KAN: Knowledge-aware Attention Network for Fake News Detection

Yaqian Dun[1, 2], **Kefei Tu**[1, 2], **Chen Chen**[1, 2*], **Chunyan Hou**[3], **Xiaojie Yuan**[1, 2]
[1] College of Computer Science, Nankai University, Tianjin, China
[2] Tianjin Key Laboratory of Network and Data Security Technology, Tianjin, China
[3] School of Computer Science and Engineering, Tianjin University of Technology, Tianjin, China
{dunyaqian, tukefei}@dbis.nankai.edu.cn, {nkchenchen, yuanxj}@nankai.edu.cn, chunyanhou@163.com

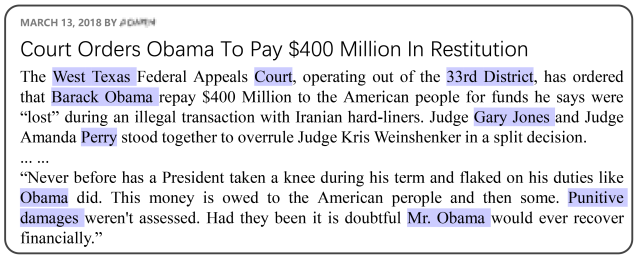
Abstract

The explosive growth of fake news on social media has drawn great concern both from industrial and academic communities. There has been an increasing demand for fake news detection due to its detrimental effects. Generally, news content is condensed and full of knowledge entities. However, existing methods usually focus on the textual contents and social context, and ignore the knowledge-level relationships among news entities. To address this limitation, in this paper, we propose a novel Knowledge-aware Attention Network (KAN) that incorporates external knowledge from knowledge graph for fake news detection. Firstly, we identify entity mentions in news contents and align them with the entities in knowledge graph. Then, the entities and their contexts are used as external knowledge to provide complementary information. Finally, we design News towards Entities (N-E) attention and News towards Entities and Entity Contexts (N-E^{2}C) attention to measure the importances of knowledge. Thus, our proposed model can incorporate both semantic-level and knowledge-level representations of news to detect fake news. Experimental results on three public datasets show that our model outperforms the state-of-the-art methods, and also validate the effectiveness of knowledge attention.

Introduction

Social media has become a platform for people to obtain and exchange information. Due to the increasing usage and convenience of social media, more and more people publish and read news online. Meanwhile, it also gradually becomes an ideal place for the widespread of fake news. Since fake news distorts and fabricates facts maliciously, its extensive dissemination has extremely negative impacts on individuals and society. Therefore, it is desirable and socially beneficial to detect fake news in social media.

For fake news detection, early studies mainly focus on machine learning model based on feature engineering (Castillo, Mendoza, and Poblete 2011; Feng, Banerjee, and Choi 2012). After the emergence of deep learning, various deep-learning-based approaches are proposed and greatly improve the detection performances. *Ma et al.* (Ma et al. 2016) use recurrent neural networks to extract linguistic features within the news contents. *Wang et al.* (Wang et al. 2018b) propose a deep neural network to capture multi-modal features for fake news detection. *Liu et al.* (Liu and Wu 2018) utilizes CNN and GRU to capture useful patterns from user profiles.


MARCH 13, 2018 BY [redacted]

Court Orders Obama To Pay $400 Million In Restitution

The West Texas Federal Appeals Court, operating out of the 33rd District, has ordered that Barack Obama repay $400 Million to the American people for funds he says were "lost" during an illegal transaction with Iranian hard-liners. Judge Gary Jones and Judge Amanda Perry stood together to overrule Judge Kris Weinshenker in a split decision.
... ...
"Never before has a President taken a knee during his term and flaked on his duties like Obama did. This money is owed to the American people and then some. Punitive damages weren't assessed. Had they been it is doubtful Mr. Obama would ever recover financially."

Figure 1: A piece of fake news on PolitiFact.

Although the existing deep learning methods have achieved great success to detect fake news based on the high-level feature representations of news contents, they ignore the external knowledge by which people usually judge the authenticity of the news. News content is highly condensed and comprised of a large number of entity mentions. A named entity could possibly denote different entity mentions because a named entity may have multiple textual forms, such as its aliases, abbreviations and alternate spellings. For example, as shown in Figure 1, a piece of news contains the following entity mentions: "West Texas", "33rd District", "Barack Obama", "Gary Jones", "Mr. Obama", "Punitive damages". People usually read the news content first and then realize "West Texas" is a location, "Barack Obama" and "Gary Jones" are politicians, and can judge that "Barack Obama", "Mr. Obama" and "Obama" refer to the same person. "Barack Obama", "Mr. Obama" and "Obama" are entity mentions of entity "Barack Obama". These knowledge-level judgments and connections are helpful to evaluate the credibility of the news. However, these mentions cannot be understood directly based on the text content of news. Thus, the introduction of external knowledge is much important for fake news detection.

To extract deep logical connections among entities, it is necessary to incorporate the knowledge information in knowledge graph. Knowledge graph is a multi-relational graph which is composed of entities as nodes and relation-

*Corresponding author.

ships as edges with different types. An edge can describe the directed relationship between the two entities. This knowledge is beneficial to understand news because: (1) the ambiguous entity mentions usually occur in news contents. The ambiguity of mentions can be avoided by associating each mention in news content with its corresponding entity in knowledge graph. (2) knowledge graph also can provide more complementary information about relevant entities, which is helpful for learning knowledge-level relationships among entities in news and improving the performance of fake news detection.

In this paper, we propose a novel Knowledge-aware Attention Network (KAN) that incorporates external knowledge from knowledge graph for fake news detection. First, we identify entity mentions in the news contents, and then obtain corresponding entities through external knowledge graph such as YAGO (Suchanek, Kasneci, and Weikum 2007), Freebase (Bollacker et al. 2008), Wikidata (Vrandecic and Krötzsch 2014) and Probase (Wu et al. 2012). Next, we extract the entity context of each entity (i.e., its directly connected neighbors in knowledge graph) as auxiliary information. Finally, these entities and their entity contexts serve as external knowledge so as to learn both semantic-level and knowledge-level representations of news.

To fuse external knowledge into the model effectively, it is key to figure out the relative importance of each entity associated with news content. Thus, we use News towards Entities (N-E) attention to calculate the semantic similarity between news contents and its corresponding entities, where each entity is assigned a weight to represent its importance. For the purpose of integrating entity contexts, we design News towards Entities and Entity contexts (N-E^{2}C) attention to assign a weight to the entity context by the vitality of the corresponding entity. Finally, the representations of news, entities and their contexts are concatenated and fed into a fully-connected network to predict the veracity of the news. The main contributions of this paper are summarized as follows:

- We propose to incorporate entities and their entity contexts which are distilled from knowledge graph for fake news detection.
- We propose a Knowledge-aware Attention Network for fake news detection. To integrate knowledge into news more reasonable and effective, we introduce two attention mechanisms (i.e., N-E and N-E^{2}C) to obtain the relative importance of entities and entity contexts.
- We conduct extensive experiments on three standard datasets for fake news detection. The results show that our model outperforms the state-of-the-art methods.

Related Works

In this section, we take a brief revisit on the following related topics: fake news detection, knowledge graph and attention mechanism.

Fake News Detection has attracted great concern in recent years. Following the previous work (Shu et al. 2017; Ruchansky, Seo, and Liu 2017), we specify the definition of fake news as news which is intentionally fabricated and can be verified as fake. Fake news detection is a challenging task because it is associated with a variety of characteristics of news such as its authenticity, author intention and literal form. Early works on detecting fake news mainly focus on designing a complementary set of hand-crafted features based on linguistic features (Castillo, Mendoza, and Poblete 2011; Feng, Banerjee, and Choi 2012). These studies require a great deal of effort to explore the effectiveness of manual features. To expand beyond the specificity of hand-crafted features, *Wang et al.* (Wang et al. 2018b) propose a deep neural network to capture multi-modal data features for fake news detection. *Liu et al.* (Liu and Wu 2018) utilize CNN and GRU to capture the useful patterns from user profiles. The major difference between prior works and ours is that we use knowledge graph to capture latent knowledge-level connections among news entities for better exploration in fake news detection.

Knowledge Graph has millions of entries that describe real world entities like people, places and organizations. Within a knowledge graph, entities are represented as nodes and relationships between these nodes are described as edges. The knowledge graph has been used in many applications, such as movie recommendation (Zhang et al. 2016), machine reading (Yang and Mitchell 2017), text classification (Wang et al. 2017) and question answering (Dong et al. 2015). Especially, knowledge graph is widely used by means of entity linking (Shen, Wang, and Han 2015). Entity linking with knowledge graphs aim to automatically map mentions in text to the corresponding entities in a knowledge graph. Entity linking can be split into two classes of approaches: End-to-End and Disambiguation-Only. End-to-End methods process a piece of text to extract the entity mentions and then disambiguate these extracted mentions to the correct entry in knowledge graph. In contrast, disambiguation-Only methods directly take standard named entities as input and only disambiguate them to the correct entry in a given knowledge graph. Recently, there are many deep learning methods proposed for this task. *Landau et al.* (Francis-Landau, Durrett, and Klein 2016) use CNN to capture semantic correspondence between a mentions context and a proposed target entity for entity linking. *Moro et al.* (Moro, Raganato, and Navigli 2014) propose a unified graph-based approach Babelfy to entity linking and word sense disambiguation. *Chen et al.* (Chen et al. 2018) use the concepts of entities as topics to entity linking.

Attention Mechanism has been successfully used in many NLP tasks. *Bahdanau et al.* (Bahdanau, Cho, and Bengio 2015) first propose attention mechanism which can be used in machine translation. *Yang et al.* (Yang et al. 2016) used hierarchical attention network (HAN) for document classification. *Hao et al.* (Hao et al. 2017) employ a vanilla attention mechanism to measure the similarity between the question and the answer in question answering task. *Fung et al.* (Fung and Mak 2018) apply a multi-head attention mechanism in an end-to-end recurrent network on machine translation tasks. *Du et al.* (Du and Qian 2018) propose a hierarchical gated convolutional network with multi-head attention for text classification. Inspired by these works, we em-

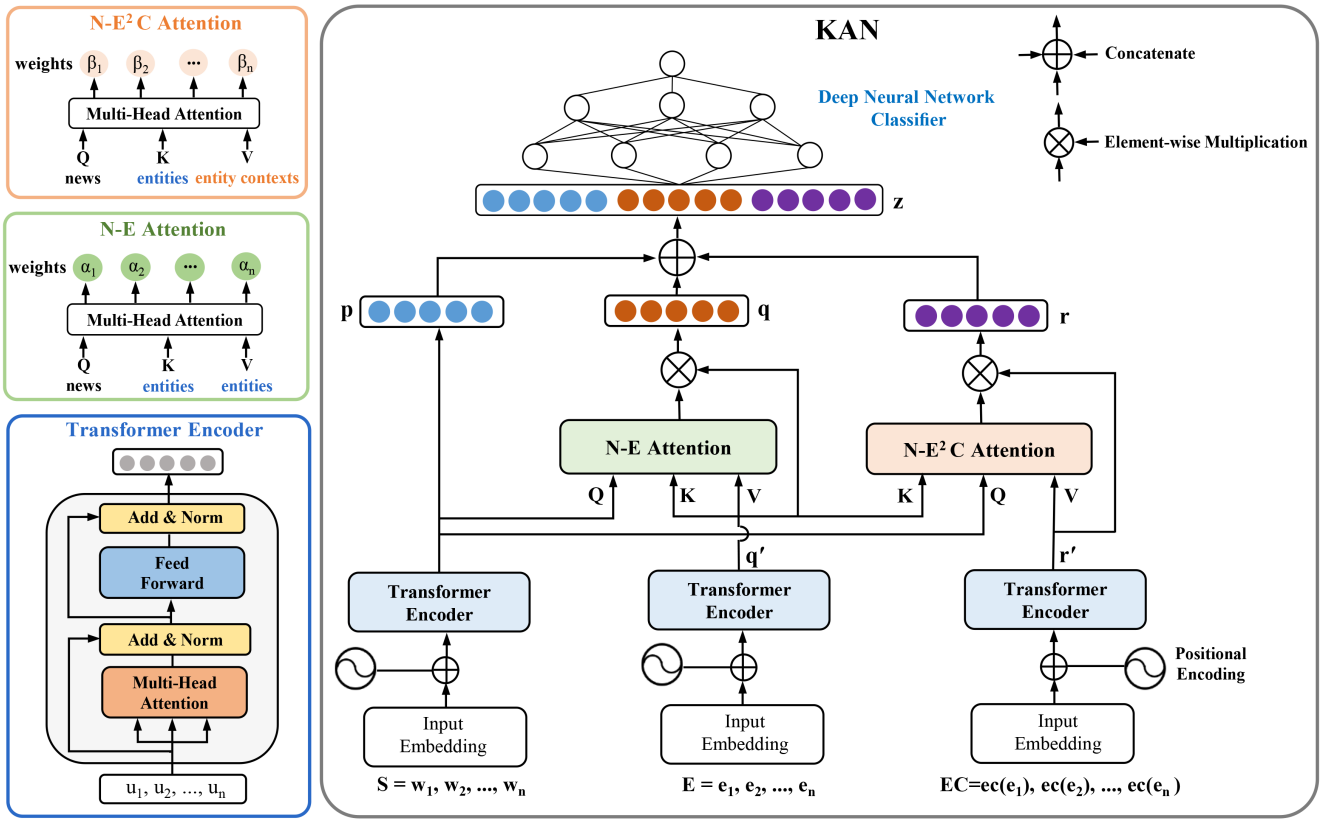


Figure 2: Illustration of the KAN framework.

ploy two attention mechanisms to measure the importance of two kinds of knowledge.

Problem Statement

We formulate the fake news detection problem in this paper as follows. Fake news detection task can be defined as a binary classification problem, i.e., each news article can be real ($y = 0$) or fake ($y = 1$). Each piece of news S is composed of a sequence of words, i.e., $\boldsymbol{S} = \{w_1, w_2, \cdots, w_n\}$, where one or several words may be associated with an entity e_i in the knowledge graph. In addition, each entity e_i have many immediate neighbors in the knowledge graph. The neighbor entities of entity e_i is defined as *"entity context"* $ec(e_i)$ of the entity e_i. Formally, given a news article $\boldsymbol{S} = \{w_i\}$ as well as the relevant entities $\boldsymbol{E} = \{e_i\}$ and entity contexts $\boldsymbol{EC} = \{ec(e_i)\}$, we aim to learn such a fake news detection function F: $F(\boldsymbol{S}, \boldsymbol{E}, \boldsymbol{EC}) \rightarrow y$, where $y \in \{0, 1\}$ is the ground-truth label of news. Practically, we use the average value of the vector representations of the neighbor entities to represent $ec(e_i)$.

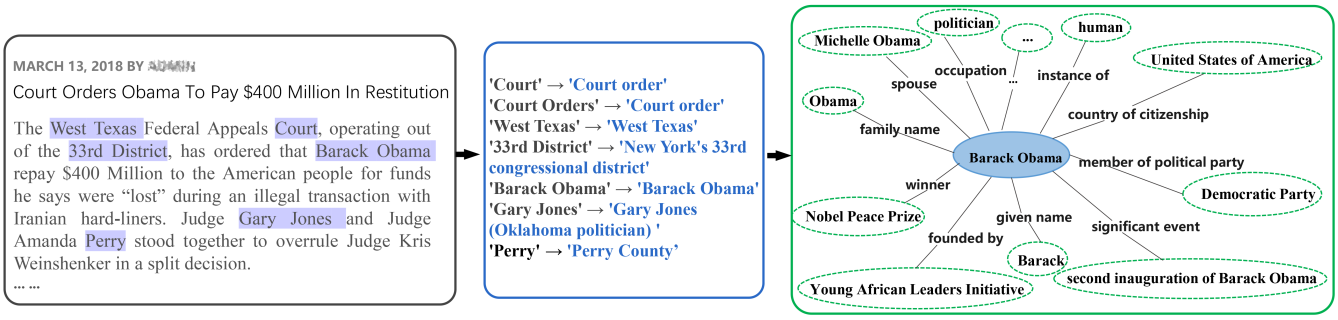
Our Model

KAN Framework

We give a brief overview of the proposed Knowledge-aware Attention Network (KAN). The framework of KAN is depicted in Figure 2. We introduce the architecture of KAN from the bottom up. The input to KAN consists of news contents, entities, and entity contexts. The output of KAN is the probability distribution of labels over classes. For each piece of news, a Transformer Encoder (Vaswani et al. 2017) is used to encode news contents and generate the representation of news. The details are presented in the *Text Encoding* subsection. We will describe the extraction of entities and entity contexts from knowledge graph in *Knowledge Extraction* subsection. Then, these two kinds of extracted external knowledge are encoded by transformer encoders respectively to produce the knowledge encoding, the details are described in *Knowledge Encoder* subsection. To fuse knowledge encoding into the model effectively, we design two attention mechanisms (i.e., N-E and N-E^{2}C) to measure the relative importance of entities and entity contexts, and then aggregate their vector representations with different weights. The details of attention module are presented in *Knowledge-aware Attention* subsection. Finally, the representation of news, entities, and their contexts are concatenated and fed into a fully-connected network to predict the veracity of the news.

Text Encoding

The Text Encoding module aims to produce the news content representation p. To capture the representation of news contents, we employ Transformer Encoder (Vaswani et al. 2017)



(a) A Piece of News (b) Entity Linking (c) Entity Context Extraction

Figure 3: The process of knowledge extraction.

as the core of the module. The reason why we chose the Transformer Encoder is explained as follows. Transformer contains self-attention layers which can learn long-term dependency. Meanwhile, it is able to capture the sequence information through positional encoding and has a strong ability to extract semantic features.

The transformer encoder generates the text encoding from the original word sequence and positional encoding. Given a piece of news $\boldsymbol{S} = \{w_1, w_2, \cdots, w_n\}$ of length n, each word w_i is projected into a continuous word embedding w_i' from an word embedding matrix $M \in \mathbb{R}^{V \times d}$, where V is the vocabulary size and d is the embedding dimension. Then, we obtain the news vectors $\boldsymbol{S'} = \{w_1', w_2', \cdots, w_n'\}$, $S' \in \mathbb{R}^{n \times d}$. Moreover, in order to make use of the word order in the news, positional encodings (Vaswani et al. 2017) are used and combined with the word embeddings:

$$u_t = w_t' + pos_t \tag{1}$$

where pos_t is the position encoding for t-th word in the news, we denote $u = u_0, \cdots, u_n \in \mathbb{R}^{n \times d}$ as input encodings to the bottoms of transformer encoder. In general, the architecture of encoder is stacked with identical layers. Each layer is constructed by multi-head self-attention mechanism, residual connection, layer normalization and fully connected feed-forward network. In this paper, we employ one-layer Transformer Encoder to process the input encodings u :

$$\tilde{a} = MultiHeadAttention(u), \tag{2}$$

$$a = LayerNorm(\tilde{a} + u), \tag{3}$$

$$\tilde{u} = FeedForwardNetwork(a), \tag{4}$$

$$p = LayerNorm(\tilde{u} + a), \tag{5}$$

Inside the layer, the input encodings u are first transformed by the sub-layer of multi-head self-attention mechanism. In the next step, the output representations are sent into the sub-layer of point-wise feed-forward neural network. A residual connection (He et al. 2016) is applied around each of the two sub-layers, followed by layer normalization (Ba, Kiros, and Hinton 2016). Then we construct the new representation $\boldsymbol{p}$ with the output of transformer encoder.

Knowledge Extraction

The goal of this module is to retrieve relevant entities from the knowledge graph. The process of entity extraction is shown in Figure 3, which includes the following steps: (1) Through entity linking (Milne and Witten 2008; Sil and Yates 2013), the entity mentions in the news contents are identified and aligned with their counterpart entities in the knowledge graph. As shown in Figure 3(b), the mention "West Texas" is linked to entity "West Texas", and the mention "33rd District" is linked to entity "New York's 33rd congressional district". After that, we can acquire entities sequence $\boldsymbol{E} = \{e_1, e_2, \cdots e_n\}$. (2) The entity contexts are chosen out according to the linked entities in the former step. The "entity context" of entity e_i is defined as the immediate neighbors in the knowledge graph. We extract neighbors entities with one hop distance related to the current entity:

$$\boldsymbol{ec}(e_i) = \{e | (e, rel, e_i) \in G \; or \; (e_i, rel, e) \in G\}, \tag{6}$$

where rel is a relation between two entities and G is the knowledge graph. Figure 3(c) illustrates an example of entity context extraction process. From the picture we can see that entity "Barack Obama" contains many neighbors, such as "politician", "United States of America", "Democratic Party". These neighbors compose the "entity context" of "Barack Obama". After the entity context is distilled from a knowledge graph, each entity is associated with an entity context set, then we can obtain entity contexts sequence $\boldsymbol{EC} = \{ec(e_1), ec(e_2), \cdots, ec(e_n)\}$.

Knowledge Encoder

The introduction of external knowledge can provide more complementary information and reduce the ambiguity caused by entity mentions in news. Given a piece of news, entities and entity contexts related to this news can help to boost the detection performance. The extracted entities sequence E and entity contexts sequence EC are embedded by word2vec (Mikolov et al. 2013), and then we obtain the entities embedding $E' = \{e_1', e_2', \cdots, e_n'\}$, $E' \in \mathbb{R}^{n \times d}$ and entity contexts embedding $EC' = \{ec_1', ec_2', \cdots, ec_n'\}$, $EC' \in \mathbb{R}^{n \times d}$, where d is the embedding dimension. The entity context embedding ec_i' is calculated as the average of its

context entities:

$$ec'_i = \frac{1}{|ec(e_i)|} \sum_{e_t \in ec(e_i)} e'_t, \tag{7}$$

where e'_t is the entity embedding, $\boldsymbol{ec}(e_i)$ is the neighbors entities set with one hop distance of e_i in knowledge graph. After the acquisition of the embeddings of entities and entity contexts, we encode each of them with a transformer encoder and take the outputs q' and r' as the intermediate encoding of entity and entity contexts.

Knowledge-aware Attention

The external knowledge obtained from knowledge graph provides rich information to help detect the class labels for news. To characterize the relative importance of external knowledge, we design two attention networks based on multi-head attention, which allows the model to consider information from different representation subspaces at different positions. The formula to calculate the attention is as follows:

$$Attn(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V, \tag{8}$$

$$MultiHead(Q, K, V) = Concat(Attn_1, \cdots, Attn_H), \tag{9}$$

where queries, keys and values are packed together into matrices Q, K and V, d_k is the dimension of queries and keys, H is the number of heads.

Since not all entities contribute equally to the meaning of a news, we design *News towards Entities (N-E) attention* to measure the importance of each entity with respect to the news content. In N-E attention, as shown in Figure 2, queries come from the news representation p, keys and values come from the entities intermediate encoding q'. By calculating the semantic similarity between news and its corresponding entities, each entity is assigned a weight α_i to represent its importance:

$$Q = W_Q \boldsymbol{p}, K = W_K \boldsymbol{q'}, V = W_V \boldsymbol{q'}, \tag{10}$$

$$\boldsymbol{\alpha} = softmax(\frac{QK^T}{\sqrt{d_k}}), \boldsymbol{q} = \boldsymbol{\alpha} V, \tag{11}$$

where $\boldsymbol{q}$ denotes the entities representation, W_Q, W_K and W_V are parameter matrices, α refers to the attention distribution.

In addition, in order to take into account the relative importance of entity contexts, we propose *News towards Entities and Entity Contexts (N-E^{2}C) attention* to measure the importance of each entity context according to news and its entities. As depicted in Figure 2, in N-E^{2}C attention, queries come from the news representation p, keys come from the entities intermediate encoding q', values come from the entity context intermediate encoding r'. Through calculating the semantic similarity between news and its corresponding entities, the weight $\boldsymbol{\beta_i}$ is assigned to each entity context according to the vitality of the corresponding entity:

$$Q = W_Q \boldsymbol{p}, K = W_K \boldsymbol{q'}, V = W_V \boldsymbol{r'}, \tag{12}$$

$$\boldsymbol{\beta} = softmax(\frac{QK^T}{\sqrt{d_k}}), \boldsymbol{r} = \boldsymbol{\beta} V, \tag{13}$$

where $\boldsymbol{r}$ denotes the entity contexts representation.

Deep Neural Network Classifier

The final representation of news, i.e., $\boldsymbol{z}$, can be obtained by concatenating $\boldsymbol{p}$, $\boldsymbol{q}$ and $\boldsymbol{r}$. After that, $\boldsymbol{z}$ is fed into a fully connected layer followed by a *softmax* function to predict the distribution $\boldsymbol{P}$ over news labels on the target:

$$\boldsymbol{P} = softmax(\boldsymbol{W_o z} + \boldsymbol{b_o}), \tag{14}$$

It is trained to minimize the cross entropy loss function:

$$J = -\sum_{i \in D} log \boldsymbol{P}_i(c_i) + \frac{\lambda}{2} \|\Theta\|_2^2. \tag{15}$$

where D denotes the overall training corpus, c_i refers to the ground truth label for news i, $P_i(c_i)$ denotes the probability of the ground truth label, Θ denote the parameters of *KAN*, and λ is the coefficient of $L2$ regularizer.

Experiment

Dataset

In order to evaluate the performance of KAN, we conduct experiments on three benchmark datasets. The first two datasets *PolitiFact* and *GossipCop* are the benchmark datasets called FakeNewsNet (Shu et al. 2018, 2017) for fake news detection. The third dataset is *PHEME* (Zubiaga, Liakata, and Procter 2017). The PHEME dataset is constituted by tweets on the Twitter platform and collected from 5 breaking news. The details of these news datasets are reported in Table 1.

Table 1: Statistics of the news datasets. "#" denotes "the number of"

Statistic	PolitiFact	GossipCop	PHEME
# True news	443	4219	1886
# Fake news	372	3393	856
# Total news	815	7612	2742
avg.# words per news	1427	705	410
avg.# entities per news	55	36	20

Experimental Setup

In the process of knowledge extraction, we use entity linking tools *TagMe* (Ferragina and Scaiella 2010) to disambiguate entity mentions in news contents and link them to corresponding entities in the knowledge graph *Wikidata* (Vrandecic and Krötzsch 2014). In the procedure of entity contexts extraction, we retrieve neighbors entities from *Wikidata*. The word embedding and entity embedding is pre-trained on each dataset using *word2vec* (Mikolov et al. 2013) and the dimension of both word embeddings and entity embeddings are set as is 100. We use a one-layer Transformer Encoder as news and knowledge encoder. The Transformer FFN inner representation size is set to dff = 2048 and the number of attention heads h = 4. An *Adam* optimizer (Kingma and Ba 2015) is adopted to train KAN by optimizing the log loss and the dropout rate is set to 0.5.

We hold out 10% of the news in each dataset as validation sets for tuning the hyper parameters, and for the rest of the

Table 2: Comparison with Existing Methods.

Datasets	Metric	SVM	RFC	DTC	GRU-2	B-TransE	KCNN	KAN
PolitiFact	Precision	0.746	0.7470	0.7476	0.7083	0.7739	0.7852	**0.8687**
	Recall	0.6826	0.7361	0.7454	0.7048	0.7658	0.7824	**0.8499**
	F1	0.6466	0.7362	0.7450	0.7041	0.7641	0.7804	**0.8539**
	Accuracy	0.6694	0.7406	0.7486	0.7109	0.7694	0.7827	**0.8586**
	AUC	0.6826	0.8074	0.7454	0.7896	0.834	0.8488	**0.9197**
GossipCop	Precision	0.7493	0.7015	0.6921	0.7176	0.7369	0.7483	**0.7764**
	Recall	0.6254	0.6707	0.6922	0.7079	0.733	0.7422	**0.7696**
	F1	0.5955	0.6691	0.6919	0.7079	0.734	0.7433	**0.7713**
	Accuracy	0.6643	0.6918	0.6959	0.718	0.7394	0.7491	**0.7766**
	AUC	0.6253	0.7389	0.6929	0.7516	0.7995	0.8125	**0.8435**
PHEME	Precision	0.7357	0.6602	0.648	0.7003	0.6834	0.6832	**0.7593**
	Recall	0.6116	0.6090	0.6541	0.6901	0.6061	0.6419	**0.7437**
	F1	0.6120	0.6138	0.6499	0.6917	0.6074	0.6489	**0.7461**
	Accuracy	0.7379	0.7128	0.6909	0.7371	0.72	0.7265	**0.783**
	AUC	0.6115	0.6833	0.6541	0.7552	0.7278	0.745	**0.8373**

news, we conduct 5-fold cross-validation and use Precision, Recall, F1, Accuracy and AUC as evaluation metrics.

Baselines

We compare our KAN model against several strong baselines as follows:

- **SVM** (Yang et al. 2015): A support vector machine classifier is utilized to detect the fake news based on features extracted from the news.
- **RFC** (Kwon et al. 2013): The random forest classifier using identified characteristics of news to detect whether a news is fake or true.
- **DTC** (Castillo, Mendoza, and Poblete 2011): The news information credibility model using decision tree classifier based on various hand-crafted features.
- **GRU-2** (Ma et al. 2016): The GRU-2 model based on GRUs by adding a second GRU layer that captures higher level feature interactions between different time steps.
- **B-TransE** (Pan et al. 2018): The B-TransE model combines positive and negative single models to detect fake news based on news content and knowledge graphs.
- **KCNN** (Wang et al. 2018a): The KCNN is a state-of-the-art model utilize CNN to learn the representation of news. In this paper, the input for KCNN consists of three parts: news embeddings, entities embeddings and entity contexts embeddings.

Result and Analysis

Comparison with Existing Methods. Table 2 shows the results of all the compared models based on the three datasets. From Table 2, we can draw the following observations:

For content-based methods, such as SVM, RFC, DTC and GRU-2, SVM performs the worst among all the methods. DTC and RFC do not achieve good performance on three datasets. This is because they are built with such hand-crafted features or rules that are inferior to latent features learned by deep learning methods. GRU-2 performs better than hand-crafted models in GossipCop and PHEME, which suggests the superiority of feature extraction of deep neural networks. However, GRU-2 achieves slightly lower results on PolitiFact. This is probably because GRU-2 is limited to deal with long sentences in the dataset.

In addition, those methods using both news content and external knowledge achieve consistently better results than the methods which are purely based on news contents, i.e., KAN $>$ KCNN $>$ B-TransE $>$ GRU-2, SVM, RFC and DTC. This indicates that models can successfully incorporate the external knowledge and significantly boost the detection performance.

Moreover, for using both news contents and knowledge methods, KAN achieves better performance than KCNN and B-TransE. As shown in Table 2, KAN consistently outperforms KCNN on three datasets. For example, in contrast to KCNN in terms of F1 score and Accuracy, KAN achieves performance improvement by 7.4% and 7.6% on PolitiFact, 2.8% and 2.8% on GossipCop, and 9.7% and 5.7% on PHEME. We attribute the superiority of KAN to two reasons: 1) KAN uses the knowledge-aware network which can eliminate the ambiguity caused by the entity mentions in the news and learn knowledge-level connections among news entities. 2) KAN employs the attention network which can measure the importances of entity and entity context knowledge and effectively fuse them into news representation.

Comparison among KAN variants. Further, we compare the components of KAN in terms of the following two aspects to demonstrate the efficiency of KAN framework: the usage of knowledge and the usage of attention mechanisms. The variants of KAN are as follows:

- KAN**EC**: KAN\EC is a variant of KAN without entity contexts sequence when the information is fed into the

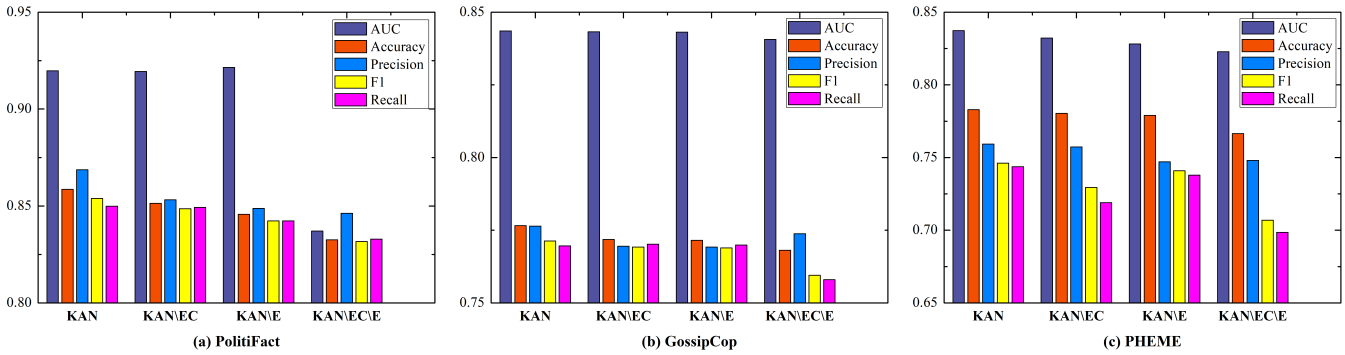


Figure 4: Impact analysis of entities, entity contexts, entities and entity contexts jointly influence for fake news detection.

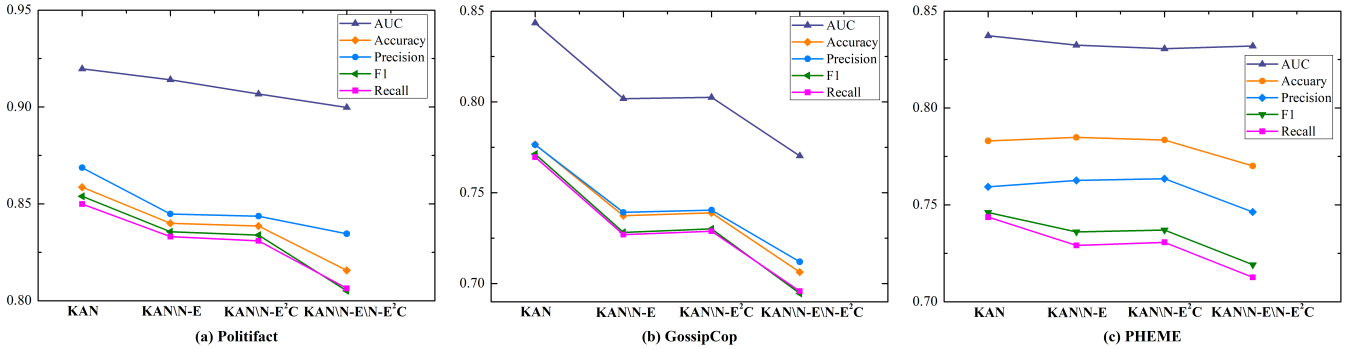


Figure 5: Impact analysis of N-E attention and N-E^{2}C attention for fake news detection.

model.

- KAN**E**: KAN\E is a variant of KAN without entities sequence when the information is fed into the model.
- KAN**EC****E**: KAN\EC\E is a variant of KAN, which eliminates entities and entity contexts and only detect fake news by news contents.
- KAN**N-E**: KAN\N-E is a variant of KAN without counting News contents towards Entities (N-E) attention.
- KAN**N-E^{2}C**: KAN\N-E^{2}C is a variant of KAN without considering News contents towards Entities and Entity Contexts (N-E^{2}C) attention.
- KAN**N-E****N-E^{2}C**: KAN\N-E\N-E^{2}C eliminates both N-E attention and N-E^{2}C attention.

The hyper-parameters in all the variants are determined by the validation set. We evaluate the variants with 5-fold cross-validation. The effects of using knowledge and attention are shown in Figure 4 and Figure 5 respectively, from which we can conclude that:

When we eliminate the entity contexts knowledge, the results are reduced. It suggests that the comprehensive information of entity contexts is helpful for understanding entities in news.

When we disregard the entities sequence, the performance of KAN\E degrades in comparison with KAN on three datasets. The results suggest the entities play an important role in disambiguation of entity mentions in the news. At the same time, it also provides the basis for effectively incorporating entity contexts.

When the external knowledge is removed from KAN, the results of KAN\EC\E degrade in comparison with KAN in all datasets. For example, in contrast to KAN in terms of F1 score, the performance reduces 2.2% on PolitiFact, 1.2% on GossipCop and 1.3% on PHEME. The results suggest the importance to consider knowledge of news to guide fake news detection in KAN.

The usage of N-E attention and N-E^{2}C attention can improve performance respectively, and we can achieve even better performance by using them together. For example, the result of using N-E attention and N-E^{2}C attention together is improved by 2.2% on PolitiFact and 6.2% on GossipCop in terms of Accuracy. The results validate the effectiveness of proposed attention mechanisms.

Conclusion and Future Works

Our work attempts to incorporate entities and entity contexts knowledge from knowledge graph for fake news detection. We propose Knowledge-aware Attention Network that effectively integrates the two kinds of knowledge with news through attention mechanisms. We have demonstrated the effectiveness of our proposed approach by conducting experiments on three real-world datasets. For future work, we will search for better representation form of knowledge to incorporate it into neural networks as explicit features to further boost fake news detection performance.

Acknowledgments

This work is partially supported by NFSC-General Technology Joint Fund for Basic Research (No. U1936105, No. U1936206) and the National Natural Science Foundation of China (No. 61702285). We thank the AC, SPC, PC and reviewers for their insightful comments on this paper.

References

Ba, L. J.; Kiros, J. R.; and Hinton, G. E. 2016. Layer Normalization. *CoRR* .

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *3rd International Conference on Learning Representations*.

Bollacker, K. D.; Evans, C.; Paritosh, P.; Sturge, T.; and Taylor, J. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 1247–1250.

Castillo, C.; Mendoza, M.; and Poblete, B. 2011. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, 675–684. ACM.

Chen, L.; Liang, J.; Xie, C.; and Xiao, Y. 2018. Short Text Entity Linking with Fine-grained Topics. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 457–466.

Dong, L.; Wei, F.; Zhou, M.; and Xu, K. 2015. Question Answering over Freebase with Multi-Column Convolutional Neural Networks. In *In the Association for Computational Linguistics*, 260–269.

Du, H.; and Qian, J. 2018. Hierarchical Gated Convolutional Networks with Multi-Head Attention for Text Classification. In *5th International Conference on Systems and Informatics*, 1170–1175. IEEE.

Feng, S.; Banerjee, R.; and Choi, Y. 2012. Syntactic Stylometry for Deception Detection. In *The 50th Annual Meeting of the Association for Computational Linguistics*, 171–175.

Ferragina, P.; and Scaiella, U. 2010. TAGME: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM Conference on Information and Knowledge Management*, 1625–1628.

Francis-Landau, M.; Durrett, G.; and Klein, D. 2016. Capturing Semantic Similarity for Entity Linking with Convolutional Neural Networks. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, 1256–1261.

Fung, I.; and Mak, B. 2018. Multi-Head Attention for End-to-End Neural Machine Translation. In *11th International Symposium on Chinese Spoken Language Processing*, 250–254.

Hao, Y.; Zhang, Y.; Liu, K.; He, S.; Liu, Z.; Wu, H.; and Zhao, J. 2017. An End-to-End Model for Question Answering over Knowledge Base with Cross-Attention Combining Global Knowledge. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, 221–231.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kwon, S.; Cha, M.; Jung, K.; Chen, W.; and Wang, Y. 2013. Prominent Features of Rumor Propagation in Online Social Media. In *2013 IEEE 13th International Conference on Data Mining*, 1103–1108.

Liu, Y.; and Wu, Y. B. 2018. Early Detection of Fake News on Social Media Through Propagation Path Classification with Recurrent and Convolutional Networks. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 354–361.

Ma, J.; Gao, W.; Mitra, P.; Kwon, S.; Jansen, B. J.; Wong, K.; and Cha, M. 2016. Detecting Rumors from Microblogs with Recurrent Neural Networks. In *IJCAI*, 3818–3824.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient Estimation of Word Representations in Vector Space. In *1st International Conference on Learning Representations*.

Milne, D. N.; and Witten, I. H. 2008. Learning to link with wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, 509–518. ACM.

Moro, A.; Raganato, A.; and Navigli, R. 2014. Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Trans. Assoc. Comput. Linguistics* 231–244.

Pan, J. Z.; Pavlova, S.; Li, C.; Li, N.; Li, Y.; and Liu, J. 2018. Content Based Fake News Detection Using Knowledge Graphs. In *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference, Monterey, CA, USA, October 8-12, 2018, Proceedings, Part I*, volume 11136 of *Lecture Notes in Computer Science*, 669–683.

Ruchansky, N.; Seo, S.; and Liu, Y. 2017. CSI: A Hybrid Deep Model for Fake News Detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 797–806.

Shen, W.; Wang, J.; and Han, J. 2015. Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions. *IEEE Transactions on Knowledge and Data Engineering* 27(2): 443–460.

Shu, K.; Mahudeswaran, D.; Wang, S.; Lee, D.; and Liu, H. 2018. FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media. *CoRR* abs/1809.01286.

Shu, K.; Sliva, A.; Wang, S.; Tang, J.; and Liu, H. 2017. Fake News Detection on Social Media: A Data Mining Perspective. *SIGKDD Explorations* 22–36.

Sil, A.; and Yates, A. 2013. Re-ranking for joint named-entity recognition and linking. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, 2369–2374.

Suchanek, F. M.; Kasneci, G.; and Weikum, G. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, 697–706. ACM.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30*, 5998–6008.

Vrandecic, D.; and Krötzsch, M. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM* 57(10): 78–85.

Wang, H.; Zhang, F.; Xie, X.; and Guo, M. 2018a. DKN: Deep Knowledge-Aware Network for News Recommendation. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, 1835–1844.

Wang, J.; Wang, Z.; Zhang, D.; and Yan, J. 2017. Combining Knowledge with Deep Convolutional Neural Networks for Short Text Classification. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017*, 2915–2921.

Wang, Y.; Ma, F.; Jin, Z.; Yuan, Y.; Xun, G.; Jha, K.; Su, L.; and Gao, J. 2018b. EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. In Guo, Y.; and Farooq, F., eds., *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 849–857.

Wu, W.; Li, H.; Wang, H.; and Zhu, K. Q. 2012. Probase: a probabilistic taxonomy for text understanding. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 481–492. ACM.

Yang, B.; and Mitchell, T. M. 2017. Leveraging Knowledge Bases in LSTMs for Improving Machine Reading. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 1436–1446.

Yang, F.; Yu, X.; Liu, Y.; and Yang, M. 2015. Automatic Detection of Rumor on Sina Weibo. In *Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics (MDS '12)*.

Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A. J.; and Hovy, E. H. 2016. Hierarchical Attention Networks for Document Classification. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1480–1489.

Zhang, F.; Yuan, N. J.; Lian, D.; Xie, X.; and Ma, W. 2016. Collaborative Knowledge Base Embedding for Recommender Systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 353–362.

Zubiaga, A.; Liakata, M.; and Procter, R. 2017. Exploiting Context for Rumour Detection in Social Media. In *Social Informatics - 9th International Conference*, volume 10539 of *Lecture Notes in Computer Science*, 109–123.