Integrating Pattern- and Fact-based Fake News Detection via Model Preference Learning

Qiang Sheng, Xueyao Zhang
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
shengqiang18z@ict.ac.cn
zhangxueyao19s@ict.ac.cn

Juan Cao
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
caojuan@ict.ac.cn

Lei Zhong
Institute of Computing Technology, Chinese Academy of Sciences
University of Chinese Academy of Sciences
zhonglei18s@ict.ac.cn

ABSTRACT

To defend against fake news, researchers have developed various methods based on texts. These methods can be grouped as 1) *pattern-based* methods, which focus on shared patterns among fake news posts rather than the claim itself; and 2) *fact-based* methods, which retrieve from external sources to verify the claim's veracity without considering patterns. The two groups of methods, which have different preferences of textual clues, actually play complementary roles in detecting fake news. However, few works consider their integration. In this paper, we study the problem of integrating pattern- and fact-based models into one framework via modeling their preference differences, i.e., making the pattern- and fact-based models focus on respective preferred parts in a post and mitigate interference from non-preferred parts as possible. To this end, we build a Preference-aware Fake News Detection Framework (**Pref-FEND**), which learns the respective preferences of pattern- and fact-based models for joint detection. We first design a heterogeneous dynamic graph convolutional network to generate the respective preference maps, and then use these maps to guide the joint learning of pattern- and fact-based models for final prediction. Experiments on two real-world datasets show that Pref-FEND effectively captures model preferences and improves the performance of models based on patterns, facts, or both.

CCS CONCEPTS

• **Information systems** → **Data mining**; • **Computing methodologies** → **Natural language processing**.

KEYWORDS

fake news detection, preference learning, graph neural networks, pattern mining, fact-checking

The authors are at the Key Lab of Intelligent Information Processing of Chinese Academy of Sciences. Qiang Sheng and Xueyao Zhang contributed equally. Corresponding Author: Juan Cao.

CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482440

ACM Reference Format:
Qiang Sheng, Xueyao Zhang, Juan Cao, and Lei Zhong. 2021. Integrating Pattern- and Fact-based Fake News Detection via Model Preference Learning. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3459637.3482440

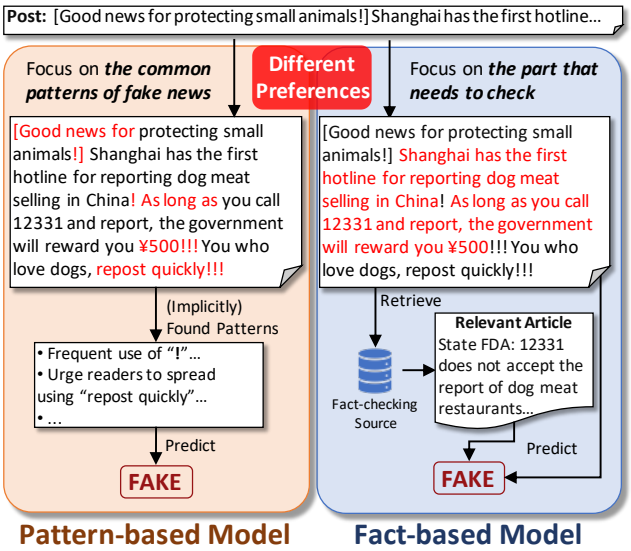

Figure 1: A motivating example. Ideally, given the same news post, the pattern-based and the fact-based model have *different preferences* on textual clues to predict whether the post is fake. The post is translated into English.

1 INTRODUCTION

Fake news that spreads on "online" social media continually causes "offline" real-world harms in crucial domains, such as politics [11], finance [25], and public security [36]. The most recent example is the COVID-19 infodemic [30] where thousands of fake news pieces spread through social media [53]. Under such severe circumstances, developing fake news detection systems has been critical for maintaining a trustful online news ecosystem.

To detect fake news on social media, researchers propose to extract hand-crafted features or deep-learning features [4] from contents, social contexts, propagation networks, etc. In this paper, we focus on the deep learning methods based on textual contents, which can be grouped as: **1) Pattern-based methods** (e.g.,

[13, 21, 52, 64]), which aim at learning shared features (patterns) among fake news posts and expect these features to generalize to unseen news posts. Once trained, they can operate without reliance on external resources. **2) Fact-based methods** (e.g., [27, 34, 48, 57]), which focus on the claim's veracity itself with the help from external fact-checking sources. The key difference between these two methods lies in their different preferences of textual clues. As Figure 1 shows, given the post about a newly opened hotline that accepts reports of dog meat selling, an ideal pattern-based model tends to predict the veracity relying more on the highly frequent use of exclamation marks or the words that urge readers to repost ("repost quickly"), while an ideal fact-based one retrieves to check whether the hotline accepts reports of dog meat selling. From the motivating example, we see that the different preferences of the two models lead to their *complementary* roles. This inspires us to integrate patten- and fact-based models with considering their preferences, which may bring additional gain for fake news detection. However, how to effectively integrate them remains under-explored by existing works.

In this paper, we first study the problem of integrating the pattern- and fact-based models into one framework. The challenge lies in preference modeling: The models, though having different preferences, generally lack the constraints to make themselves focus on preferred parts and ignore non-preferred parts of inputs. As a consequence, a pattern-based model may overfit by *memorizing* frequently shown non-preferred words (e.g., event-specific words) in the training set, and a fact-based one may be distracted from the part that describes a verifiable event. Moreover, the preference of each model should be dynamically determined with contexts, making rule-based modeling inapplicable.

To address these aforementioned challenges, we propose to learn the models' preferences simultaneously with joint fake news detection and build Preference-aware Fake News Detection Framework (**Pref-FEND**). As Figure 2(a) shows, Pref-FEND generates preference maps to assist each model to focus on its expected preferred part. Specifically, we exploit the prior knowledge verified by existing works (e.g., [5, 46, 64]) to recognize cue tokens for patterns and facts, and obtain three sets of tokens (i.e., stylistic tokens, entities, and others). Then, we use a graph-based preference learner to dynamically learn the preferences within the contexts, as presented in Figure 2(b). We construct a heterogeneous graph using these sets and design a Heterogeneous Dynamic Graph Convolutional Network (HetDGCN) for node correlation learning. The final correlation matrix is used by two preference-aware readout functions to generate the Fact and the Pattern Preference Map, respectively. For joint fake news detection, we feed the post and the Map to each model and fuse their last-layer features for final prediction. During training, besides the normal classification loss, we design two auxiliary losses as enhancements, which respectively minimize the similarity between the two maps and the classification loss when the input maps are exchanged and ground-truth labels are reversed. Experimental results on two real-world datasets show that our proposed Pref-FEND can effectively learn the models' preferences and improve the performance of both single preference (pattern- or fact-based) and integrated (pattern-and-fact-based) models. Our contributions are summarized as follows:

- To the best of our knowledge, our work is the first that combines pattern- and fact-based fake news detection. We discuss their complementary roles in fake news detection and propose to consider their preferences for better integration.
- We propose a novel framework, Pref-FEND, which leverages a heterogeneous dynamic GCN to learn model preferences and effectively integrates them for fake news detection.
- Extensive experiments on two newly constructed datasets demonstrate the effectiveness of Pref-FEND on learning models' preferences and improving the detection performance for both single-preference models and integrated models. The code and datasets are available at https://github.com/ICTMCG/Pref-FEND.

2 RELATED WORK

2.1 Fake News Detection

Fake news detection aims at automatically classifying a news piece as real or fake. Existing methods mostly capture features from contents (texts or/and images) and social contexts that generate in the spread process, such as propagation networks [32, 41, 44, 65], user profiles [43], metadata [28], and crowd feedbacks [37, 40, 64]. In this paper, we focus on the text-based methods which can be grouped as:

Pattern-based Fake News Detection. As fake news often contains opinionated and inflammatory language to attract readers [42], common patterns that are different from real news are shared across fake news pieces of different topics. In the very first work on evaluating information credibility on social media, Castillo et al. [5] list a series of post-based features, including the length, whether the post contains exclamation or question marks, etc. Following this line, Volkova et al. [49] inject subjectivity, psycholinguistic, and moral foundations features into deep neural networks (CNNs and RNNs). Przybyla [35] focuses on writing styles. Some works attempt to differentiate the patterns across multiple topical categories [31, 45]. A recent trend of pattern-based methods is to refocus on the sentiment and emotional patterns [1, 13, 14, 64], as the use of eye-catching terms in deceptive and fake posts may manipulate the readers' emotions [6].

Fact-based Fake News Detection. These methods judge the veracity of a news piece (or, a claim) more objectively, with references to pre-constructed external resources such as knowledge graphs [7, 62], online encyclopedias [46], and scientific articles [50]. A more flexible way is to directly use articles retrieved by search engines as evidence to predict the news veracity [2, 34]. Popat et al. [34] use post-specific attention to model the post-article interactions, while the following works [27, 48, 57–59] consider text entailment, such as coherence and conflicts using the attention mechanism. Note that the claims provided by the datasets for evaluation of fact-based methods is generally *normalized* by the human fact-checkers to be declarative and concise, so they are not suitable to evaluate the pattern-based ones. In this paper, we construct two new datasets (in English and Chinese) by referring to existing datasets and external sources for evaluation of pattern-and-fact-based methods.

Different from the above methods, our work do not develop better pattern- or fact-based methods, but integrate the existing ones for comprehensively detecting fake news based on texts.

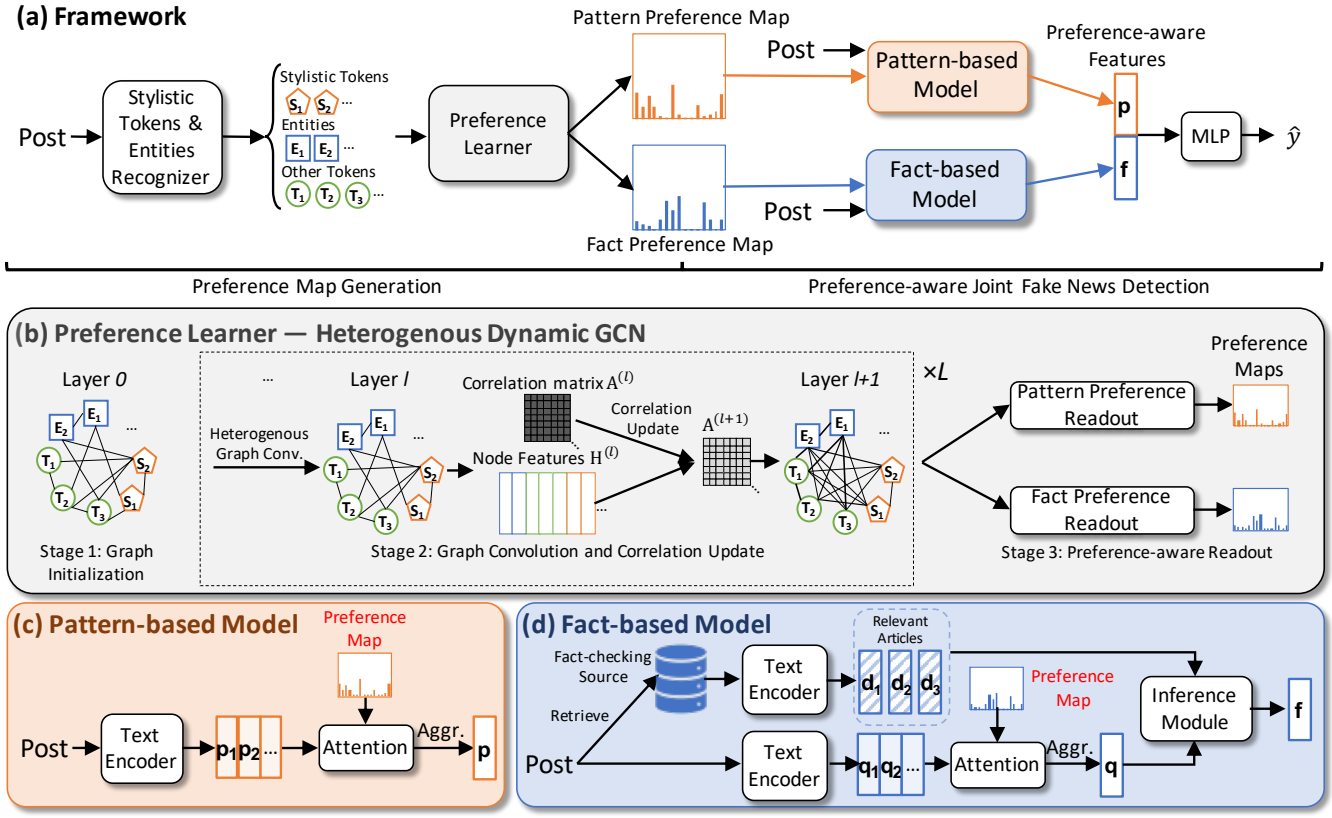

Figure 2: Architecture of Pref-FEND. (a) Overall framework. The post is divided into three sets: stylistic tokens, entities, and others. Then they are fed into a preference learner to generate two preference maps, which highlight the preferred information of the downstream detection models. The preference-aware features are concatenated for final prediction. (b) The Preference Learner, heterogeneous Dynamic GCN, leverages a heterogeneous graph convolution to aggregate multi-type neighbors and updates the correlation matrix every layer. Two readouts use the correlation matrix of the L-th layer to generate preference maps. Only parts of nodes and edges are shown. (c) and (d) exemplify how a pattern-based and a fact-based model works with the preference map, respectively. With the Maps, (c) and (d) attend to helpful tokens for capturing patterns or facts.

2.2 Graph Neural Networks for Text Mining

Due to its expressive power for integrating structural and semantic information, graph neural networks (GNNs) have been widely used for applications in text mining such as information extraction [17] and sentiment analysis [51]. Most works use homogenous GNNs which treat nodes as the same type. Hu et al. [19] leverages a heterogeneous GNN to handle multiple types of nodes such as topics and entities for text classification. Similarly, we use heterogeneous GNN to obtain the preference scores of each token, but our graph is dynamic as its node correlation matrix is adjustable (inspired by [61]). The final adjusted correlations will be aggregated to obtain preference scores.

3 PROBLEM STATEMENT

Let P be a news post on social media containing n tokens. Let D be the set of relevant articles of P. D is retrieved from a fact-checking source $\mathcal{D}$. Following most existing works, we treat fake news detection as a binary classification problem. The ground-truth label y is 1 if P is fake, otherwise 0. We formulate the following tasks:

Pattern-based Fake News Detection: Given P, learn a function $f_P : f_P(P) \rightarrow \hat{y}$, such that it maximizes the predictive accuracy w.r.t. y.

Fact-based Fake News Detection: Given P, retrieve relevant articles D from $\mathcal{D}$, learn a function $f_F : f_F(P, D) \rightarrow \hat{y}$, such that it maximizes the predictive accuracy w.r.t. y.

Joint Pattern-and-Fact-based Fake News Detection: Given P, D, a pattern-based model f_P and a fact-based model f_F, learn a function $f : f(P, D, f_P, f_F) \rightarrow \hat{y}$, such that it maximizes the predictive accuracy w.r.t. y.

4 PROPOSED FRAMEWORK

Figure 2(a) overviews the architecture of the proposed Pref-FEND, whose goal is to learn the models' preferences and employ them for better joint fake news detection. Given a post P, Pref-FEND

Table 1: Types of Stylistic Tokens and References.

Type	For Weibo	For Twitter
Negation Word	HowNet Bilingual Dictionary [9]	
Degree Word		
Sentiment Word		
Proposition Word		
Punctuation	[64]	
Pronoun		
Emoticon	List of Emoticons [55] [64]	
Emotional Ontology	Affective Lexicon Ontology [60]	NEC Emotion Lexicon [29]

first respectively generates preference maps (i.e., token-level preference scores) for the pattern- and fact-based model with a heterogeneous dynamic GCN. Then, the preference maps are fed into the corresponding model along with P to help the model focus on its preferred information. Finally, the models' output features are fused to predict if P is real or fake. Besides the normal classification loss, we design two auxiliary losses as enhancements, whose goals are to minimize the similarity between the two maps and to minimize the classification loss when the input maps are exchanged and ground-truth labels are reversed, respectively. (see Section 4.2)

4.1 Preference Map Generation

Assuming that P has n tokens, a preference map is a score distribution of length n where the i-th score represents to what extent the i-th token is preferred by the corresponding fake news detection model. For the pattern- and the fact-based model, we respectively generate Pattern Preference Map and Fact Preference Map

$$\mathbf{m_P} = [\mathrm{m_P}_i]_{i=1}^{n}, \mathbf{m_F} = [\mathrm{m_F}_i]_{i=1}^{n}, \tag{1}$$

where all scores are in $[0, 1]$ and the sum of each map is 1.

4.1.1 Stylistic Tokens & Entities Recognition. As illustrated in Section 1, a pattern-based model focuses on common patterns (generally, writing styles) while a fact-based one focuses on verifiable objective claims. To guide the map generation, we exploit the prior knowledge with reference to the existing pattern- and fact-based works. Specifically, we recognize tokens that are likely to represent writing styles or key objective elements. To indicate patterns, we recognize a set of *stylistic tokens* $S = \{s_1, \ldots, s_{n_s}\}$ (e.g., emotional words, pronouns, punctuation marks) [64]; and to indicate facts, we extract the *entities* $E = \{e_1, \ldots, e_{n_e}\}$ because a verifiable claim generally contains at least one entity [46]. These indicating tokens are derived using pre-constructed dictionaries and public tools. In detail, to recognize stylistic tokens, we follow [64], which summarizes diverse emotion-related features and other useful linguistic features to represent textual patterns, and then generate a stylistic token table for each dataset. The types and references are shown in Table 1. A simple exact matching is performed to recognize the stylistic tokens in posts. To recognize the entities, we use two public tools: `LAC` [20][1] for Chinese and `TexSmart` [26, 63][2] for English. The tokens excluded by S and E are in a set $T = \{t_1, \ldots, t_{n_t}\}$ where $n_t = n - n_s - n_e$.

[1] https://github.com/baidu/lac/

[2] https://ai.tencent.com/ailab/nlp/texsmart/en/index.html. We use the v0.2.0 (Large).

4.1.2 heterogeneous Dynamic GCN. Although the stylistic tokens and entities recognized by general dictionaries or tools provide a good prior to what tokens might be preferred, directly using the recognition result for map generation is insufficient: First, the coverage is limited, leading the map to overlook some other preferred and useful tokens for detection models; Second, a token's preference score should be dynamically determined in its context (i.e., the post) rather than static rules. To enable the information of different types of nodes to dynamically and sufficiently interact with each other, we design a graph-based preference learner, Heterogeneous Dynamic Graph Convolutional Network (HetDGCN). As shown in Figure 2(b), we first construct a heterogeneous graph that contains multi-type nodes (tokens) with a learnable correlation matrix (i.e., adjacent matrix). Then, we leverage a heterogeneous graph convolution to enable message passing among different types of nodes. The final preference scores are obtained using the learned correlation matrix. The stages are as follows:

Graph Initialization. Recall that we have divided the tokens in P into three parts: stylistic tokens S, entities E, and others T. To preserve their different roles, we construct a heterogeneous graph G, where each node corresponds to a token in S, E, or T and the weight of each edge represents the correlation between the connected tokens. The node representation is initialized with the pre-trained language model (here, BERT [8]), denoted as $\mathbf{H}^{(0)} \in \mathbb{R}^{n \times d}$ where d is the dimensionality of each node vector. Note that this matrix is stacked with the representation of S, E, and T, i.e., $\mathbf{H}^{(0)} = [\mathbf{H}_S^{(0)}; \mathbf{H}_E^{(0)}; \mathbf{H}_T^{(0)}]$. The edge weights (correlations) are initialized with calculating the cosine similarity of token pairs [19] which is scaled to $[0, 1]$:

$$\mathbf{A}^{(0)}(i,j) = \frac{\mathbf{h}_i^{(0)} \cdot \mathbf{h}_j^{(0)}}{2\|\mathbf{h}_i^{(0)}\|\|\mathbf{h}_j^{(0)}\|} + 0.5, \tag{2}$$

where $\mathbf{h}_i^{(0)}$ and $\mathbf{h}_j^{(0)}$ are the initial node features, and $\mathbf{A}^{(0)}(i,j) \in [0, 1]$ is the initial weight of the edge connecting the i-th and the j-th node. Following [23], we define the normalized correlation matrix of the l-th layer $\hat{\mathbf{A}}^{(l)} = (\mathbf{D}^{(l)})^{-\frac{1}{2}}\mathbf{A}^{(l)}(\mathbf{D}^{(l)})^{-\frac{1}{2}}$. $\mathbf{D}^{(l)}$ is the degree matrix of the l-th layer where $\mathbf{D}^{(l)}(i,i) = \sum_j \mathbf{A}^{(l)}(i,j)$.

Graph Convolution & Correlation Update. Different types of nodes describe different aspects of the given text which we expect to distinguish for preference learning. Therefore, instead of using standard graph convolution for node interaction [23], we use a heterogeneous graph convolution [19], which separately handle the neighbors of different types and then aggregate the interacted features. Further, we use a dynamic correlation matrix which is updated each layer according to the present node similarity and expect the final correlations (edge weights) could reflect the bias of the nodes in the context. In detail, the feature matrix of $(l+1)$-th layer is calculated with

$$\mathbf{H}^{(l+1)} = \mathrm{ReLU}\left(\sum_{\tau \in \mathcal{T}} \hat{\mathbf{A}}_\tau^{(l)} \mathbf{H}_\tau^{(l)} \mathbf{W}_\tau^{(l)}\right), \tag{3}$$

where $\hat{\mathbf{A}}_\tau^{(l)}$ is the submatrix of the correlation matrix of the l-th layer $\hat{\mathbf{A}}^{(l)}$ whose rows contain all the nodes and columns record their correlation with nodes of the type $\tau \in \{S, E, T\}$. $\mathbf{W}_\tau^{(l)}$ is the

learnable weight matrix of the type τ in this layer. Then, the correlation matrix is updated using

$$\Delta\mathbf{A}^{(l+1)} = \sigma\left(\mathbf{H}^{(l+1)}\mathbf{W}_A^{(l+1)}\mathbf{H}^{(l+1)T}\right), \tag{4}$$

$$\mathbf{A}^{(l+1)} = \alpha\mathbf{A}^{(l)} + (1-\alpha)\Delta\mathbf{A}^{(l+1)}, \tag{5}$$

where $\mathbf{W}_A^{(l+1)}$ is the learnable weight matrix for updating correlations, σ denotes the sigmoid function and α is a trade-off factor in $[0, 1]$.

Preference-aware Readout. After the L-layer HetDGCN, we obtain the correlation matrix $\mathbf{A}^{(L)}$, on which we expect to estimate the preference levels to pattern- and fact-based models of each token. For the i-th node, the pattern preference score $\mathrm{m}_{\mathrm{P}i}$ is calculated by its correlation with any nodes except those representing entity tokens:

$$\mathrm{m}_{\mathrm{P}i} = \sum_{j=1}^{n}\mathbf{A}^{(L)}(i,j) - \sum_{k=1}^{n_e}\mathbf{A}_E^{(L)}(i,k). \tag{6}$$

Similarly, the fact preference score excludes the correlation with the stylistic nodes:

$$\mathrm{m}_{\mathrm{F}i} = \sum_{j=1}^{n}\mathbf{A}^{(L)}(i,j) - \sum_{k=1}^{n_s}\mathbf{A}_S^{(L)}(i,k). \tag{7}$$

Finally, the preference maps are obtained by normalized the correlation sums of each token:

$$\mathbf{m_P} = \left[\frac{\mathrm{m}_{\mathrm{P}i}}{\sum_j \mathrm{m}_{\mathrm{P}j}}\right]_{i=1}^{n}, \mathbf{m_F} = \left[\frac{\mathrm{m}_{\mathrm{F}i}}{\sum_j \mathrm{m}_{\mathrm{F}j}}\right]_{i=1}^{n}. \tag{8}$$

4.2 Preference-aware Joint Fake News Detection

As the fact-based and pattern-based models are diverse, we here use the typical pattern- and fact-based detection process to illustrate how to integrate the generated preference maps into them. Most specific models can be easily reformulated similarly to accommodate our framework.

4.2.1 Pattern-based Model. As shown in Figure 2(c), a typical pattern-based uses a textual feature extractor to obtain a vector for final prediction. Here, we use the Pattern Preference Map as attention weights to make the model attend to its preferred tokens in the post P. For example, if the extractor is a BERT [8] or an LSTM [18] whose output is $[\mathbf{p}_1; \ldots; \mathbf{p}_n]$, the aggregated vector is calculated as

$$\mathbf{p} = \sum_{i=1}^{n}\mathrm{m}_{\mathrm{P}i}\mathbf{p}_i. \tag{9}$$

Note that our preference map is at the token level, for the extractor that does not output n vectors such as TextCNN [22], the map might be used before the extractor, right after we obtain token embeddings from pre-trained models.

4.2.2 Fact-based Model. In a typical fact-based model, the post P are first used to retrieve from a fact-checking source to collect the related articles (or, evidence) D. Assuming n_f articles are returned, we represent the articles in D as $[\mathbf{d}_1; \ldots; \mathbf{d}_{n_f}]$. Then the post and evidence vectors are fed into an inference module, which is often designed to capture the complicated interactions such as coherence and conflicts between P and D (e.g., [27, 57]). The output vectors of inference module $\mathbf{f}$, which implicitly represent the relationship of the post-evidence pairs, is used for final prediction.

To avoid the interference of non-check-worthy parts (e.g., the publisher's remark), the Fact Preference Map guides the inference module by using the attention mechanism to aggregate the token vectors in P before post-evidence inference. The final vector is calculated as

$$\mathbf{q} = \sum_{i=1}^{n}\mathrm{m}_{\mathrm{F}i}\mathbf{q}_i, \tag{10}$$

$$\mathbf{f} = \mathrm{InferenceModule}(\mathbf{q}, [\mathbf{d}_1; \ldots; \mathbf{d}_{n_f}]), \tag{11}$$

where $\mathbf{q}_i$ is the representation of the i-th token in P for fact-based methods.

4.2.3 Joint Detection. For final prediction, we concatenate the output vectors of pattern- and fact-based models and feed it into a multi-layer Perceptron (MLP) and obtain the prediction $\hat{y}$:

$$\hat{y} = \mathrm{MLP}([\mathbf{p}; \mathbf{f}]). \tag{12}$$

4.2.4 Losses. During training, we use three losses to supervise 1) the prediction of binary (fake and real) classification; and 2) the differentiation of the two preference maps. For the first goal, we minimize the cross-entropy loss between the prediction $\hat{y}$ and the label y

$$\mathcal{L}_{cls}(y, \hat{y}) = \mathrm{CELoss}(y, \hat{y}), \tag{13}$$

where $\mathrm{CELoss}(y, p) = -y\log p - (1-y)\log(1-p)$. For the second goal, we consider the reciprocal roles of the two models and let them supervise *mutually*. In detail, we minimize the cosine similarity between the Pattern and the Fact Preference Map

$$\mathcal{L}_{cos} = \frac{\mathbf{m_P}\cdot\mathbf{m_F}}{\|\mathbf{m_P}\|\|\mathbf{m_F}\|}, \tag{14}$$

and the cross-entropy loss under the condition that the input maps for the two models are exchanged and the ground-truth label is reversed

$$\mathcal{L}_{cls}(y_{rev}, \hat{y}') = \mathrm{CELoss}(y_{rev}, \hat{y}'), \tag{15}$$

where $y_{rev} = |1-y|$ and the predictive value $\hat{y}' = \mathrm{MLP}([\mathbf{p}'; \mathbf{f}'])$. $\mathbf{p}'$ and $\mathbf{f}'$ are respectively the output of the pattern-based and the fact-based model with each other's preference map as input. When receiving non-preferred information, the models are expected to be misled and generate non-distinctive features. The total loss of a sample to minimize is

$$\mathcal{L} = \beta_1\mathcal{L}_{cls}(y, \hat{y}) + \beta_2\mathcal{L}_{cos} + \beta_3\mathcal{L}_{cls}(y_{rev}, \hat{y}'), \tag{16}$$

where β_1, β_2 and β_3 are trade-off factors in $[0, 1]$. We average the loss of samples in each mini-batch before backpropagation.

5 EXPERIMENTS

We conduct experiments on two datasets to answer the following evaluation questions:

EQ1: Can Pref-FEND improve the performance of fake news detection models with single preference?

EQ2: Can Pref-FEND improve the performance for fake news detection that is integrated by pattern- and fact-based models?

EQ3: How effective are the designed components of Pref-FEND?

EQ4: How different are the Fact and the Pattern Preference Map?

Table 2: Statistics of the Weibo and the Twitter dataset.

Number of	Weibo			Twitter		
	Train	Val	Test	Train	Val	Test
Fake News	1,896	632	633	3,419	1,140	1,140
Real News	1,920	640	641	5,406	1,802	1,802
Total	3,816	1,272	1,274	8,825	2,942	2,942
		(6,362)			(14,709)	
Relevant Articles		17,849			12,419	

5.1 Datasets

As no existing dataset of fake news detection provides social media posts and relevant articles (as the fact-checking source) simultaneously, we construct two datasets of different languages (Chinese and English) based on the existing data and external sources. The statistics are shown in Table 2. The details are as follows:

Weibo Dataset

Post. We utilize the Weibo-20 dataset [64] which contains 6,362 news posts and the ratio of fake and real news posts is roughly 1:1. We keep its original temporal split with a ratio of 6:2:2 for train, validation, and test set.

Relevant Articles. We collect fact-checking articles and other relevant articles to construct our fact-checking source. In detail, we use the fact-checking articles crawled by Sheng et al. [39][3] from multiple websites such as *Jiaozhen*[4], *Zhuoyaoji*[5], and *Baidu Piyao*[6]. Then, we crawl other relevant articles from Baidu News, with the keywords in the Weibo posts as queries. The keywords are extracted using `jieba`[7]. For each query, we obtain at most 30 items and attempt to download full articles using `Newspaper3k`[8]. Finally, the de-duplication of all accessible articles lead to an article base containing 17,849 articles.

Twitter Dataset

Post. We first combine two datasets for detecting previously fact-checked claims released by Shaar et al. [38] and Vo and Lee [47], respectively, as they not only provide tweets but also relevant articles from *Snopes*[9]. As our task is formulated as a binary classification task, we merge *true*, *mostly-true*, *correct-attribution* into *real*, and *false*, *mostly-false*, *misattributed*, and *legend* into *fake*. The other categories are dropped. As these two datasets are largely imbalanced (1,047 real and 8,992 fake), we utilize PHEME [24] dataset as a supplement, whose annotation files provide some referred news links. For PHEME, we merge *real* and *non-rumor* into *real* and obtain 5,090 real and 638 fake news posts. After pre-processing using `TexSmart` and dropping failure cases, we obtain 14,709 posts.

Relevant Articles. Because the Twitter dataset has fewer topics than the Weibo dataset, we start from the articles in these datasets to construct the relevant article base. First, we incorporate the fact-checking articles from the datasets released in [38] and [47], and referred news articles in the PHEME dataset (if accessible). Then, we use their titles (tokenized using NLTK [3]) as queries and search on Google News using `GNews`[10]. After post-processing, we obtain an article base containing 12,419 articles. Note that we do not use the existing DeClarE [34] and MultiFC [2] datasets which provide both claims (posts) and relevant articles (or webpages) because its claims are normalized and thus with weak patterns of social media posts. We split the train, validation, test set temporally with a ratio of 6:2:2.

5.2 Base Models

We use six representative text-based models as base models:

Pattern-based Models

- **Bi-LSTM** [15] is widely used in many existing works of our task for text encoding [16, 21, 37]. We implement a one-layer Bi-LSTM with a maximum sequence length of 100 and a hidden size of 128. We average all the hidden states as representations of posts which are further fed into an MLP for prediction.
- **EANN-Text** [52] is a model that tries to distract the fake news detection model from memorizing event-specific features. It uses TextCNN for text representation and adds an auxiliary task of event classification for adversarial learning using gradient reversal layer [12]. We re-implement the model according to the public code[11]. The complete EANN is a multi-modal model but we here use its text-only version. For TextCNN, the number of filters is 20 and the window sizes are $\{1, 2, 3, 4\}$. The labels for the auxiliary event classification task are derived by clustering the training set with K-means where $K = 300$.
- **BERT-Emo** [64] is a model that uses BERT to encode the text and captures the emotion that news publishers express. As we focus on the contents rather than social contexts, we adopt a simplified version where emotions in comments are not considered. We use the author-released code[12]. The maximum sequence length is 150 and the size of embedding vectors is 768.

Fact-based Models

- **DeClarE** [34] is a model which uses claim-specific attention to focus on salient words in relevant articles. We remove the source embedding which is unavailable in the datasets. We re-implement the model according to the third-party code[13]. The text encoder is a one-layer Bi-LSTM with the hidden size of 128.
- **EVIN** [57] is an evidence inference network, which captures the semantic conflicts between the post and relevant articles using the attention mechanism. We re-implement the model according to the paper as no public code is available. The hidden size of one-layer Bi-LSTM is 60. The maximum sequence length is 200.
- **MAC** [48] is a hierarchical multi-head attentive network that combines word- and article-level attention. We re-implement according to the author-released code[14]. We use one-layer Bi-LSTM networks with a hidden size of 300 to build MAC. Two multi-head attention modules have 5 and 2 heads, respectively.

Note that when base models are used as a module in Pref-FEND, we extract the last-layer feature before the MLP layer.

[3] https://github.com/ICTMCG/MTM/
[4] https://fact.qq.com
[5] http://piyao.sina.cn/
[6] https://author.baidu.com/home?app_id=15060. *Piyao* means "refuting false claims".
[7] https://github.com/fxsjy/jieba
[8] https://newspaper.readthedocs.io/
[9] https://www.snopes.com/
[10] https://github.com/ranahaani/GNews
[11] https://github.com/yaqingwang/EANN-KDD18
[12] https://github.com/RMSnow/WWW2021
[13] https://github.com/atulkumarin/DeClare/
[14] https://github.com/nguyenvo09/EACL2021/

Table 3: Performance comparison with pattern- or fact-based models. *w/* Pref-FEND$_S$ means the model is incorporated as a module of Pref-FEND$_S$ framework.

Method	Weibo								Twitter							
	Acc.	macF1	P_{fake}	R_{fake}	$F1_{fake}$	P_{real}	R_{real}	$F1_{real}$	Acc.	macF1	P_{fake}	R_{fake}	$F1_{fake}$	P_{real}	R_{real}	$F1_{real}$
Pattern-based																
Bi-LSTM	0.667	0.660	0.626	0.820	0.710	0.744	0.516	0.610	0.767	0.732	0.753	0.923	0.829	0.811	0.522	0.635
w/ Pref-FEND$_S$	**0.709**	**0.709**	0.696	0.735	0.715	0.723	0.683	0.702	**0.793**	**0.788**	0.870	0.779	0.822	0.700	0.816	0.754
EANN-Text	0.692	0.690	0.860	0.785	0.717	0.739	0.601	0.663	0.770	0.725	0.742	0.960	0.837	0.881	0.472	0.614
w/ Pref-FEND$_S$	**0.740**	**0.740**	0.760	0.697	0.727	0.723	0.783	0.752	**0.798**	**0.788**	0.837	0.832	0.834	0.737	0.744	0.741
BERT-Emo	0.712	0.708	0.667	0.839	0.743	0.787	0.587	0.672	0.794	0.762	0.769	0.950	0.850	0.873	0.550	0.675
w/ Pref-FEND$_S$	**0.746**	**0.744**	0.703	0.847	0.768	0.811	0.647	0.720	**0.804**	**0.776**	0.781	0.945	0.855	0.870	0.582	0.697
Fact-based																
DeClarE	0.684	0.678	0.642	0.820	0.720	0.755	0.549	0.636	0.786	0.753	0.765	0.941	0.844	0.853	0.543	0.663
w/ Pref-FEND$_S$	**0.706**	**0.701**	0.661	0.840	0.740	0.785	0.574	0.663	**0.798**	**0.785**	0.823	0.854	0.838	0.754	0.710	0.731
EVIN	0.707	0.706	0.683	0.768	0.690	0.738	0.647	0.690	0.783	0.761	0.788	0.884	0.833	0.773	0.623	0.690
w/ Pref-FEND$_S$	**0.712**	**0.711**	0.682	0.787	0.731	0.752	0.638	0.690	**0.795**	**0.774**	0.794	0.899	0.843	0.797	0.631	0.705
MAC	0.724	0.723	0.695	0.793	0.741	0.763	0.657	0.706	0.791	0.764	0.777	0.924	0.844	0.829	0.581	0.683
w/ Pref-FEND$_S$	**0.749**	**0.748**	0.728	0.790	0.758	0.773	0.708	0.739	**0.804**	**0.784**	0.800	0.907	0.850	0.814	0.642	0.718

Table 4: Performance comparison with integrated (pattern-and-fact-based) models.

Method	Weibo								Twitter							
	Acc.	macF1	P_{fake}	R_{fake}	$F1_{fake}$	P_{real}	R_{real}	$F1_{real}$	Acc.	macF1	P_{fake}	R_{fake}	$F1_{fake}$	P_{real}	R_{real}	$F1_{real}$
Bi-LSTM (Pattern-based) + DeClarE (Fact-based)																
Last-layer Fusion	0.697	0.696	0.721	0.637	0.676	0.678	0.757	0.715	0.798	0.768	0.775	0.945	0.851	0.866	0.566	0.685
Logits Average	0.692	0.685	0.646	0.840	0.730	0.776	0.544	0.640	0.784	0.750	0.762	0.943	0.843	0.855	0.534	0.657
Pref-FEND	**0.714**	**0.712**	0.684	0.788	0.732	0.754	0.640	0.692	**0.812**	**0.792**	0.803	0.917	0.857	0.832	0.645	0.727
BERT-Emo (Pattern-based) + MAC (Fact-based)																
Last-layer Fusion	0.735	0.731	0.683	0.874	0.766	0.828	0.599	0.695	0.804	0.798	0.871	0.798	0.833	0.718	0.813	0.763
Logits Average	0.736	0.734	0.693	0.842	0.760	0.802	0.632	0.707	0.778	0.741	0.754	0.946	0.839	0.857	0.514	0.642
Pref-FEND	**0.756**	**0.754**	0.714	0.848	0.775	0.816	0.665	0.733	**0.814**	**0.801**	0.829	0.877	0.853	0.786	0.715	0.749

5.3 Experimental Setup

Evaluation Metrics. We report accuracy (Acc.) and macro F1 score (macF1). For each class, we also report precision, recall, and F1 score, denoted as P_{cls}, R_{cls}, and $F1_{cls}$ where $cls = \{fake, real\}$.

Implementation Details. In Pref-FEND, the number of layers in HetDGCN L is 2. We perform grid search in a small interval and finally let $\alpha = 0.5$, $\beta_1 = 2$, $\beta_2 = 1$, and $\beta_3 = 1$. For all base models and our Pref-FEND, the initial token embeddings are obtained from pre-trained models in HuggingFace's `Transformers`[56] (specifically, `bert-base-chinese` and `bert-base-uncased`). For all fact-based models, the top 5 retrieved articles are considered. Other hyper-parameters have been described in Section 5.2. The methods are implemented with PyTorch [33] and Pytorch Geometric [10].

5.4 Performance Comparison (EQ1 & EQ2)

5.4.1 Comparing with Pattern- and Fact-based Methods. To fairly compare with existing single-preference (i.e., pattern- or fact-based) models, we reduce our framework to a single-model version named **Pref-FEND$_S$**. In detail, when comparing with a pattern-based model, we remove the fact-based model but preserve the Fact Preference Map for training; and vice versa. From the results in Table 3, we have the following observations:

First, Pref-FEND$_S$ successfully improves the performance of all the pattern-based and fact-based models on the two datasets. This verifies our observation that the original base models might be distracted from non-preferred information, which thus limits their generalizability to unseen samples. With the help of Pref-FEND$_S$, the base models are more focused during training.

Second, BERT-Emo outperforms Bi-LSTM and EANN-Text. This is as expected because BERT can generate expressive representations and the additional emotion-related features are proved helpful for this task. With the guidance of Pref-FEND$_S$, it gains a boost of 3.6 percent points in macro F1 scores on Weibo and a boost of 1.4 percent points on Twitter. This reveals the importance of preference modeling for alleviating the overfitting of specific features.

Third, MAC outperforms DeClarE and EVIN, though they are all based on the attention mechanism. This might be because it effectively uses multi-head attention to capture multi-aspect information. However, some heads might be distracted from the event description in the post, which can be alleviated by our framework.

5.4.2 Comparing with Integrated (Pattern-and-fact-based) Methods. We implement the following methods which fuse the information from pattern- and fact-based models:

- **Last-layer Fusion** which uses the post as input and concatenates the last-layer features of two models for final prediction;
- **Logits Average** which averages the models' logits (which are in $[0, 1]$) for final prediction.

Table 5: Ablation study of Pref-FEND.

Method	Weibo								Twitter							
	Acc.	macF1	P_{fake}	R_{fake}	$F1_{fake}$	P_{real}	R_{real}	$F1_{real}$	Acc.	macF1	P_{fake}	R_{fake}	$F1_{fake}$	P_{real}	R_{real}	$F1_{real}$
Bi-LSTM (Pattern-based) + DeClarE (Fact-based)																
Pref-FEND	**0.714**	**0.712**	0.684	0.788	0.732	0.754	0.640	0.692	**0.812**	**0.792**	0.803	0.917	0.857	0.832	0.645	0.727
w/ rand init maps	0.694	0.693	0.676	0.736	0.705	0.715	0.652	0.682	0.788	0.765	0.787	0.896	0.838	0.790	0.616	0.692
w/o $\mathcal{L}_{cos}$	0.701	0.703	0.672	0.787	0.725	0.747	0.621	0.678	0.794	0.785	0.845	0.813	0.829	0.721	0.764	0.742
w/o $\mathcal{L}_{cls}(y_{rev}, \hat{y}\prime)$	0.703	0.702	0.710	0.679	0.694	0.696	0.725	0.710	0.792	0.764	0.775	0.932	0.846	0.842	0.571	0.681
w/ only $\mathcal{L}_{cls}(y, \hat{y})$	0.700	0.702	0.672	0.782	0.723	0.743	0.622	0.677	0.789	0.747	0.752	0.979	0.851	0.936	0.490	0.643
BERT-Emo (Pattern-based) + MAC (Fact-based)																
Pref-FEND	**0.756**	**0.754**	0.714	0.848	0.775	0.816	0.665	0.733	**0.814**	**0.801**	0.829	0.877	0.853	0.786	0.715	0.749
w/ rand init maps	0.723	0.716	0.666	0.886	0.761	0.833	0.562	0.671	0.806	0.786	0.801	0.911	0.852	0.820	0.642	0.720
w/o $\mathcal{L}_{cos}$	0.747	0.745	0.706	0.842	0.768	0.807	0.654	0.722	0.807	0.801	0.874	0.799	0.835	0.721	0.819	0.767
w/o $\mathcal{L}_{cls}(y_{rev}, \hat{y}\prime)$	0.745	0.740	0.690	0.883	0.775	0.841	0.608	0.706	0.808	0.789	0.806	0.903	0.852	0.811	0.657	0.726
w/ only $\mathcal{L}_{cls}(y, \hat{y})$	0.741	0.735	0.682	0.896	0.775	0.851	0.588	0.696	0.792	0.787	0.869	0.778	0.821	0.699	0.815	0.752

Table 6: Frequently preferred tokens (separated by "|") in the pattern-preferred and the fact-preferred token set.

Set	Category	Token (Translated into English)
Pattern-preferred	Punctuation	, | . | ! | : | ? | [| " | (| . . . |] | @ | < |) | # | ~ | > | ; | /
	Negation	not | no | don't
	Pronoun	we | they | you all
	Others	find | such | think | may | certainly | so | release | some | as | careful | become | focus | loving heart | but | kind of
Fact-preferred	Evidence-related	claim | video | link | webpage | full text | say | picture | investigation | according to | uncover
	Entity-related	China | Beijing | police | place | car | Shanghai | official
	Pronoun	he | its | it | you
	Others	's | done | just | also | and | already | will | wait | go to | do | female | too | want | what | certain | die | pass | death | when | in | second | more | make | suffer | night | society

We implement these fusion methods and Pref-FEND with two groups of base models, Bi-LSTM+DeClarE and BERT-Emo+MAC. The results are shown in Table 4. Our observations are as follows:

First, Pref-FEND outperforms the two pattern-and-fact-based methods, which validates its effectiveness for integrating pattern- and fact-based models.

Second, comparing with the results in Table 3, Pref-FEND brings further improvements based on the remarkable performance of Pref-FEND$_S$ w.r.t the same base models. For example, on the Weibo dataset, Pref-FEND with Bi-LSTM and DeClarE gains another increase of macro F1 by 0.3 percent points than Pref-FEND$_S$ with Bi-LSTM and 1.1 percent points than Pref-FEND$_S$ with DeClarE. This proves that our framework is applicable to both the single-preference models and the integrated models based on them.

Third, the last-layer fusion does not necessarily perform better than the simple logits average. This indicates that last-layer fusion may be insufficient to align the feature spaces of the pattern- and the fact-based model, which leads to negative fusion effects.

5.5 Ablation Study (EQ3)

We study the effectiveness of our designed components and strategies based on the Pref-FEND models. The results are shown in Table 4.

5.5.1 Effectiveness of Model Preference Learning. Instead of recognizing the entities and stylistic tokens according to the prior knowledge, we randomly initialize preference maps (named as Pref-FEND *w/ rand init maps*). That forces the generation of preference maps to rely only on the supervision of ground-truth labels. The results show that although Pref-FEND *w/ rand init maps* is superior or comparable to the base models on both of the two datasets in terms of accuracy and macro F1, it falls behind the complete Pref-FEND. This proves the effectiveness of our model preference learning, which exploits prior knowledge in a dynamic graph representation learning process.

5.5.2 Effective of Losses for Differentiating the Preference Maps. We remove one of the two losses which aim at differentiating the two preference maps, or both. The variants are with the suffixes *w/o* $\mathcal{L}_{cos}$, *w/o* $\mathcal{L}_{cls}(y_{rev}, \hat{y}')$, and *w/ only* $\mathcal{L}_{cls}(y, \hat{y})$, respectively. We see that removing these losses brings performance drops w.r.t. accuracy. The largest drop occurs when removing both the two losses. This indicates that the auxiliary losses are effective and necessary to generate better preference maps for integration of models with different preferences.

5.6 Preference Map Analysis (EQ4)

5.6.1 Analysis on Most Frequent Token Set. To explore how different the Fact and the Pattern Preference Map are, we analyze the frequently preferred tokens in the Maps. For each post in the validation and test sets of Weibo, we first divide the tokens into a pattern group and a fact group, which indicates this token is scored higher in the Pattern or the Fact Preference Map. Then we extract the top 10 tokens in each group of all the posts and construct two token sets for frequency analysis. The frequent tokens in each set

Table 7: Three examples of fake news posts. Red represents pattern-preferred tokens and blue represents fact-preferred tokens. Darker color indicates a higher preference score.

#	Post (Translated into English)
1	A group of city administration officials in Sishui , Shandong , chased an old man until all his eggs were broken on the ground . The old man sat there helplessly . The officials ran away after hitting . The white-haired man should be about 80 years old , and he can't make much money by selling eggs . So why be aggressive ? Is there no moment for the officials to be alone ? If the officials only oppresses citizens , what's the good of having these officials ? You will be punished sooner or later for bullying the underprivileged . **Ground Truth: Fake** **Judgment:** Bi-LSTM (Fake), DeClarE (Real), Pref-FEND (Fake)
2	[A student of ZJU jumping to the West Lake for a crazy graduation photo drowned] On June 29 , Xin (not his real name) from ZJU and his classmates went to the waters near the scenic spot of " Konggu Chuanyin " in Gushan , Beili Lake , West Lake in Hangzhou . Xin asked his classmates to take pictures of his swimming underwater . He jumped into the West Lake from the side of Xiling brige on Beishan Road and swam to the lotus pool of Gushan park on the other side . He drowned when swimming to the center of the lake . Recently , he has received a full PhD scholarship from a U.S. university . **Ground Truth: Fake** **Judgment:** Bi-LSTM (Real), DeClarE (Fake), Pref-FEND (Fake)
3	Is anyone in Shanghai interested in raising a dog ? No Charge . Golden Retriever , Poodle , Samoyed , and other more breeds . There are dog-killing slaughterhouses being destroyed . If no one adopts , they will be euthanized . Let these little cute lives accompany with you . If you are really not able to raise them , please forward this message . **Ground Truth: Fake** **Judgment:** Bi-LSTM (Real), DeClarE (Real), Pref-FEND (Fake)

are shown with fine-grained categories in Table 6. We have the following observations:

First, in a pattern-preferred token set, punctuation marks and negation words are important as they express the publishers' tones and emotions. The other frequent tokens are closely related to self-expression, like "think", "may", and "kind of".

Second, in the fact-preferred set, evidence-related tokens that indicate materials and actions (e.g., "video", "webpage", "picture", "claim", and "uncover") and entity-related tokens (places, positions, etc.) are more focused. Some of the other words do not directly describe an event, but are often around the elements of a news event (e.g., 5W in journalism [54]), such as "already" and "when".

Third, the preferences of pronouns of the Pattern and the Fact Preference Map are different. Plural personal pronouns ("we", "they", and "you all") are frequently focused by pattern-based models, while single ones ("he", "it", and "you") are preferred by fact-based models. The reason might be that a post with significant fake news patterns often discusses some groups or inspires the audience to take action, while a post with an event description is generally related to specific persons or things.

Our analysis reveals that the learned preference maps are highly correlated to the ideal model preferences and thus effective for the guidance of models' focuses.

5.6.2 Case study. In Table 7, we show three fake news posts that are successfully judged by Pref-FEND with Bi-LSTM and DeClarE. Case 1 conveys strong signals of emotional patterns, which are preferred by pattern-based models, such as "helplessly", and "aggressive". Case 2 contains a large number of places and event descriptions, which is friendly to utilize the evidential texts in relevant articles. Due to the different dominant signals, the pattern-based Bi-LSTM judges correctly in Case 1, but fails in Case 2. And the judgments of the fact-based DeClarE are the opposite. However, in Case 3, both of them wrongly judge this post as real. Based on the observation, a pattern-based model can attend to the emotion trigger tokens like "cute" and "really', while a fact-based model can use the place ("Shanghai") and the dog breed ("Golden Retriever") to find evidence. Generally, it is unlikely that the two models both fail. We speculate that the failure is led by the negative interference from the non-preferred information. With the help of model preference learning, our Pref-FEND, however, succeed in judging all three posts as fake. These cases demonstrate the necessity of model preference learning and the effectiveness of Pref-FEND.

6 CONCLUSION AND FUTURE WORK

We propose the framework Pref-FEND to integrate the pattern-based and fact-based fake news detection models in a preference-aware fashion. The learned preference maps guide the models to focus more on their preferred parts with less interference by the non-preferred parts. Experiments on the two newly constructed datasets show that Pref-FEND outperforms the existing detection models. Further analysis demonstrates that preference learning helps models of different preferences more focused and thus makes both the single-preference and the integrated models better-performing.

How to enhance the interaction between the preference map generation and specific models and how to extend the framework to multi-class and multi-preference scenarios are expected to be explored in the future. The acquisition and exploitation of prior knowledge in this task are also worth studying further to improve overall performance.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions. This work was supported by the National Key Research and Development Program of China (2017YFC0820604), and the National Natural Science Foundation of China (U1703261).

REFERENCES

[1] Oluwaseun Ajao, Deepayan Bhowmik, and Shahrzad Zargari. 2019. Sentiment aware fake news detection on online social networks. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2507–2511. https://doi.org/10.1109/ICASSP.2019.8683170

[2] Isabelle Augenstein, Christina Lioma, Dongsheng Wang, Lucas Chaves Lima, Casper Hansen, Christian Hansen, and Jakob Grue Simonsen. 2019. MultiFC: A Real-World Multi-Domain Dataset for Evidence-Based Fact Checking of Claims. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (Hong Kong, China). Association for Computational Linguistics, 4685–4697. https://doi.org/10.18653/v1/D19-1475

[3] Steven Bird. 2006. NLTK: The Natural Language Toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions* (Sydney, Australia). Association for Computational Linguistics, 69–72. https://doi.org/10.3115/1225403.1225421

[4] Juan Cao, Junbo Guo, Xirong Li, Zhiwei Jin, Han Guo, and Jintao Li. 2018. Automatic rumor detection on microblogs: A survey. *arXiv* arXiv:1807.03505 (2018). https://arxiv.org/abs/1807.03505 version 1.

[5] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information Credibility on Twitter. In *Proceedings of the 20th International Conference on World Wide Web* (Hyderabad, India) *(WWW '11)*. Association for Computing Machinery, New York, NY, USA, 675–684. https://doi.org/10.1145/1963405.1963500

[6] Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, and Niloy Ganguly. 2016. Stop Clickbait: Detecting and Preventing Clickbaits in Online News Media. In *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (Davis, California) *(ASONAM '16)*. IEEE Press, 9–16.

[7] Limeng Cui, Haeseung Seo, Maryam Tabar, Fenglong Ma, Suhang Wang, and Dongwon Lee. 2020. DETERRENT: Knowledge Guided Graph Attention Network for Detecting Healthcare Misinformation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Virtual Event, CA, USA). Association for Computing Machinery, 492–502. https://doi.org/10.1145/3394486.3403092

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (Minneapolis, Minnesota). Association for Computational Linguistics, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[9] Zhendong Dong and Qiang Dong. 2003. HowNet: A hybrid language and knowledge resource. In *Proceedings of 2003 International Conference on Natural Language Processing and Knowledge Engineering*. IEEE, 820–824. https://doi.org/10.1109/NLPKE.2003.1276017

[10] Matthias Fey and Jan Eric Lenssen. 2019. Fast Graph Representation Learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*. http://arxiv.org/abs/1903.02428

[11] Marc Fisher, John Woodrow Cox, and Peter Hermann. 2016. Pizzagate: From rumor, to hashtag, to gunfire in DC. *The Washington Post* (2016). https://www.washingtonpost.com/local/pizzagate-from-rumor-to-hashtag-to-gunfire-in-dc/2016/12/06/4c7def50-bbd4-11e6-94ac-3d324840106c_story.html

[12] Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised Domain Adaptation by Backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning* (Lille, France) *(Proceedings of Machine Learning Research, Vol. 37)*, Francis Bach and David Blei (Eds.). PMLR, 1180–1189. http://proceedings.mlr.press/v37/ganin15.html

[13] Bilal Ghanem, Simone Paolo Ponzetto, Paolo Rosso, and Francisco Rangel. 2021. FakeFlow: Fake News Detection by Modeling the Flow of Affective Information. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* (Online). Association for Computational Linguistics, 679–689. https://www.aclweb.org/anthology/2021.eacl-main.56

[14] Anastasia Giachanou, Paolo Rosso, and Fabio Crestani. 2019. Leveraging Emotional Signals for Credibility Detection. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, 877–880. https://doi.org/10.1145/3331184.3331285

[15] Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural networks* 18, 5-6 (2005), 602–610. https://doi.org/10.1016/j.neunet.2005.06.042

[16] Han Guo, Juan Cao, Yazi Zhang, Junbo Guo, and Jintao Li. 2018. Rumor detection with hierarchical social attention network. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (Torino, Italy). Association for Computing Machinery, 943–951. https://doi.org/10.1145/3269206.3271709

[17] Zhijiang Guo, Yan Zhang, and Wei Lu. 2019. Attention Guided Graph Convolutional Networks for Relation Extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (Florence, Italy). Association for Computational Linguistics, 241–251. https://doi.org/10.18653/v1/P19-1024

[18] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[19] Linmei Hu, Tianchi Yang, Chuan Shi, Houye Ji, and Xiaoli Li. 2019. Heterogeneous Graph Attention Networks for Semi-supervised Short Text Classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (Hong Kong, China). Association for Computational Linguistics, 4821–4830. https://doi.org/10.18653/v1/D19-1488

[20] Zhenyu Jiao, Shuqi Sun, and Ke Sun. 2018. Chinese Lexical Analysis with Deep Bi-GRU-CRF Network. *arXiv* arXiv:1807.01882 (2018). https://arxiv.org/abs/1807.01882 version 1.

[21] Hamid Karimi and Jiliang Tang. 2019. Learning Hierarchical Discourse-level Structure for Fake News Detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (Minneapolis, Minnesota). Association for Computational Linguistics, 3432–3442. https://doi.org/10.18653/v1/N19-1347

[22] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Doha, Qatar). Association for Computational Linguistics, 1746–1751. https://doi.org/10.3115/v1/D14-1181

[23] Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *Proceedings of the fifth International Conference on Learning Representations*. https://openreview.net/forum?id=SJU4ayYgl

[24] Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. All-in-one: Multi-task Learning for Rumour Verification. In *Proceedings of the 27th International Conference on Computational Linguistics* (Santa Fe, New Mexico, USA). Association for Computational Linguistics, 3402–3413. https://www.aclweb.org/anthology/C18-1288

[25] Shimon Kogan, Tobias J Moskowitz, and Marina Niessner. 2020. Fake news: Evidence from financial markets. *Available at SSRN 3237763* (2020). https://doi.org/10.2139/ssrn.3237763

[26] Lemao Liu, Haisong Zhang, Haiyun Jiang, Yangming Li, Enbo Zhao, Kun Xu, Linfeng Song, Suncong Zheng, Botong Zhou, Dick Zhu, Xiao Feng, Tao Chen, Tao Yang, Dong Yu, Feng Zhang, ZhanHui Kang, and Shuming Shi. 2021. TexSmart: A System for Enhanced Natural Language Understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations* (Online). Association for Computational Linguistics, 1–10. https://doi.org/10.18653/v1/2021.acl-demo.1

[27] Jing Ma, Wei Gao, Shafiq Joty, and Kam-Fai Wong. 2019. Sentence-Level Evidence Embedding for Claim Verification with Hierarchical Attention Networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (Florence, Italy). Association for Computational Linguistics, 2561–2571. https://doi.org/10.18653/v1/P19-1244

[28] Rahul Mishra and Vinay Setty. 2019. SADHAN: Hierarchical Attention Networks to Learn Latent Aspect Embeddings for Fake News Detection. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval* (Santa Clara, CA, USA) *(ICTIR '19)*. Association for Computing Machinery, New York, NY, USA, 197–204. https://doi.org/10.1145/3341981.3344229

[29] Saif M Mohammad and Peter D Turney. 2013. Crowdsourcing a word–emotion association lexicon. *Computational intelligence* 29, 3 (2013), 436–465. https://doi.org/10.1111/j.1467-8640.2012.00460.x

[30] Salman Bin Naeem and Rubina Bhatti. 2020. The Covid-19 'infodemic': a new front for information professionals. *Health Information & Libraries Journal* 37, 3 (2020), 233–239. https://doi.org/10.1111/hir.12311

[31] Qiong Nan, Juan Cao, Yongchun Zhu, Yanyan Wang, and Jintao Li. 2021. MDFEND: Multi-domain Fake News Detection. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management* (Virtual Event, QLD, Australia). Association for Computing Machinery. https://doi.org/10.1145/3459637.3482139

[32] Van-Hoang Nguyen, Kazunari Sugiyama, Preslav Nakov, and Min-Yen Kan. 2020. FANG: Leveraging Social Context for Fake News Detection Using Graph Representation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (Virtual Event, Ireland). Association for Computing Machinery, 1165–1174. https://doi.org/10.1145/3340531.3412046

[33] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems* (Vancouver, Canada), H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.), Vol. 32. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf

[34] Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. DeClarE: Debunking Fake News and False Claims using Evidence-Aware Deep Learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (Brussels, Belgium). Association for Computational

Linguistics, 22–32. https://doi.org/10.18653/v1/D18-1003

[35] Piotr Przybyla. 2020. Capturing the Style of Fake News. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. AAAI Press, 490–497. https://doi.org/10.1609/aaai.v34i01.5386

[36] Zamira Rahim. 2019. Bangladesh mobs lynch eight people over child abduction rumours. Retrieved April 30, 2021 from https://www.independent.co.uk/news/world/asia/bangladesh-lynchings-mobs-child-abduction-rumours-dhaka-a9020031.html

[37] Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of Varying Shades: Analyzing Language in Fake News and Political Fact-Checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (Copenhagen, Denmark). Association for Computational Linguistics, 2931–2937. https://doi.org/10.18653/v1/D17-1317

[38] Shaden Shaar, Nikolay Babulkov, Giovanni Da San Martino, and Preslav Nakov. 2020. That is a Known Lie: Detecting Previously Fact-Checked Claims. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Online). Association for Computational Linguistics, 3607–3618. https://doi.org/10.18653/v1/2020.acl-main.332

[39] Qiang Sheng, Juan Cao, Xueyao Zhang, Xirong Li, and Lei Zhong. 2021. Article Reranking by Memory-Enhanced Key Sentence Matching for Detecting Previously Fact-Checked Claims. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (Online). Association for Computational Linguistics, 5468–5481. https://doi.org/10.18653/v1/2021.acl-long.425

[40] Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. dEFEND: Explainable Fake News Detection. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Anchorage, AK, USA). Association for Computing Machinery, 395–405. https://doi.org/10.1145/3292500.3330935

[41] Kai Shu, Deepak Mahudeswaran, Suhang Wang, and Huan Liu. 2020. Hierarchical propagation networks for fake news detection: Investigation and exploitation. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 14. AAAI Press, 626–637. https://ojs.aaai.org/index.php/ICWSM/article/view/7329

[42] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake News Detection on Social Media: A Data Mining Perspective. *SIGKDD Explorations Newsletter* 19 (2017), 22–36. https://doi.org/10.1145/3137597.3137600

[43] Kai Shu, Xinyi Zhou, Suhang Wang, Reza Zafarani, and Huan Liu. 2019. The Role of User Profiles for Fake News Detection. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (Vancouver, British Columbia, Canada). Association for Computing Machinery, 436–439. https://doi.org/10.1145/3341161.3342927

[44] Amila Silva, Yi Han, Ling Luo, Shanika Karunasekera, and Christopher Leckie. 2021. Propagation2Vec: Embedding partial propagation networks for explainable fake news early detection. *Information Processing & Management* 58, 5 (2021), 102618. https://doi.org/10.1016/j.ipm.2021.102618

[45] Amila Silva, Ling Luo, Shanika Karunasekera, and Christopher Leckie. 2021. Embracing Domain Differences in Fake News: Cross-domain Fake News Detection using Multi-modal Data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 557–565. https://ojs.aaai.org/index.php/AAAI/article/view/16134

[46] James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018. The Fact Extraction and VERification (FEVER) Shared Task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)* (Brussels, Belgium). Association for Computational Linguistics, 1–9. https://doi.org/10.18653/v1/W18-5501

[47] Nguyen Vo and Kyumin Lee. 2020. Where Are the Facts? Searching for Fact-checked Information to Alleviate the Spread of Fake News. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Online). Association for Computational Linguistics, 7717–7731. https://doi.org/10.18653/v1/2020.emnlp-main.621

[48] Nguyen Vo and Kyumin Lee. 2021. Hierarchical Multi-head Attentive Network for Evidence-aware Fake News Detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* (Online). Association for Computational Linguistics, 965–975. https://www.aclweb.org/anthology/2021.eacl-main.83

[49] Svitlana Volkova, Kyle Shaffer, Jin Yea Jang, and Nathan Hodas. 2017. Separating Facts from Fiction: Linguistic Models to Classify Suspicious and Trusted News Posts on Twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (Vancouver, Canada). Association for Computational Linguistics, 647–653. https://doi.org/10.18653/v1/P17-2102

[50] David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or Fiction: Verifying Scientific Claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Online). Association for Computational Linguistics, 7534–7550. https://doi.org/10.18653/v1/2020.emnlp-main.609

[51] Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational Graph Attention Network for Aspect-based Sentiment Analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Online). Association for Computational Linguistics, 3229–3238. https://doi.org/10.18653/v1/2020.acl-main.295

[52] Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. 2018. EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (London, United Kingdom). Association for Computing Machinery, 849–857. https://doi.org/10.1145/3219819.3219903

[53] Wikipedia. 2020. *Misinformation related to the COVID-19 pandemic*. Retrieved October 19, 2020 from https://en.wikipedia.org/wiki/Misinformation_related_to_the_COVID-19_pandemic

[54] Wikipedia. 2021. *Five Ws*. Retrieved May 26, 2021 from https://en.wikipedia.org/wiki/Five_Ws

[55] Wikipedia. 2021. *List of emoticons*. Retrieved May 15, 2021 from https://en.wikipedia.org/wiki/List_of_emoticons

[56] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (Online). Association for Computational Linguistics, 38–45. https://doi.org/10.18653/v1/2020.emnlp-demos.6

[57] Lianwei Wu, Yuan Rao, Ling Sun, and Wangbo He. 2021. Evidence Inference Networks for Interpretable Claim Verification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. AAAI Press, 14058–14066. https://ojs.aaai.org/index.php/AAAI/article/view/17655

[58] Lianwei Wu, Yuan Rao, Xiong Yang, Wanzhen Wang, and Ambreen Nazir. 2020. Evidence-Aware Hierarchical Interactive Attention Networks for Explainable Claim Verification. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20* (Yokohama, Japan). International Joint Conferences on Artificial Intelligence Organization, 1388–1394. https://doi.org/10.24963/ijcai.2020/193

[59] Lianwei Wu, Yuan Rao, Cong Zhang, Yongqiang Zhao, and Ambreen Nazir. 2021. Category-controlled Encoder-Decoder for Fake News Detection. *IEEE Transactions on Knowledge and Data Engineering* (2021). https://doi.org/10.1109/TKDE.2021.3103833

[60] Linhong Xu, Hongfei Lin, Yu Pan, Hui Ren, and Jianmei Chen. 2008. Constructing the Affective Lexicon Ontology. *Journal of the China society for scientific and technical information* 27, 2 (2008), 180–185. https://doi.org/10.3969/j.issn.1000-0135.2008.02.004

[61] Jin Ye, Junjun He, Xiaojiang Peng, Wenhao Wu, and Yu Qiao. 2020. Attention-driven dynamic graph convolutional network for multi-label image recognition. In *European Conference on Computer Vision*. Springer, 649–665.

[62] Huaiwen Zhang, Quan Fang, Shengsheng Qian, and Changsheng Xu. 2019. Multi-Modal Knowledge-Aware Event Memory Network for Social Media Rumor Detection. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France). Association for Computing Machinery, 1942–1951. https://doi.org/10.1145/3343031.3350850

[63] Haisong Zhang, Lemao Liu, Haiyun Jiang, Yangming Li, Enbo Zhao, Kun Xu, Linfeng Song, Suncong Zheng, Botong Zhou, Jianchen Zhu, Xiao Feng, Tao Chen, Tao Yang, Dong Yu, Feng Zhang, Zhanhui Kang, and Shuming Shi. 2020. TexSmart: A Text Understanding System for Fine-Grained NER and Enhanced Semantic Analysis. *arXiv* arXiv:2012.15639 (2020). https://arxiv.org/abs/2012.15639 version 1.

[64] Xueyao Zhang, Juan Cao, Xirong Li, Qiang Sheng, Lei Zhong, and Kai Shu. 2021. Mining Dual Emotion for Fake News Detection. In *Proceedings of the Web Conference 2021* (Ljubljana, Slovenia) *(WWW '21)*. Association for Computing Machinery, New York, NY, USA, 3465–3476. https://doi.org/10.1145/3442381.3450004

[65] Xinyi Zhou and Reza Zafarani. 2019. Network-Based Fake News Detection: A Pattern-Driven Approach. *SIGKDD Explorations Newsletter* 21, 2 (Nov. 2019), 48–60. https://doi.org/10.1145/3373464.3373473