dEFEND: Explainable Fake News Detection

Kai Shu
Arizona State University
Tempe, AZ 85201
kai.shu@asu.edu

Limeng Cui
Penn State University
University Park, PA 16802
lzc334@psu.edu

Suhang Wang
Penn State University
University Park, PA 16802
szw494@psu.edu

Dongwon Lee
Penn State University
University Park, PA 16802
dongwon@psu.edu

Huan Liu
Arizona State University
Tempe, AZ 85201
huan.liu@asu.edu

ABSTRACT

In recent years, to mitigate the problem of fake news, computational detection of fake news has been studied, producing some promising early results. While important, however, we argue that a critical missing piece of the study be the explainability of such detection, i.e., *why* a particular piece of news is *detected* as fake. In this paper, therefore, we study the *explainable detection* of fake news. We develop a sentence-comment co-attention sub-network to exploit both news contents and user comments to jointly capture explainable top-k check-worthy sentences and user comments for fake news detection. We conduct extensive experiments on real-world datasets and demonstrate that the proposed method not only significantly outperforms 7 state-of-the-art fake news detection methods by at least 5.33% in F1-score, but also (concurrently) identifies top-k user comments that explain why a news piece is fake, better than baselines by 28.2% in NDCG and 30.7% in Precision.

CCS CONCEPTS

• **Security and privacy** → *Social aspects of security and privacy*.

KEYWORDS

Fake news; explainable machine learning; social network

ACM Reference Format:
Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. dEFEND: Explainable Fake News Detection. In *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19), August 4–8, 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3292500.3330935

1 INTRODUCTION

Social media platforms provide convenient conduit for users to create, access, and share diverse information. Due to the increased usage and convenience of social media, more people seek out and receive timely news information online. For example, the Pew Research Center announced that approximately 68% of US adults get news from social media in 2018, while only 49% reported seeing news on social media in 2012[1]. However, at the same time, social media enables users to get exposed to a myriad of misinformation and disinformation, including **fake news**, i.e., news stories with intentionally false information [1, 40]. For example, a report estimated that over 1 million tweets were related to the fake news story "Pizzagate" by the end of 2016 presidential election[2].

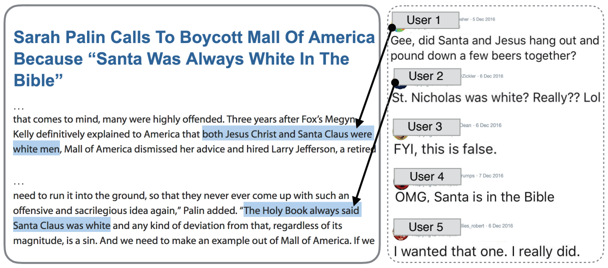
Such widespread of fake news has detrimental societal effects. First, it significantly weakens the public trust in governments and journalism. For example, the reach of fake news during the 2016 U.S. presidential election campaign for top-20 fake news pieces was, ironically, larger than the top-20 most-discussed true stories[3]. Second, fake news may change the way people respond to legitimate news. A study has shown that people's trust in mass media has dramatically degraded across different age groups and political parties[4]. Third, rampant "online" fake news can lead to "offline" societal events. For example, fake news claiming that Barack Obama was injured in an explosion wiped out $130 billion in stock value[5]. Therefore, it has become critically important to be able to curtail the spread of fake news on social media, promoting trust in the entire news ecosystem.

However, detecting fake news on social media presents unique challenges. First, as fake news is *intentionally* written to mislead readers, it is non-trivial to detect fake news simply based on its content. Second, social media data is large-scale, multi-modal, mostly user-generated, sometimes anonymous and noisy. Addressing these challenges, recent research advancements aggregate users' social engagements on news pieces to help infer which articles are fake [13, 37], giving some promising early results. For example, Natali *et al.* [37] propose a hybrid deep learning framework to model news text, user response, and post source simultaneously for fake news detection. Guo *et al.* [13] utilize a hierarchical neural network to detect fake news, modeling user engagements with social attention that selects important user comments.

Despite the success of existing deep learning based fake news detection methods, however, the majority of these methods focus on *detecting* fake news effectively with latent features but cannot explain "why" a piece of news was detected as fake news. Being able to *explain* why news was determined as fake is much desirable because: (1) the derived explanation can provide new insights and knowledge originally hidden to practitioners; and (2) extracting explainable features from noisy auxiliary information can further

[1] https://tinyurl.com/ybcy2foa
[2] https://tinyurl.com/z38z5zh
[3] https://tinyurl.com/y8dckwhr
[4] https://tinyurl.com/y9kegobd
[5] https://tinyurl.com/ybs4tgpg

KDD '19, August 4–8, 2019, Anchorage, AK, USA
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6201-6/19/08…$15.00
https://doi.org/10.1145/3292500.3330935



Sarah Palin Calls To Boycott Mall Of America Because "Santa Was Always White In The Bible"

...
that comes to mind, many were highly offended. Three years after Fox's Megyn Kelly definitively explained to America that both Jesus Christ and Santa Claus were white men, Mall of America dismissed her advice and hired Larry Jefferson, a retired

...
need to run it into the ground, so that they never ever come up with such an offensive and sacrilegious idea again," Palin added. "The Holy Book always said Santa Claus was white and any kind of deviation from that, regardless of its magnitude, is a sin. And we need to make an example out of Mall of America. If we

User 1 sher · 5 Dec 2016
Gee, did Santa and Jesus hang out and pound down a few beers together?

User 2 Zickler · 6 Dec 2016
St. Nicholas was white? Really?? Lol

User 3 Dean · 6 Dec 2016
FYI, this is false.

User 4 rumps · 7 Dec 2016
OMG, Santa is in the Bible

User 5 llies_robert · 6 Dec 2016
I wanted that one. I really did.

Figure 1: A piece of fake news on PolitiFact, and the user comments on social media. Some explainable comments are directly related to the sentences in news contents.

help improve fake news detection performance. However, to our best knowledge, there has been no prior attempt to computationally detect fake news with proper explanation on social media.

In particular, we propose to derive explanation from the perspectives of news contents and user comments (See Figure 1). First, news contents may contain information that is verifiably false. For example, journalists manually check the claims in news articles on fact-checking websites such as PolitiFact[6], which is usually labor-intensive and time-consuming. Researchers also attempt to use external sources to fact-check the claims in news articles to decide and explain whether a news piece is fake or not [6], which may not be able to check newly emerging events (that has not been fact-checked). Second, user comments have rich information from the crowd on social media, including opinions, stances, and sentiment, that are useful to detect fake news. For example, researchers propose to use social features to select important comments to predict fake news pieces [13]. Moreover, news contents and user comments inherently are *related* each other and can provide important cues to explain why a given news article is fake or not. For example, in Figure 1, we can see users discuss different aspects of the news in comments such as "`St. Nicholas was white? Really??Lol`," which directly responds to the claims in the news content "`The Holy Book always said Santa Claus was white`."

Therefore, in this paper, we study the problem of fake news detection by jointly exploring explainable information from news contents and user comments. To this end, we build an explainable fake news detection framework through a coherent process which consists of: (1) a component to encode news contents (to learn the news sentence representations through a hierarchical attention neural network to capture the semantic and syntactic cues), (2) a component to encode user comments (to learn the latent representations of user comments through a word-level attention sub-network), and (3) a sentence-comment co-attention component (to capture the correlation between news contents and comments and to select top-k explainable sentences and comments).

In essence, in this paper, we address the following challenges: (1) How to perform explainable fake news detection that can improve detection performance and explainability simultaneously; (2) How to extract explainable comments without the ground truth during training; and (3) How to model the correlation between news contents and user comments jointly for explainable fake news detection? Our solutions to these challenges result in a novel framework named as dEFEND (Explainable FakE News Detection). Our main contributions are summarized as follows:

- We study a novel problem of explainable fake news detection on social media.
- We provide a principled way to exploit both news contents and user comments jointly to capture explainable user comments for fake news detection; and
- We conduct extensive experiments on real-world datasets to demonstrate the effectiveness of dEFEND for detecting fake news and explaining fake news results.

2 RELATED WORK

In this section, we briefly review the related works on fake news detection and explainable machine learning.

2.1 Fake News Detection

Fake news detection methods generally focus on using *news contents* and *social contexts* [40, 51, 52]. News content features are mainly extracted from textual and visual aspects. Textual features capture specific writing styles [34] and sensational emotions [12] that commonly occur in fake news contents. In addition, latent textual representations are modeled using tensor factorization [15], deep neural networks [20, 21, 44], which achieve good performance to detect fake news with news contents. Visual features are extracted from visual elements (e.g. images and videos) to capture the different characteristics for fake news [19].

For social context based approaches, the features mainly include user-based, post-based and network-based. User-based features are extracted from user profiles to measure their characteristics [3, 42]. Post-based features represent users' social response in term of stances [43], topics [13], or credibility [18]. Network-based features are extracted by constructing specific networks, such as the diffusion networks [46], interaction networks [41], and propagation networks [30, 39]. Recently, research also focuses on challenging problems of fake news detection, such as fake news early detection by adversarial learning [45] and user response generating [35], semi-supervised detection [11] and unsupervised detection [15, 49], and explainable detection of fake news through meta attributes [48].

In this paper, we study the novel problem of explainable fake news detection which aims to improve fake news detection performance, and highlight explainable user comments and check-worthy news sentences simultaneously.

2.2 Explainable Machine Learning

Our work is also related to explainable machine learning, which can generally be grouped into two categories: *intrinsic* explainability and *post-hoc* explainability [8]. Intrinsic explainability is achieved by constructing self-explanatory models which incorporate explainability directly into their structures. The explainability is achieved by finding the features with large coefficients that play key roles in interpreting the predictions [7]. In contrast, the post-hoc explainability requires to create a second model to provide explanation for an existing model. Koh *et al.* [24] proposed to identify training points which are most related to a given prediction result through influence functions. Liu *et al.* propose to interpret network embedding representations via an induction of taxonomy structure [27].

[6] https://www.politifact.com/

Different from traditional machine learning algorithms, the learned representations of deep learning models (DNNs) are usually *not* interpretable by human [8]. Therefore, the explanation for deep neural networks (DNNs) mainly focuses on understanding the representations captured by neurons at intermediate layers of DNNs [9, 26, 28]. Liu *et al.* utilize the interpretation of machine learning models to perform adversarial detection [28]. Du *et al* propose to instance-level interpretation of neural networks through guided feature inversion [47]. Karpathy *et al.* [22] analyzed the interpretability of RNN activation patterns using character level language modeling. Research [33] found that RNN can learn contextual representations by inspecting representations at different hidden layers.

In this paper, we propose to utilize a co-attention mechanism to jointly capture the intrinsic explainability of news sentences and user comments and improve fake news detection performance.

3 PROBLEM STATEMENT

Let A be a news article, consisting of N sentences $\{s_i\}_{i=1}^{N}$. Each sentence $s_i = \{w_1^i, \cdots, w_{M_i}^i\}$ contains M_i words. Let $C = \{c_1, c_2, ..., c_T\}$ be a set of T comments related to the news A, where each comment $c_j = \{w_1^j, \cdots, w_{Q_j}^j\}$ contains Q_j words. Similar to previous research [18, 40], we treat fake news detection problem as the binary classification problem, i.e., each news article can be true ($y = 1$) or fake ($y = 0$). At the same time, we aim to learn a rank list RS from all sentences in $\{s_i\}_{i=1}^{N}$, and a rank list RC from all comments in $\{c_j\}_{j=1}^{T}$, according to the degree of exaplainability, where RS_k (RC_k) denotes the k_{th} most explainable sentence (comment). The explainability of sentences in news contents represent the degree of how check-worthy they are, while the explainability of comments denote the degree of how much users believe if news is fake or real, closely related to the major claims in news. Formally, we can represent the problem as *Explainable Fake News Detection*:

Problem: Explainable Fake News Detection. Given a news article A and a set of related comments C, learn a fake news detection function f: $f(A, C) \rightarrow (\hat{y}, RS, RC)$, such that it maximizes prediction accuracy with explainable sentences and comments ranked highest in RS and RC respectively.

4 dEFEND: EXPLAINABLE FAKE NEWS DETECTION FRAMEWORK

In this section, we present the details of the proposed framework for explainability fake news detection, named as dEFEND (Explainable FakE News Detection). It consists of four major components (see Figure 2): (1) a news content encoder (including word encoder and sentence encoder) component, (2) a user comment encoder component, (3) a sentence-comment co-attention component, and (4) a fake news prediction component.

Specifically first, the news content encoder component describes the modeling from the news linguistic features to latent feature space through a hierarchical word- and sentence-level encoding; next, the user comment encoder component illustrates the comment latent feature extraction through word-level attention networks; then, the sentence-comment co-attention component models the mutual influences between the news sentences and user comments for learning feature representations, and the explainability degree of sentences and comments are learned through the attention weights within co-attention learning; finally, the fake news prediction component shows the process of concatenating news content and user comment features for fake news classification.

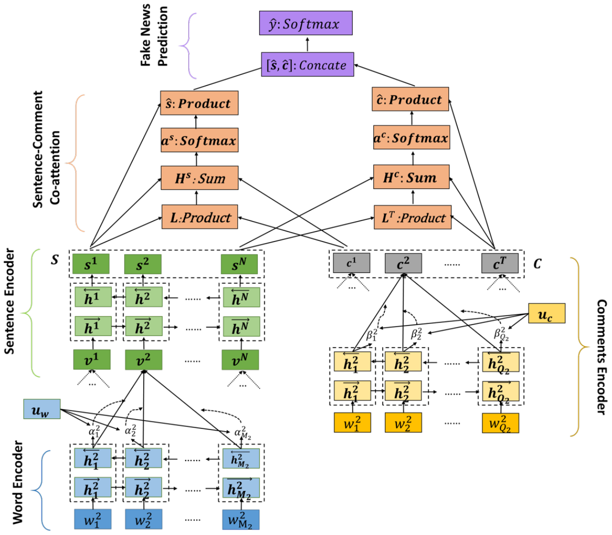


Figure 2: The proposed framework dEFEND **consists of four components: (1) a news content (including word-level and sentence-level) encoder, (2) a user comment encoder, (3) a sentence-comment co-attention component, and (4) a fake news prediction component.**

4.1 News Contents Encoding

As fake news pieces are intentionally created to spread inaccurate information, they often have opinionated and sensational language styles, which have the potential to help detect fake news. In addition, a news document contains linguistic cues with different levels such as word-level and sentence-level, which provide different degrees of importance for the explainability of why the news is fake. For example, in a fake news claim “`Pence: Michelle Obama is the most vulgar first lady we've ever had`”, the word “`vulgar`” contributes more signals to decide whether the news claim is fake rather than other words in the sentence.

Recently, researchers find that hierarchical attention neural networks [50] are very practical and useful to learn document representations [4] with highlighting important words or sentences for classification. It adopts a hierarchical neural network to model word-level and sentence-level representations through self-attention mechanisms. Inspired by [4], we proposed to learn the news content representations through a hierarchical structure. Specifically, we first learn the sentence vectors by using the word encoder with attention and then learn the sentence representations through sentence encoder component.

4.1.1 Word Encoder. We learn the sentence representation via a recurrent neural network (RNN) based word encoder. Though in theory, RNN is able to capture long-term dependency, in practice, the old memory will fade away as the sequence becomes longer. To capture long-term dependencies of RNN, Gated recurrent units (GRU) [5] are used to ensure a more persistent memory. Similar to [50], we adopt GRU to encode the word sequence. To further capture the contextual information of annotations, we use bidirectional GRU [2] to model word sequences from both directions of words. The bidirectional GRU contains the forward GRU $\overrightarrow{f}$ which reads sentence s_i from word w_1^i to $w_{M_i}^i$ and a backward GRU $\overleftarrow{f}$ which reads sentence s_i from word $w_{M_i}^i$ to w_1^i:

$$\begin{aligned} \overrightarrow{\mathbf{h}_t^i} &= \overrightarrow{GRU}(\mathbf{w}_t^i), t \in \{1, \ldots, M_i\} \\ \overleftarrow{\mathbf{h}_t^i} &= \overleftarrow{GRU}(\mathbf{w}_t^i), t \in \{M_i, \ldots, 1\} \end{aligned} \tag{1}$$

We obtain an annotation of word w_t^i by concatenating the forward hidden state $\overrightarrow{\mathbf{h}_t^i}$ and backward hidden state $\overleftarrow{\mathbf{h}_t^i}$, i.e., $\mathbf{h}_t^i = [\overrightarrow{\mathbf{h}_t^i}, \overleftarrow{\mathbf{h}_t^i}]$, which contains the information of the whole sentence centered around w_t^i. Note that not all words contribute equally to the representation of the sentence meaning. Therefore, we introduce an attention mechanism to learn the weights measuring word importance, and the sentence vector $\mathbf{v}^i \in \mathbb{R}^{2d \times 1}$ is computed as follows,

$$\mathbf{v}^i = \sum_{t=1}^{M_i} \alpha_t^i \mathbf{h}_t^i \tag{2}$$

where α_t^i measures the importance of t^{th} word for the sentence s_i, and α_t^i is calculated as follows,

$$\begin{aligned} \mathbf{u}_t^i &= \tanh(\mathbf{W}_w \mathbf{h}_t^i + \mathbf{b}_w) \\ \alpha_t^i &= \frac{\exp(\mathbf{u}_t^i \mathbf{u}_w^\top)}{\sum_{k=1}^{M_i} \exp(\mathbf{u}_k^i \mathbf{u}_w^\top)} \end{aligned} \tag{3}$$

where α_t^i measures the importance of t^{th} word for the sentence s_i, $\mathbf{u}_t^i$ is a hidden representation of $\mathbf{h}_t^i$ obtained by feeding the hidden state $\mathbf{h}_t^i$ to a fully embedding layer, and $\mathbf{u}_w$ is the weight parameter that represents the world-level context vector.

4.1.2 Sentence Encoder. Similar to word encoder, we utilize RNNs with GRU units to encode each sentence in news. We capture the context information in the sentence-level to learn the sentence representations $\mathbf{h}^i$ from the learned sentence vector $\mathbf{v}^i$. Specifically, we can use the bidirectional GRU to encode the sentences as follows:

$$\begin{aligned} \overrightarrow{\mathbf{h}^i} &= \overrightarrow{GRU}(\mathbf{v}^i), i \in \{1, \ldots, N\} \\ \overleftarrow{\mathbf{h}^i} &= \overleftarrow{GRU}(\mathbf{v}^i), i \in \{N, \ldots, 1\} \end{aligned} \tag{4}$$

We obtain sentence annotation $\mathbf{s}^i \in \mathbb{R}^{2d \times 1}$ by concatenating the forward and backward hidden states, i.e., $\mathbf{s}^i = [\overrightarrow{\mathbf{h}^i}, \overleftarrow{\mathbf{h}^i}]$, which captures the context from neighbor sentences around sentence s_i.

4.2 User Comments Encoding

People express their emotions or opinions towards fake news through social media posts such as comments, such as skeptical opinions, sensational reactions, etc. These textual information has been shown to be related to the content of original news pieces. Thus, comments may contain useful semantic information that has the potential to help fake news detection. Next, we demonstrate how to encode the comments to learn the latent representations. The comments extracted from social media are usually short text, so we use RNNs to encode the word sequence in comments directly to learn the latent representations of comments. Similar to the word encoder, we adopt bidirectional GRU to model the word sequences in comments. Specifically, given a comment c_j with words $w_t^j, t \in \{1, \cdots, Q_j\}$, we first map each word w_t^j into the word vector $\mathbf{w}_t^j \in \mathbb{R}^d$ with an embedding matrix. Then, we can obtain the feedforward hidden states $\overrightarrow{\mathbf{h}_t^j}$ and backward hidden states $\overleftarrow{\mathbf{h}_t^j}$ as follows,

$$\begin{aligned} \overrightarrow{\mathbf{h}_t^j} &= \overrightarrow{GRU}(\mathbf{w}_t^j), t \in \{1, \ldots, Q_j\} \\ \overleftarrow{\mathbf{h}_t^j} &= \overleftarrow{GRU}(\mathbf{w}_t^j), t \in \{Q_j, \ldots, 1\} \end{aligned} \tag{5}$$

We further obtain the annotation of word w_t^j by concatenating $\overrightarrow{\mathbf{h}_t^j}$ and $\overleftarrow{\mathbf{h}_t^j}$, i.e., $\mathbf{h}_t^j = [\overrightarrow{\mathbf{h}_t^j}, \overleftarrow{\mathbf{h}_t^j}]$. We also introduce the attention mechanism to learn the weights to measure the importance of each word, and the comment vector $\mathbf{c}^j \in \mathbb{R}^{2d}$ is computed as follows:

$$\mathbf{c}^j = \sum_{t=1}^{Q_j} \beta_t^j \mathbf{h}_t^j \tag{6}$$

where β_t^j measures the importance of t^{th} word for the comment c_j, and β_t^j is calculated as follows,

$$\begin{aligned} \mathbf{u}_t^j &= \tanh(\mathbf{W}_c \mathbf{h}_t^j + \mathbf{b}_c) \\ \beta_t^j &= \frac{\exp(\mathbf{u}_t^j \mathbf{u}_c^\top)}{\sum_{k=1}^{Q_j} \exp(\mathbf{u}_k^j \mathbf{u}_c^\top)} \end{aligned} \tag{7}$$

where $\mathbf{u}_t^j$ is a hidden representation of $\mathbf{h}_t^j$ obtained by feeding the hidden state $\mathbf{h}_t^j$ to a fully embedding layer, and $\mathbf{u}_c$ is the weight.

4.3 Sentence-Comment Co-attention

We observe that not all sentences in news contents are fake, and in fact, some sentences are true but only for supporting wrong claim sentences [10]. Thus, news sentences may not be equally important in determining and explaining whether a piece of news is fake. For example, the sentence "`Michelle Obama is so vulgar she's not only being vocal..`" is strongly related to the fake claim "`Pence: Michelle Obama Is The Most Vulgar First Lady We've Ever Had`", while "`The First Lady denounced the Republican presidential nominee`" expresses some fact and is less helpful in detecting and explaining whether the news is fake.

Similarly, user comments may contain relevant information about the important aspects that explain why a piece of news is fake, while they may also be less informative and noisy. For example, a comment "`Where did Pence say this? I saw him on CBS this morning and he didn't say these things..`" is more explainable and useful to detect the fake news, than other comments such as "`Pence is absolutely right`".

Thus, we aim to select news sentences and user comments that can explain why a piece of news is fake. As they provide a good explanation, they should also be helpful in detecting fake news. This suggests us to design attention mechanisms to give high weights of representations of news sentences and comments that are beneficial to fake news detection. Specifically, we use sentence-comment co-attention because it can capture the semantic affinity of sentences and comments and further help learn the attention weights of sentences and comments simultaneously. We construct the feature matrix of news sentences $\mathbf{S} = [\mathbf{s}^1; \cdots, \mathbf{s}^N] \in \mathbb{R}^{2d \times N}$ and the feature map of user comments $\mathbf{C} = \{\mathbf{c}^1, \cdots, \mathbf{c}^T\} \in \mathbb{R}^{2d \times T}$, the co-attention attends to the sentences and comments simultaneously. Similar to [29], we first compute the affinity matrix $\mathbf{F} \in \mathbb{R}^{T \times N}$ as follows,

$$\mathbf{F} = \tanh(\mathbf{C}^{\mathsf{T}} \mathbf{W}_l \mathbf{S}) \tag{8}$$

where $\mathbf{W}_l \in \mathbb{R}^{2d \times 2d}$ is a weight matrix to be learned through the networks. Following the optimization strategy in [29], we can consider the affinity matrix as a feature and learn to predict sentence and comment attention maps as follows,

$$\begin{aligned} \mathbf{H}^s &= \tanh(\mathbf{W}_s \mathbf{S} + (\mathbf{W}_c \mathbf{C})\mathbf{F}) \\ \mathbf{H}^c &= \tanh(\mathbf{W}_c \mathbf{C} + (\mathbf{W}_s \mathbf{S})\mathbf{F}^{\mathsf{T}}) \end{aligned} \tag{9}$$

where $\mathbf{W}_s, \mathbf{W}_c \in \mathbb{R}^{k \times 2d}$ are the weight parameters. The attention weights of sentences and comments are calculated as follows,

$$\begin{aligned} \mathbf{a}^s &= \text{softmax}(\mathbf{w}_{hs}^{\mathsf{T}} \mathbf{H}^s) \\ \mathbf{a}^c &= \text{softmax}(\mathbf{w}_{hc}^{\mathsf{T}} \mathbf{H}^c) \end{aligned} \tag{10}$$

where $\mathbf{a}^s \in \mathbb{R}^{1 \times N}$ and $\mathbf{a}^c \in \mathbb{R}^{1 \times T}$ are the attention probabilities of each sentence $\mathbf{s}^i$ and comment $\mathbf{c}^j$, respectively. $\mathbf{w}_{hs}, \mathbf{w}_{hc} \in \mathbb{R}^{1 \times k}$ are the weight parameters. The affinity matrix $\mathbf{F}$ transforms user comment attention space to news sentence attention space, and vice versa for $\mathbf{F}^{\mathsf{T}}$. Based on the above attention weights, the comment and sentence attention vectors are calculated as the weighted sum of the comment features and sentence features, i.e.,

$$\hat{\mathbf{s}} = \sum_{i=1}^{N} \mathbf{a}_i^s \mathbf{s}^i, \qquad \hat{\mathbf{c}} = \sum_{j=1}^{T} \mathbf{a}_j^c \mathbf{c}^j \tag{11}$$

where $\hat{\mathbf{s}} \in \mathbb{R}^{1 \times 2d}$ and $\hat{\mathbf{c}} \in \mathbb{R}^{1 \times 2d}$ are the learned features for news sentences and user comments through co-attention.

4.4 The Proposed Framework: dEFEND

We have introduced how we can encode news contents by modeling the hierarchical structure from word level and sentence level, how we encode comments by word-level attention networks, and the component to model co-attention to learn sentences and comments representations. We further integrate these components together and predict fake news with the following objective,

$$\hat{\mathbf{y}} = \text{softmax}([\hat{\mathbf{s}}, \hat{\mathbf{c}}]\mathbf{W}_f + \mathbf{b}_f) \tag{12}$$

where $\hat{\mathbf{y}} = [\hat{\mathbf{y}}_0, \hat{\mathbf{y}}_1]$ is the predicted probability vector with $\hat{\mathbf{y}}_0$ and $\hat{\mathbf{y}}_1$ indicate the predicted probability of label being 0 (real news) and 1 (fake news) respectively. $y \in \{0, 1\}$ denotes the ground truth label of news. $[\hat{\mathbf{s}}, \hat{\mathbf{c}}]$ means the concatenation of learned features for news sentences and user comments. $\mathbf{b}_f \in \mathbb{R}^{1 \times 2}$ is the bias term. Thus, for each news piece, the goal is to minimize the cross-entropy loss function as follows,

$$\mathcal{L}(\theta) = -y \log(\hat{\mathbf{y}}_1) - (1 - y) \log(1 - \hat{\mathbf{y}}_0) \tag{13}$$

where θ denotes the parameters of the network.

The parameters in the network are learned through RMSprop, which is an adaptive learning rate method which divides the learning rate by an exponentially decaying average of squared gradients. We choose RMSprop as the optimizer because it is a popular and effective method for determining the learning rate abortively, which is widely used for training neural networks.

5 EXPERIMENTS

In this section, we present the experiments to evaluate the effectiveness of the proposed dEFEND framework. Specifically, we aim to answer the following evaluation questions:

- **EQ1** Can dEFEND improve fake news classification performance by modeling news contents and user comments simultaneously?
- **EQ2** How effective are news contents and user comments, respectively, in improving the detection performance of dEFEND?
- **EQ3** Can dEFEND capture the news sentences and user comments that can explain why a piece of news is fake?

Table 1: The statistics of FakeNewsNet dataset

Platform	PolitiFact	GossipCop
# Users	68,523	156,467
# Comments	89,999	231,269
# Candidate news	415	5,816
# True news	145	3,586
# Fake news	270	2,230

5.1 Datasets

We utilize one of the comprehensive fake news detection benchmark dataset called FakeNewsNet [38, 40]. The dataset is collected from two platforms with fact-checking: *GossipCop* and *PolitiFact*, both containing news content with labels and social context information. News content includes the meta attributes of the news (e.g., body text), and social context includes the related user social engagements of news items (e.g., user comments in Twitter). Note that we keep news pieces with at least 3 comments. The detailed statistics of the datasets are shown in Table 1.

5.2 Compared Fake News Detection Methods

The representative state-of-the-art fake news detection algorithms are listed as follows:

- **RST** [36]: RST stands for Rhetorical Structure Theory, which builds a tree structure to represent rhetorical relations among the words in the text. RST can extract news style features by mapping the frequencies of rhetorical relations to a vector space[7].
- **LIWC** [32]: LIWC stands for Linguistic Inquiry and Word Count, which is widely used to extract the lexicons falling into psycholinguistic categories. It learns a feature vector from psychology and deception perspective[8].
- **HAN** [50]: HAN utilizes a hierarchical attention neural network framework on news contents for fake news detection. It encodes news contents with word-level attentions on each sentence and sentence-level attentions on each document.
- **text-CNN** [23]: text-CNN utilizes convolutional neural networks to model news contents, which can capture different granularity of text features with multiple convolution filters.
- **TCNN-URG** [35]: TCNN-URG consists of two major components: a two-level convolutional neural network to learn representations from news content, and a conditional variational auto-encoder to capture features from user comments.
- **HPA-BLSTM** [13]: HPA-BLSTM is a neural network model that learns news representation through a hierarchical attention network on word-level, post-level, and sub-event level of user engagements on social media. In addition, post features are extracted to learn the attention weights during post-level.

[7] The code is available at https://github.com/jiyfeng/DPLP

[8] The readers can find more details about the software and feature description at http://liwc.wpengine.com/

Table 2: The performance comparison for fake news detection

Datasets	Metric	RST	LIWC	text-CNN	HAN	TCNN-URG	HPA-BLSTM	CSI	dEFEND
PolitiFact	Accuracy	0.607	0.769	0.653	0.837	0.712	0.846	0.827	**0.904**
	Precision	0.625	0.843	0.678	0.824	0.711	0.894	0.847	**0.902**
	Recall	0.523	0.794	0.863	0.896	0.941	0.868	0.897	**0.956**
	F1	0.569	0.818	0.760	0.860	0.810	0.881	0.871	**0.928**
GossipCop	Accuracy	0.531	0.736	0.739	0.742	0.736	0.753	0.772	**0.808**
	Precision	0.534	**0.756**	0.707	0.655	0.715	0.684	0.732	0.729
	Recall	0.492	0.461	0.477	0.689	0.521	0.662	0.638	**0.782**
	F1	0.512	0.572	0.569	0.672	0.603	0.673	0.682	**0.755**

- **CSI** [37]; CSI is a hybrid deep learning model that utilizes information from text, response, and source. The news representation is modeled via an LSTM neural network with the Doc2Vec [25] embedding on the news contents and user comments as input, and for a fair comparison, the user features are ignored.

Note that for a fair comparison, we choose the above fake news methods that extract features from following aspects: (1) only **news contents**, such as RST, LIWC, text-CNN, HAN; (2) only **user comments**, such as HPA-BLSTM, and (3) both **news contents** and **user comments**, such as TCNN-URG and CSI. For feature extraction methods such as RST and LIWC, we feed them into different learning algorithms and choose the one that achieves the best performance. The algorithms include Logistic Regression, Naive Bayes, Decision, Decision Tree, and Random Forest. We run these algorithms using scikit-learn [31] with default parameter settings.

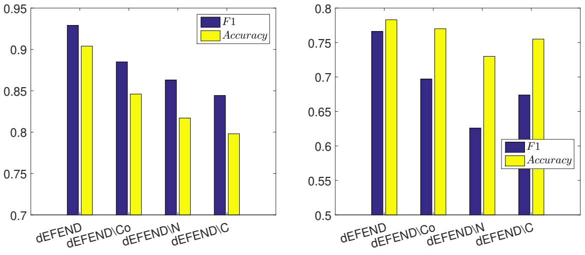
5.3 Fake News Detection Performance

To answer **EQ1**, we first compare dEFEND with representative fake news detection algorithms introduced in Section 5.2.

To evaluate the performance of fake news detection algorithms, we use the following metrics, which are commonly used to evaluate classifiers in related areas: Accuracy, Precision, Recall, and F1. We randomly choose 75% of news pieces for training and remaining 25% for testing, and the process is performed for 5 times and the average performance is reported in Table 2. From the table, we make the following observations:

- For news content based methods RST, LIWC and HAN, we see that $HAN > LIWC > RST$ for both datasets. It indicates: 1) HAN can better capture the syntactic and semantic cues through hierarchical attention neural networks in news contents to differentiate fake and real news; 2) LIWC can better capture the linguistic features in news contents. The good results of LIWC demonstrate that fake news pieces are different from real news in terms of choosing the words that reveal psychometrics characteristics.
- In addition, methods using both news contents and user comments perform better than those methods purely based on news contents, and those methods only based on user comments, i.e., dEFEND $>$ HAN or $HPA - BLSTM$ and $CSI > HAN$ or $HPA - BLSTM$. This indicates that features extracted from news content and corresponding user comments have complementary information, and thus boost the detection performance.
- Moreover, the performance of user comment based methods are slightly better than news content based methods. For example, we have $HPA - BLSTM > HAN$ in terms of Accuracy and F1 on both PolitiFact and Gossipcop data. It shows that features extracted from user comments have more discriminative power than those only on news content for predicting fake news.
- Generally, for methods based on both news content and user comments (i.e., dEFEND, CSI, and $TCNN-URG$), we can see that dEFEND consistently outperforms CSI and $TCNN - URG$ and, i.e., dEFEND $>$ $CSI > TCNN - URG$, in terms of all evaluation metrics on both datasets. For example, dEFEND achieves average relative improvement of 4.5%, 3.6% on PolitiFact and 4.7%, 10.7% on Gossipcop, comparing with CSI in terms of $Accuracy$ and $F1$ score. It supports the importance of modeling co-attention of news sentences and user comments for fake news detection.

5.4 Assessing Impacts of News Contents and User Comments



(a) PolitiFact (b) GossipCop

Figure 3: Impact analysis of news contents, comments, and sentence-comment co-attention for fake news detection.

In addition to news contents, we also capture information from user comments and integrate it with news contents with co-attention. In order to answer **EQ2**, we further investigate the effects of these components by defining three variants of dEFEND:

- dEFEND\C: dEFEND\C is a variant of dEFEND without considering information from user comments. It first encodes news contents with word-level attentions on each sentence, and then the resultant sentence features are averaged through an average pooling layer and feed into a softmax layer for classification.
- dEFEND\N: dEFEND\N is a variant of dEFEND without considering information from news contents. It first utilizes the comment encoder to learn comment features, and then the resultant comment features are averaged through an average pooling layer and feed into a softmax layer for classification.
- dEFEND\Co:dEFEND\Co is a variant of dEFEND, which eliminates the sentence-comment co-attention. Instead, it performs self-attention on sentences and comments separately and the

resultant features are concatenated to a dense layer and feed into a softmax layer for classification.

The parameters in all the variants are determined with cross-validation and the best performances are reported in Figure 3. We make the following observations:

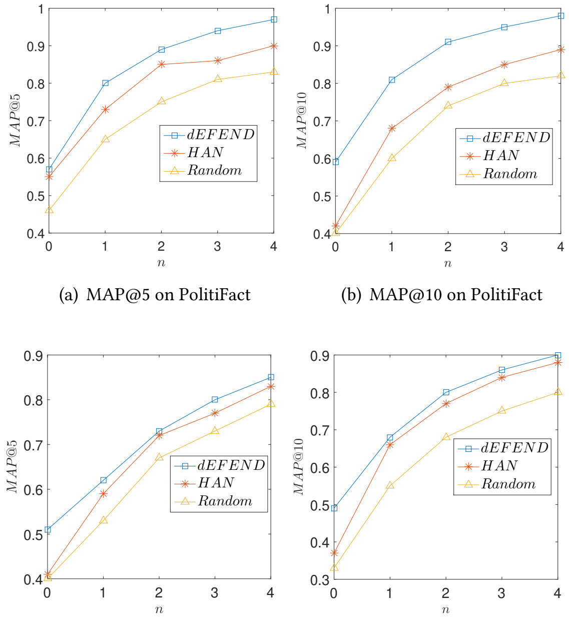
- When we eliminate the co-attention for news contents and user comments, the performances are reduced. It suggests the importance of modeling the correlation and captures the mutual influence between news contents and user comments.
- When we eliminate the effect of news contents, the performance of dEFEND\N degrades in comparison with dEFEND. For example, the performance reduces 4.2% and 6.6% in terms of F1 and Accuracy metrics on PolitiFact, 18.2% and 6.8% on GossipCop. The results suggest that news contents in dEFEND are important.
- We have a similar observation for dEFEND\C when eliminating the effect of user comments. The results suggest the importance to consider the feature of user comments to guide fake news detection in dEFEND.

Through the component analysis of dEFEND, we conclude that (1) both components of news contents and user comments can contribute to the fake news detection performance improvement of dEFEND; (2) it is necessary to model both news contents and user comments because they contain complementary information.

5.5 Explainability Evaluation and Case Study

In this subsection, to answer **EQ3**, we evaluate the performance of explainability of dEFEND framework from the perspective of news sentences and user comments. It is worth mentioning that all of the baseline methods in 5.2 are designed for fake news detection, and none of them are initially proposed to discover explainable news sentences or user comments. To measure the performance of dEFEND for explainability, we choose HAN for comparison of news sentence explainability, and HPA-BLSTM as the baselines for user comments explainability since they can learn attention weights for news sentences and user comments, respectively. Note that HAN uses the attention mechanism to learn the document structure, while HPA-BLSTM utilizes the attention mechanism to learn the temporal structure of comments. Since there is no temporal structure in documents, so HAN cannot be used in comments; Similarly, there are no temporal relations in the document structure, so HPA-BLSTM cannot be directly applied to news contents.

5.5.1 News Sentence Explainability. In this subsection, we demonstrate the performance of the explainability rank list of news sentences, i.e., RS. Specifically, we want to see if the top-ranked explainable sentences determined by our method are more likely to be related to the major claims in fake news that are worth to check–i.e., check-worthy. Therefore, we utilize ClaimBuster [14] to obtain a ground truth rank list $\tilde{RS}$ of all check-worthy sentences in a piece of news content. ClaimBuster proposes a scoring model that utilizes various linguistics features trained using tens of thousands of sentences from past general election debates that were labeled by human coders and gives a "check-worthiness" score between 0 and 1. The higher the score, the more likely the sentence contains check-worthy factual claims. The lower the score, the more non-factual, subjective and opinionated the sentence is. We compare top-k rank list of the explainable sentences in news contents by dEFEND ($RS^{(1)}$) and HAN ($RS^{(2)}$), with top-k rank list, $\tilde{RS}$, by ClaimBuster, using the evaluation metric, MAP@k (Mean Average Precision), where k is set as 5 and 10. We also introduce another parameter n which controls the window size that allows n neighboring sentences are considered when comparing the sentences in $RS^{(1)}$ and $RS^{(2)}$ with each of the top-k sentences in $\tilde{RS}$. From Figure 4, we make the following observations:



(a) MAP@5 on PolitiFact (b) MAP@10 on PolitiFact

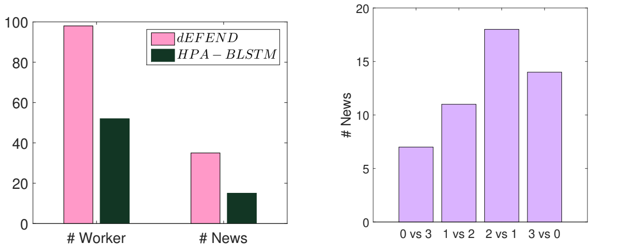
(c) MAP@5 on GossipCop (d) MAP@10 on GossipCop

Figure 4: The performance of sentence explainability on MAP@5 and MAP@10 w.r.t. the neighborhood threshold n.

- In general, we can see that dEFEND > *HAN* > *Random* for the performance of finding check-worthy sentences in news contents on both datasets. It indicates that the sentence-comment co-attention component in dEFEND can help selecting more check-worthy sentences.
- With the increase of n, we relax the condition to match check-worthy sentences in the ground truth, and thus the MAP performance is increasing.
- When $n = 1$, the performance of dEFEND on MAP@5 and MAP@10 increases to exceed 0.8 for PolitiFact, which indicates that dEFEND can detect check-worthy sentences well within 1 neighboring sentence of the ground truth sentences in $\tilde{RS}$.

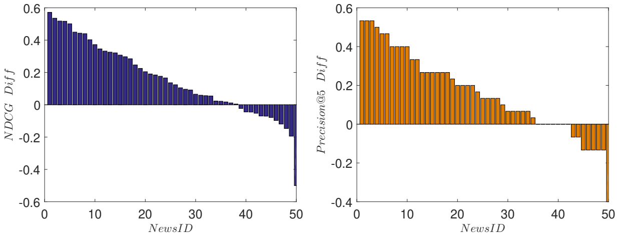
5.5.2 User Comments Explainability. We deploy several tasks using Amazon Mechanical Turk (AMT)[9] to evaluate the explainability rank list of the comments RC for fake news. We perform the following settings to deploy AMT tasks for a total of 50 fake news pieces. For each news article, we first filter out very short articles with less than 50 words. In addition, for very long articles with more than 500 words in content, we presented only the first 500 words to reduce the amount of reading for workers. As the first 3-4 paragraphs of news articles often summarize the content, the first 500 words are usually sufficient to capture the gist of the articles. Then, we recruited AMT workers located in the US (who are more likely to be familiar with the topics of the articles) with the approval rate > 0.95. To evaluate the explainability of user comments, for each news article, we have two lists of top-k comments, $L^{(1)} = (L_1^{(1)}, L_2^{(1)}, \cdots, L_k^{(1)})$ for using

[9] https://www.mturk.com/



(a) Winning Count (b) Worker Voting Ratio (dEFEND vs. HPA-BLSTM)

Figure 5: The human-evaluation of explainable comment list of dEFEND and HPA-BLSTM with Task 1.



(a) NDCG (b) precision@5

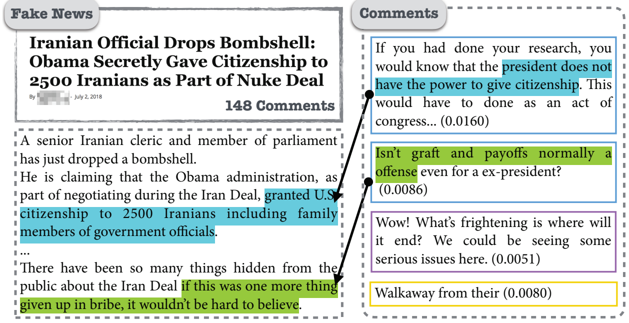
Figure 6: The discrepancy histograms of mean NDCG and mean Precision@5 of the results between two methods.

dEFEND and $L^{(2)} = (L_1^{(2)}, L_2^{(2)}, \cdots, L_k^{(2)})$ for HPA-BLSTM. The top-k comments are selected and ranked using the attention weights from the high to low. To evaluate the model ability to select topmost explainable comments, we empirically set $k = 5$. We deploy two AMT tasks to evaluate the explainable ranking performance.

For **Task 1**, we perform **list-wise** comparison. We ask workers pick a *collectively* better list between $L^{(1)}$ and $L^{(2)}$. To remove the position bias, we randomly assign the position, top and bottom, of $L^{(1)}$ and $L^{(2)}$ when presented to workers. We let each worker pick the better list between $L^{(1)}$ and $L^{(2)}$ for each news piece. We ensure each news piece is evaluated by 3 workers, and finally obtained 150 results of workers' choices. In a worker-level, we compute the number of workers that choose $L^{(1)}$ and $L^{(2)}$, and also compute the winning ratio (WR for short) for them. In a news-level, we perform majority voting for all 3 workers for each news and decide if workers choose $L^{(1)}$ or $L^{(2)}$. For each news, we also compute the worker-level choices by computing the ratio between $L^{(1)}$ and $L^{(2)}$. From Figure 5, we make the following observations:

- dEFEND can select better top-k explainable comments than HPA-BLSTM both in worker-level and news-level. First, in worker-level, 98 out of 150 workers (with WR=0.65) choose $L^{(1)}$ over $L^{(2)}$. Second, in news-level, dEFEND has better performance in 32 out of 50 news pieces (with WR=0.64) than HPA-BLSTM.
- We can see that there are more news pieces such that 3 workers vote unanimously for $L^{(1)}$ (3 vs 0) than the opposite case (0 vs 3) for their explainability i.e., 14>7. Similarly, there are more cases where 2 workers vote for dEFEND than HPA-BLSTM, i.e., 18>11.

For **Task 2**, we perform **item-wise** evaluation. For each comment in $L^{(1)}$ and $L^{(2)}$, we ask workers to choose a score from $\{0, 1, 2, 3, 4\}$, where 0 means "not explainable at all," 1 means "not explainable," 3 means "somewhat explainable," 4 means "highly explainable," and 2 means "somewhere in between." To avoid the bias caused by different user criteria, we shuffle the order of comments in $L^{(1)}$ and $L^{(2)}$, and ask workers to assess how explainable each comment is with respect to the news. To estimate rank-aware explainability of comments (i.e., having a higher ranked explainable comment is more desirable than a lower ranked one), we use NDCG (Normalized Cumulative Gain) [16] and Precision@k as the evaluation metrics. NDCG is widely used in information retrieval to measure document ranking performance in search engines. It can measure how good a ranking is by comparing the proposed ranking with the ideal ranking list measured by user feedback. Precision@k is the proportion of recommended items in a top-k set that are relevant. Similarly, we ensure each news piece is evaluated by 3 workers and obtain a total of 750 results of workers' ratings for each method. The results are shown in Figure 6, where news articles are sorted by the discrepancy in the metrics between the two methods in descending order (e.g., NDCG(dEFEND)- NDCG(HPA-BLSTM)). We show only the results of Precision@5 as those of Precision@10 are similar. We have the following observations:



Iranian Official Drops Bombshell: Obama Secretly Gave Citizenship to 2500 Iranians as Part of Nuke Deal

By [redacted] - July 2, 2018

148 Comments

A senior Iranian cleric and member of parliament has just dropped a bombshell.
He is claiming that the Obama administration, as part of negotiating during the Iran Deal, granted U.S citizenship to 2500 Iranians including family members of government officials.
...
There have been so many things hidden from the public about the Iran Deal if this was one more thing given up in bribe, it wouldn't be hard to believe.

If you had done your research, you would know that the president does not have the power to give citizenship. This would have to done as an act of congress... (0.0160)

Isn't graft and payoffs normally a offense even for a ex-president? (0.0086)

Wow! What's frightening is where will it end? We could be seeing some serious issues here. (0.0051)

Walkaway from their (0.0080)

Figure 7: The explainable comments captured by dEFEND.

- Among 50 fake news articles, dEFEND obtains higher NDCG scores than HPA-BLSRM for 38 cases in terms of the item-wise evaluation. Overall mean NDCG scores over 50 cases for dEFEND and HPA-BLSRM are 0.71 and 0.55, respectively.
- Similar results can be found on Precision@5. dEFEND is superior to HPA-BLSTM on 35 fake news articles and tied on 7 articles. Overall mean Precision@5 scores over 50 cases for dEFEND and HPA-BLSRM are 0.67 and 0.51, respectively.

Case Study. We compare dEFEND with HPA-BLSTM and demonstrate the explainable comments that we correctly ranked high but missed by HPA-BLSTM as in Figure 7. We can see that: (1) dEFEND can rank more explainable comments higher than non-explainable comments. For example, comment "`...president does not have the power to give citizenship...`" is ranked at the top, which can explain exactly why the sentence "`granted U.S. citizenship to 2500 Iranians including family members of government officials`" in the news content is fake; (2) we can give higher weights to explainable comments than those interfering and unrelated comments, which can help select more related comments to help detect fake news. For example, unrelated comment "`Walkaway from their...`" has an attention weight 0.0080, which is less than an explainable comment "`Isn't graft and payoffs normally a offense`" with an attention weight 0.0086, so the latter comment is selected to be a more important feature for fake news prediction.

6 CONCLUSION AND FUTURE WORK

Fake news detection is attracting growing attention in recent years. However, it is also important to understand why a piece of news is detected as fake. We study the novel problem of explainable fake news detection which aims to: 1) improve detection performance significantly; and 2) discover explainable news sentences and user comments to understand why news pieces are identified as fake. We propose a deep hierarchical co-attention network to learn feature representations for fake news detection and explainable sentences/comments discovery. Experiments on real-world datasets demonstrate the effectiveness of the proposed framework. For future work, first, we can incorporate the fact-checking contents from journalist experts or fact-checking websites to further guide the learning process to obtain check-worthy news sentences. Second, we will explore how to use other user engagements as side information such as likes to help discover explainable comments. Third, we can consider the credibility of the users who posts explainable comments to further improve fake news detection performance.

7 ACKOWLEDGMENTS

This material is in part supported by the NSF awards #1614576, #1742702, #1820609, and #1915801, ONR grant N00014-17-1-2605 and N000141812108, and ORAU-directed R&D program in 2018.

REFERENCES

[1] Hunt Allcott and Matthew Gentzkow. 2017. *Social media and fake news in the 2016 election.* Technical Report. National Bureau of Economic Research.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[3] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on twitter. In *WWW.*

[4] Huimin Chen, Maosong Sun, Cunchao Tu, Yankai Lin, and Zhiyuan Liu. 2016. Neural sentiment classification with user and product attention. In *EMNLP.*

[5] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[6] Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. 2015. Computational fact checking from knowledge networks. *PloS one* 10, 6 (2015), e0128193.

[7] Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608* (2017).

[8] Mengnan Du, Ninghao Liu, and Xia Hu. 2018. Techniques for Interpretable Machine Learning. *arXiv preprint arXiv:1808.00033* (2018).

[9] Mengnan Du, Ninghao Liu, Qingquan Song, and Xia Hu. 2018. Towards explanation of dnn-based prediction with guided feature inversion. In *KDD.*

[10] Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Syntactic stylometry for deception detection. In *ACL.*

[11] Gisel Bastidas Guacho, Sara Abdali, Neil Shah, and Evangelos E Papalexakis. 2018. Semi-supervised Content-based Detection of Misinformation via Tensor Embeddings. In *ASONAM.*

[12] Chuan Guo, Juan Cao, Xueyao Zhang, Kai Shu, and Miao Yu. 2019. Exploiting Emotions for Fake News Detection on Social Media. *arXiv preprint arXiv:1903.01728* (2019).

[13] Han Guo, Juan Cao, Yazi Zhang, Junbo Guo, and Jintao Li. 2018. Rumor Detection with Hierarchical Social Attention Network. In *CIKM.*

[14] Naeemul Hassan, Fatma Arslan, Chengkai Li, and Mark Tremayne. 2017. Toward automated fact-checking: Detecting check-worthy factual claims by ClaimBuster. In *KDD.*

[15] Seyedmehdi Hosseinimotlagh and Evangelos E Papalexakis. 2018. Unsupervised Content-Based Identification of Fake News Articles with Tensor Decomposition Ensembles. (2018).

[16] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *TOIS* (2002).

[17] Yangfeng Ji and Jacob Eisenstein. 2014. Representation learning for text-level discourse parsing. In *ACL.*

[18] Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. 2016. News Verification by Exploiting Conflicting Social Viewpoints in Microblogs.. In *AAAI.*

[19] Zhiwei Jin, Juan Cao, Yongdong Zhang, Jianshe Zhou, and Qi Tian. 2017. Novel visual and statistical image features for microblogs news verification. *Transactions on Multimedia* (2017).

[20] Hamid Karimi, Proteek Roy, Sari Saba-Sadiya, and Jiliang Tang. 2018. Multi-Source Multi-Class Fake News Detection. In *COLING.*

[21] Hamid Karimi and Jiliang Tang. 2019. Learning Hierarchical Discourse-level Structure for Fake News Detection. *arXiv preprint arXiv:1903.07389* (2019).

[22] Andrej Karpathy, Justin Johnson, and Li Fei-Fei. 2015. Visualizing and understanding recurrent networks. *arXiv preprint arXiv:1506.02078* (2015).

[23] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

[24] Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. *arXiv preprint arXiv:1703.04730* (2017).

[25] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning.* 1188–1196.

[26] Ninghao Liu, Mengnan Du, and Xia Hu. 2019. Representation Interpretation with Spatial Encoding and Multimodal Analytics. In *WSDM.*

[27] Ninghao Liu, Xiao Huang, Jundong Li, and Xia Hu. 2018. On Interpretation of Network Embedding via Taxonomy Induction. In *KDD.*

[28] Ninghao Liu, Hongxia Yang, and Xia Hu. 2018. Adversarial detection with model interpretation. In *KDD.*

[29] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *NIPS.*

[30] Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M Bronstein. 2019. Fake News Detection on Social Media using Geometric Deep Learning. *arXiv preprint arXiv:1902.06673* (2019).

[31] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *JMLR* (2011).

[32] James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. 2015. *The development and psychometric properties of LIWC2015.* Technical Report.

[33] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365* (2018).

[34] Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2017. A Stylometric Inquiry into Hyperpartisan and Fake News. *arXiv preprint arXiv:1702.05638* (2017).

[35] Feng Qian, Chengyue Gong, Karishma Sharma, and Yan Liu. 2018. Neural User Response Generator: Fake News Detection with Collective User Intelligence.. In *IJCAI.*

[36] Victoria L Rubin, N Conroy, and Yimin Chen. 2015. Towards news verification: Deception detection methods for news discourse. In *Hawaii International Conference on System Sciences.*

[37] Natali Ruchansky, Sungyong Seo, and Yan Liu. 2017. Csi: A hybrid deep model for fake news detection. In *CIKM.*

[38] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2018. FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media. *arXiv preprint arXiv:1809.01286* (2018).

[39] Kai Shu, Deepak Mahudeswaran, Suhang Wang, and Huan Liu. 2019. Hierarchical Propagation Networks for Fake News Detection: Investigation and Exploitation. *arXiv preprint arXiv:1903.09196* (2019).

[40] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake News Detection on Social Media: A Data Mining Perspective. *KDD exploration newsletter* (2017).

[41] Kai Shu, Suhang Wang, and Huan Liu. 2019. Beyond News Contents: The Role of Social Context for Fake News Detection. (2019).

[42] Kai Shu, Xinyi Zhou, Suhang Wang, Reza Zafarani, and Huan Liu. 2018. The Role of User Profile for Fake News Detection. *arXiv preprint arXiv:1904.13355* (2018).

[43] Eugenio Tacchini, Gabriele Ballarin, Marco L Della Vedova, Stefano Moret, and Luca de Alfaro. 2017. Some Like it Hoax: Automated Fake News Detection in Social Networks. *arXiv preprint arXiv:1704.07506* (2017).

[44] William Yang Wang. 2017. " Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. *arXiv preprint arXiv:1705.00648* (2017).

[45] Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. 20187. EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. In *KDD.*

[46] Liang Wu and Huan Liu. 2018. Tracing fake-news footprints: Characterizing social media messages by how they propagate. In *WSDM.*

[47] Fan Yang, Ninghao Liu, Suhang Wang, and Xia Hu. 2018. Towards Interpretation of Recommender Systems with Sorted Explanation Paths. In *ICDM.*

[48] Fan Yang, Shiva K Pentyala, Sina Mohseni, Mengnan Du, Hao Yuan, Rhema Linder, Eric D Ragan, Shuiwang Ji, and Xia Ben Hu. 2019. XFake: Explainable Fake News Detector with Visualizations. (2019).

[49] Shuo Yang, Kai Shu, Suhang Wang, Renjie Gu, Fan Wu, and Huan Liu. 2019. Unsupervised Fake News Detection on Social Media: A Generative Approach. In *AAAI.*

[50] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *NAACL.*

[51] Xinyi Zhou and Reza Zafarani. 2018. Fake News: A Survey of Research, Detection Methods, and Opportunities. *arXiv preprint arXiv:1812.00315* (2018).

[52] Xinyi Zhou, Reza Zafarani, Kai Shu, and Huan Liu. 2019. Fake News: Fundamental Theories, Detection Strategies and Challenges. In *WSDM.*

8 APPENDIX ON REPRODUCIBILITY

In this section, we provide more details of the experimental setting and configuration to enable the reproducibility of our work.

8.1 Fake News Detection

We compared the proposed framework, dEFEND, with 7 baseline methods discussed in Section 5.2, including RST, LIWC, text-CNN, HAN, TCNN-URG, HPA-BLSTM, and CSI. All codes that we have implemented are available under the folder "Fake new detection" through the following link: **https://tinyurl.com/ybl6gqrm**. Other codes were obtained as follows:

- RST: we used the publicly available implementation for paper [17]: https://github.com/jiyfeng/DPLP
- LIWC: we used the publicly available tool at: http://liwc.wpengine.com/
- text-CNN: we used the publicly available implementation at: https://github.com/dennybritz/cnn-text-classification-tf
- HAN: we used the publicly available implentation at: https://github.com/richliao/textClassifier
- TCNN-URG: we implemented this algorithm based on the description in the paper [35], and shared the code, named as *tcnn.py*, in the above link
- HPA-BLSTM: we used the implementation provided by the authors of [13]
- CSI: we used the implementation available at: https://github.com/sungyongs/CSI-Code
- dEFEND: we implemented our algorithm in Python–*defend.py* for main algorithm and *go_defend.py* for data processing–and shared them in the above link.

For the dataset, we also used a publicly available dataset, FakeNewsNet [38], available at: https://github.com/KaiDMML/FakeNewsNet. For parameter settings for dEFEND, we introduce the details of major parameter setting as shown in Table 3. The descriptions of the major parameters are as follows:

- MAX_SENTENCE_LENGTH: the threshold to control the maximum length of news sentences
- MAX_SENTENCE_COUNT: the threshold to control the maximum count of sentences
- MAX_COMMENT_LENGTH: the threshold to control the maximum length of user comments
- MAX_COMMENT_COUNT: the threshold to control the maximum count of user comments
- Vocabulary Size: the threshold to control the maximum size of vocabulary
- Embedding Dimension: the dimension of embedding layer
- Word Embedding: the word emending package used for initialize the word vectors
- d: the size of hidden states for BLSTM
- k: the size of attention maps as in Eqn. 9

8.2 Explainability on News and Comments

We elaborate further details on how we evaluated the explainability of sentences and comments in experiments.

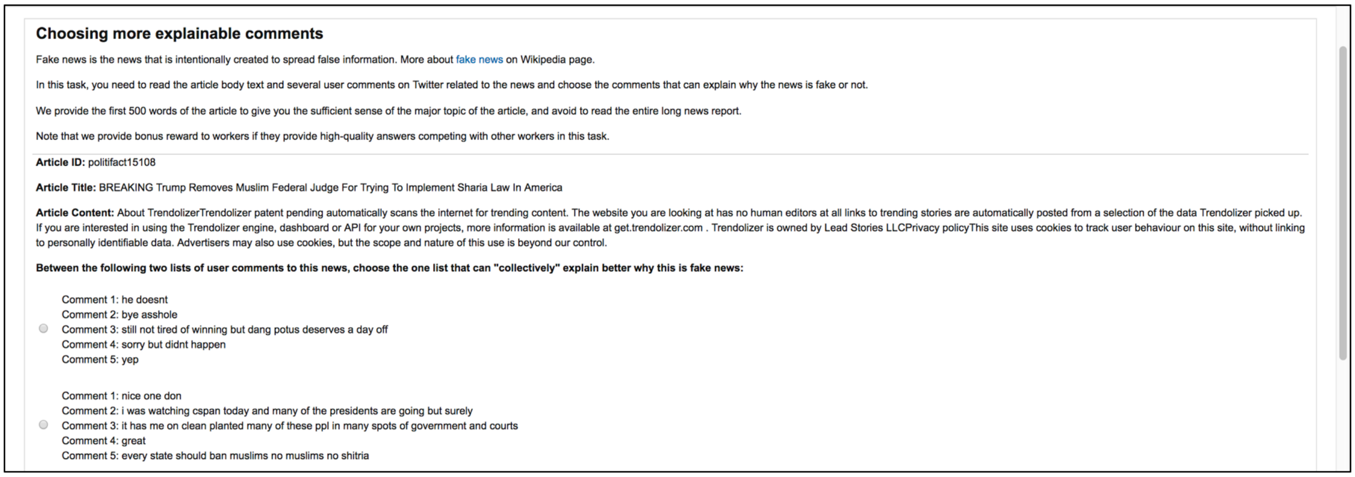
8.2.1 News Sentences. We obtained the ground truth of check-worthy sentences from the online tool, ClaimBuster, with its default setting. ClaimBuster is available at: https://idir-server2.uta.edu/claimbuster/.

Table 3: The details of the parameters of dEFEND

Parameter	PolitiFact	GossipCop
MAX_SENTENCE_LENGTH	120	120
MAX_SENTENCE_COUNT	50	50
MAX_COMMENT_COUNT	150	150
MAX_COMMENT_LENGTH	120	120
Word Embedding	Glove[10]	Glove[11]
Embedding Dimension	100	100
d	100	100
k	80	80
Batch Size	30	20
Maximum Epochs	20	20
Vocabulary Size	20,000	20,000
Learning Rate	0.01	0.001
RMSprop parameter (ρ)	0.9	0.9
RMSprop parameter (ϵ)	1e-8	1e-8
RMSprop parameter (decay)	0	0

8.2.2 User Comments. We introduce the details of the two tasks we deployed at Amazon Mechanical Turk.

- **Task 1**. We presented each fake article with two lists of top-k comments identified by dEFEND and HPA-LSTM, and let workers choose the list that can "collectively" explain better why this is fake news. In order to remove the position bias, we shuffled the order of the two lists randomly, between top and bottom. Each Human Intelligence Task (HIT) contains five fake articles, and was assigned to three distinct workers. The HIT screen-shot for Task 1 is shown in Figure 8.
  To ensure the quality of crowdsourcing task, we set two requirements for AMT workers: (1) the approved percentage of assignments of a worker should be greater than 95%; and (2) the location of a worker should be in US.
- **Task 2**. We tested the explainability of user comments detected by dEFEND and HPA-LSTM, respectively. We presented each fake news article with a list of "mixed" comments identified by two methods, and let workers to assign a score of 0-4 to each comment, where 0 means "not explainable at all," 1 means "not much explainable," 3 means "explainable a bit," 4 means "highly explainable," and 2 means "somewhere in between." Note that in order to remove the position bias, we shuffled the order of the comments randomly. Further, to ensure the quality of the task, we applied the same two requirements as in Task 1. Each HIT had one fake article, and was assigned to 3 distinct workers. The HIT screen-shot of Task 2 is shown in Figure 9.



Choosing more explainable comments

Fake news is the news that is intentionally created to spread false information. More about fake news on Wikipedia page.

In this task, you need to read the article body text and several user comments on Twitter related to the news and choose the comments that can explain why the news is fake or not.

We provide the first 500 words of the article to give you the sufficient sense of the major topic of the article, and avoid to read the entire long news report.

Note that we provide bonus reward to workers if they provide high-quality answers competing with other workers in this task.

Article ID: politifact15108

Article Title: BREAKING Trump Removes Muslim Federal Judge For Trying To Implement Sharia Law In America

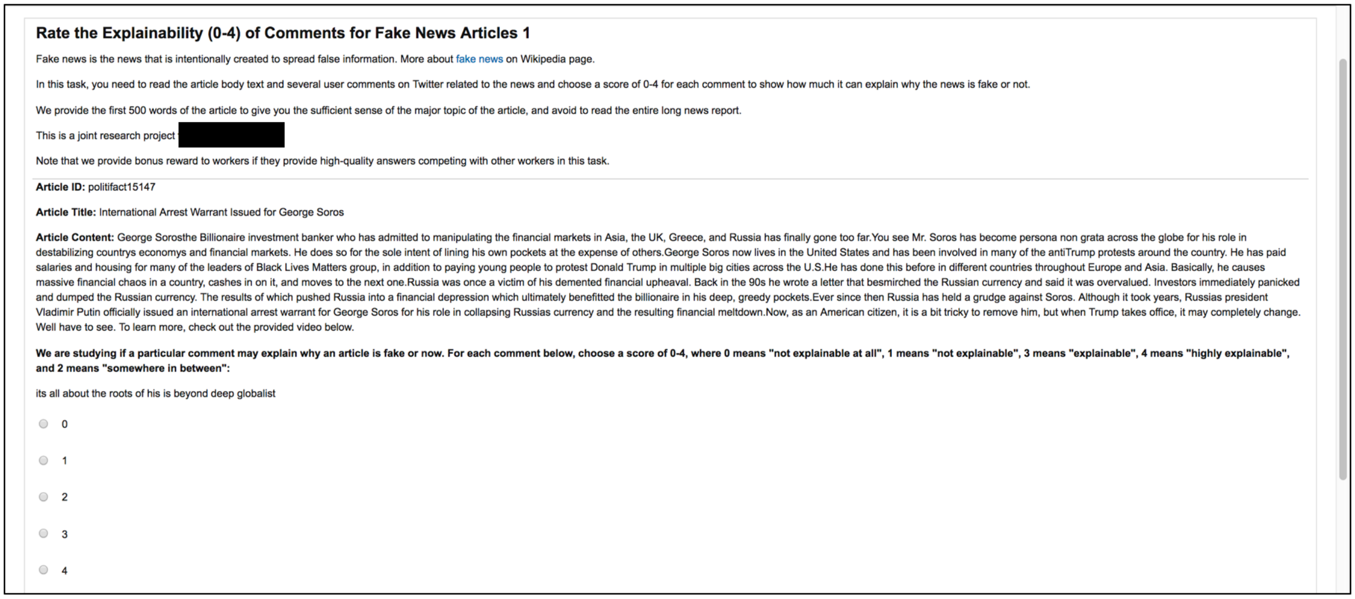
Article Content: About TrendolizerTrendolizer patent pending automatically scans the internet for trending content. The website you are looking at has no human editors at all links to trending stories are automatically posted from a selection of the data Trendolizer picked up. If you are interested in using the Trendolizer engine, dashboard or API for your own projects, more information is available at get.trendolizer.com . Trendolizer is owned by Lead Stories LLCPrivacy policyThis site uses cookies to track user behaviour on this site, without linking to personally identifiable data. Advertisers may also use cookies, but the scope and nature of this use is beyond our control.

Between the following two lists of user comments to this news, choose the one list that can "collectively" explain better why this is fake news:

○ Comment 1: he doesnt
Comment 2: bye asshole
Comment 3: still not tired of winning but dang potus deserves a day off
Comment 4: sorry but didnt happen
Comment 5: yep

○ Comment 1: nice one don
Comment 2: i was watching cspan today and many of the presidents are going but surely
Comment 3: it has me on clean planted many of these ppl in many spots of government and courts
Comment 4: great
Comment 5: every state should ban muslims no muslims no shitria

Figure 8: Task 1: Choosing collectively more explainable user comments for fake news articles.



Rate the Explainability (0-4) of Comments for Fake News Articles 1

Fake news is the news that is intentionally created to spread false information. More about fake news on Wikipedia page.

In this task, you need to read the article body text and several user comments on Twitter related to the news and choose a score of 0-4 for each comment to show how much it can explain why the news is fake or not.

We provide the first 500 words of the article to give you the sufficient sense of the major topic of the article, and avoid to read the entire long news report.

This is a joint research project [redacted]

Note that we provide bonus reward to workers if they provide high-quality answers competing with other workers in this task.

Article ID: politifact15147

Article Title: International Arrest Warrant Issued for George Soros

Article Content: George Sorosthe Billionaire investment banker who has admitted to manipulating the financial markets in Asia, the UK, Greece, and Russia has finally gone too far.You see Mr. Soros has become persona non grata across the globe for his role in destabilizing countrys economys and financial markets. He does so for the sole intent of lining his own pockets at the expense of others.George Soros now lives in the United States and has been involved in many of the antiTrump protests around the country. He has paid salaries and housing for many of the leaders of Black Lives Matters group, in addition to paying young people to protest Donald Trump in multiple big cities across the U.S.He has done this before in different countries throughout Europe and Asia. Basically, he causes massive financial chaos in a country, cashes in on it, and moves to the next one.Russia was once a victim of his demented financial upheaval. Back in the 90s he wrote a letter that besmirched the Russian currency and said it was overvalued. Investors immediately panicked and dumped the Russian currency. The results of which pushed Russia into a financial depression which ultimately benefitted the billionaire in his deep, greedy pockets.Ever since then Russia has held a grudge against Soros. Although it took years, Russias president Vladimir Putin officially issued an international arrest warrant for George Soros for his role in collapsing Russias currency and the resulting financial meltdown.Now, as an American citizen, it is a bit tricky to remove him, but when Trump takes office, it may completely change. Well have to see. To learn more, check out the provided video below.

We are studying if a particular comment may explain why an article is fake or now. For each comment below, choose a score of 0-4, where 0 means "not explainable at all", 1 means "not explainable", 3 means "explainable", 4 means "highly explainable", and 2 means "somewhere in between":

its all about the roots of his is beyond deep globalist

○ 0

○ 1

○ 2

○ 3

○ 4

Figure 9: Task 2: Rating the explainability (0-4) of user comments for fake news articles.