhruv Khattar, Jaipal Singh Goud, Manish Gupta, and Vasudeva Varma, 'Mvae: Multimodal variational autoencoder for fake news detection', in *The Web Conference*, pp. 2915–2921, (2019).

[12] Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga, 'All-in-one: Multi-task learning for rumour verification', in *COLING*, pp. 3402–3413, (2018).

[13] David MJ Lazer, Matthew A Baum, Yochai Benkler, Adam J Berinsky, Kelly M Greenhill, Filippo Menczer, Miriam J Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, et al., 'The science of fake news', *Science*, **359**(6380), 1094–1096, (2018).

[14] Ming Liang, Bin Yang, Yun Chen, Rui Hu, and Raquel Urtasun, 'Multitask multi-sensor fusion for 3d object detection', in *CVPR*, pp. 7345–7353, (2019).

[15] Yang Liu and Yi-Fang Brook Wu, 'Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks', in *AAAI*, (2018).

[16] Jing Ma, Wei Gao, and Kam-Fai Wong, 'Rumor detection on twitter with tree-structured recursive neural networks', in *ACL*, pp. 1980–1989, (2018).

[17] Jing Ma, Wei Gao, and Kam-Fai Wong, 'Detect rumors on twitter by promoting information campaigns with generative adversarial learning', in *The Web Conference*, pp. 3049–3055, (2019).

[18] Duc Minh Nguyen, Tien Huu Do, Robert Calderbank, and Nikos Deligiannis, 'Fake news detection using deep markov random fields', in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 1391–1400, (2019).

[19] Jeffrey Pennington, Richard Socher, and Christopher Manning, 'Glove: Global vectors for word representation', in *EMNLP*, pp. 1532–1543, (2014).

[20] Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum, 'Declare: Debunking fake news and false claims using evidence-aware deep learning', in *EMNLP*, pp. 22–32, (2018).

[21] Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein, 'A stylometric inquiry into hyperpartisan and fake news', in *ACL*, pp. 231–240, (2018).

[22] Feng Qian, Chengyue Gong, Karishma Sharma, and Yan Liu, 'Neural user response generator: Fake news detection with collective user intelligence', in *IJCAI*, pp. 3834–3840, (2018).

[23] Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu, 'defend: Explainable fake news detection', in *SIGKDD*, (2019).

[24] Kai Shu, Xinyi Zhou, Suhang Wang, Reza Zafarani, and Huan Liu, 'The role of user profile for fake news detection', *arXiv preprint arXiv:1904.13355*, (2019).

[25] Sebastian Tschiatschek, Adish Singla, Manuel Gomez Rodriguez, Arpit Merchant, and Andreas Krause, 'Fake news detection in social networks via crowd signals', in *The Web Conference*, pp. 517–524, (2018).

[26] William Yang Wang, '"liar, liar pants on fire": A new benchmark dataset for fake news detection', in *ACL*, pp. 422–426, (2017).

[27] Lianwei Wu, Yuan Rao, Haolin Jin, Ambreen Nazir, and Ling Sun, 'Different absorption from the same sharing: Sifted multi-task learning for fake news detection', in *EMNLP-IJCNLP*, pp. 4636–4645, (2019).

[28] Lianwei Wu, Yuan Rao, Ambreen Nazir, and Haolin Jin, 'Discovering differential features: Adversarial learning for information credibility evaluation', *Information Sciences*, **516**, 453–473, (2020).

[29] Lianwei Wu, Yuan Rao, Hualei Yu, Yiming Wang, and Nazir Ambreen, 'A multi-semantics classification method based on deep learning for incredible messages on social media', *Chinese Journal of Electronics*, **28**(4), (2019).

[30] Lianwei Wu, Yuan Rao, Hualei Yu, Yiming Wang, and Ambreen Nazir, 'False information detection on social media via a hybrid deep model', in *International Conference on Social Informatics*, pp. 323–333, (2018).

[31] Fan Yang, Shiva K Pentyala, Sina Mohseni, Mengnan Du, Hao Yuan, Rhema Linder, Eric D Ragan, Shuiwang Ji, and Xia Ben Hu, 'Xfake: Explainable fake news detector with visualizations', in *The Web Conference*, pp. 3600–3604, (2019).

[32] Kai Zhang, Hefu Zhang, Qi Liu, Hongke Zhao, Hengshu Zhu, and Enhong Chen, 'Interactive attention transfer network for cross-domain sentiment classification', in *AAAI*, (2019).

[33] Qiang Zhang, Aldo Lipani, Shangsong Liang, and Emine Yilmaz, 'Reply-aided detection of misinformation via bayesian deep learning', in *The Web Conference*, pp. 2333–2343, (2019).

[34] Zhe Zhao, Paul Resnick, and Qiaozhu Mei, 'Enquiring minds: Early detection of rumors in social media from enquiry posts', in *The Web Conference*, pp. 1395–1405, (2015).

[35] Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie, 'Analysing how people orient to and spread rumours in social media by looking at conversational threads', *PloS one*, **11**(3), e0150989, (2016).