Divide-and-Conquer: Post-User Interaction Network for Fake News Detection on Social Media

Erxue Min*
erxue.min@manchester.ac.uk
National Centre for Text Mining, Department of Computer Science, The University of Manchester
United Kingdom

Yu Rong†
yu.rong@hotmail.com
Tencent AI Lab
China

Yatao Bian
yatao.bian@gmail.com
Tencent AI Lab
China

Tingyang Xu
Peilin Zhao
tingyangxu@tencent.com
masonzhao@tencent.com
Tencent AI Lab
China

Junzhou Huang
jzhuang@uta.edu
Department of Computer Science and Engineering, University of Texas at Arlington
United States

Sophia Ananiadou
Sophia.Ananiadou@manchester.ac.uk
National Centre for Text Mining, Department of Computer Science, The University of Manchester
United Kingdom

ABSTRACT

Fake News detection has attracted much attention in recent years. Social context based detection methods attempt to model the spreading patterns of fake news by utilizing the collective wisdom from users on social media. This task is challenging for three reasons: (1) There are multiple types of entities and relations in social context, requiring methods to effectively model the heterogeneity. (2) The emergence of news in novel topics in social media causes distribution shifts, which can significantly degrade the performance of fake news detectors. (3) Existing fake news datasets usually lack of great scale, topic diversity and user social relations, impeding the development of this field. To solve these problems, we formulate social context based fake news detection as a heterogeneous graph classification problem, and propose a fake news detection model named **P**ost-**U**ser **I**nteraction **N**etwork (PSIN), which adopts a divide-and-conquer strategy to model the post-post, user-user and post-user interactions in social context effectively while maintaining their intrinsic characteristics. Moreover,we adopt an adversarial topic discriminator for topic-agnostic feature learning, in order to improve the generalizability of our method for new-emerging topics. Furthermore, we curate a new dataset for fake news detection, which contains over 27,155 news from 5 topics, 5 million posts, 2 million users and their induced social graph with 0.2 billion edges. It has been published on https://github.com/qwerfdsaplking/MC-Fake. Extensive experiments illustrate that our method outperforms SOTA baselines in both in-topic and out-of-topic settings.

*Work done during Erxue's internship at Tencent AI Lab.
†Corresponding author.

WWW '22, April 25–29, 2022, Virtual Event, Lyon, France

ACM ISBN 978-1-4503-9096-5/22/04…$15.00
https://doi.org/10.1145/3485447.3512163

CCS CONCEPTS

• **Information systems** → **Social networks**; • **Computing methodologies** → **Neural networks**.

KEYWORDS

Fake News Detection, Social Media, Graph Neural Network

ACM Reference Format:
Erxue Min, Yu Rong, Yatao Bian, Tingyang Xu, Peilin Zhao, Junzhou Huang, and Sophia Ananiadou. 2022. Divide-and-Conquer: Post-User Interaction Network for Fake News Detection on Social Media. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3485447.3512163

1 INTRODUCTION

The popularity of social media in recent years promotes the wide spread of fake news. Detecting fake news on social media is challenging as fake news pieces are intentionally written to mislead consumers, which makes it unsatisfactory to spot fake news from news content itself. For this sake, social context based fake news detection has attracted increasing attention in recent years. Generally, fake news detection based on social context meets two main challenges in modeling. The first one is that the information in social context of news is complicated and heterogeneous. As illustrated in Figure 1, there are multiple types of entities: e.g., posts, re-posts, replies and users, and multiple types of connections: e.g., responsive relations (post-post), publishing relations (user-post) and following relations (user-user). The heterogeneous characteristics and connections of these entities provides evidences of news verification from different views, but poses a challenge for effective utilization. The second one is the issue of distribution shifts [18] – where the training distribution differs from the test distribution – is prevalent in social media. For example, the fake news detector/classifier were trained on labeled data with ordinary topics covering sports, politics, entertainment et al. However, all of a sudden, some "black swan" incident such as the COVID-19 happens, which constitutes the novel test topic. The existence of distribution shift could significantly degrade the accuracy of the deployed fake news detector.

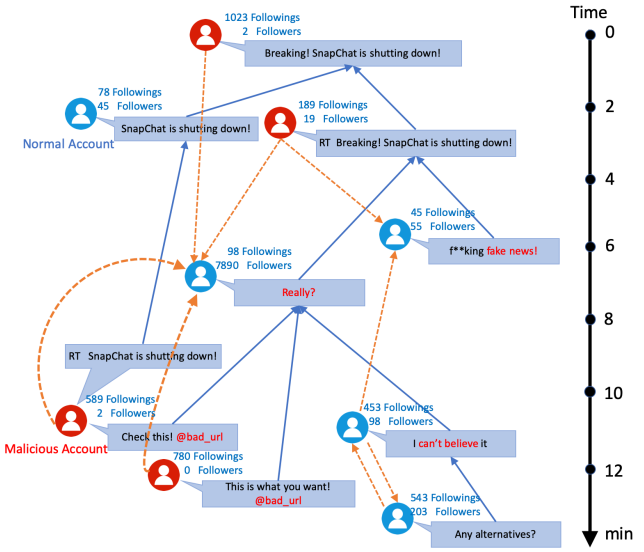


Figure 1: An example of a fake news propagation on Twitter, where the blue lines denote responsive relations and the orange lines denote following relations. The malicious account is the initial spreader of the fake news. The normal respondents are likely to query, oppose or comment on the news, but the malicious accounts/bots will target the people with many followers and reply them with links of low-credibility content to get a lot of visibility. Besides, they tend to follow the spreaders but few users follow them. In summary, the stance in posts, the diffusion structure, the account credibility and their social networks make the fake news propagation distinctive.

The difficulties in social context based fake news detection raise a requirement for both properly curated datasets and methods. An appropriate dataset shall contains rich social context information and reflect realistic distribution shifts in terms of that the out-of-topic test news be sufficiently different from the training news. None of existing fake news datasets [7, 8, 21, 24, 32, 34, 36, 44] satisfy the above two desiderata simultaneously[1]. This motivates us to curate a new dataset from Twitter which contains realistic news in various topics, and contains rich social context: e.g, posts, users, social networks. We firstly study the gap of in-topic performance and out-of-topic performance on this new dataset by benchmarking several existing baseline methods, and observe a significant drop. This verifies the realism of the new dataset in terms of distribution shifts. Moreover, we develop a new approach, named Post-User Interaction Network (PSIN), which models social context based fake news detection as a heterogeneous graph classification problem, and adopt a divide-and-conquer strategy to model the social context effectively. Additionally, an adversarial topic discriminator is applied to force the model to learn topic-agnostic features, which significantly improves model performance in out-of-topic settings (train and test with data in different topics). Our model is verified to enjoy superior performance than existing benchmarks.

[1]Among them only **FakeHealth** publishes the user network, and only **Fake-NewsNet** touches two topics: politics and entertainment.

The main contributions of this work are summarized as follows:

- We construct and publicize a new fake news dataset with social context named MC-Fake[2], which contains 27,155 news events in 5 topics, and their social context composed of 5 million posts, 2 million users and induced social graph with 0.2 billion edges.
- We propose a novel Post-User Interaction Network (PSIN), which applies divide-and-conquer strategy to model the heterogeneous relations. Specifically, we integrate the post-post, user-user and post-user subgraphs with three variants of Graph Attention Networks based on their intrinsic characteristics. Additionally, we employ an additionally adversarial topic discriminator to learn topic-agnostic features for veracity classification.
- We evaluate our proposed model on the curated dataset in two settings: in-topic split and out-of-topic split. The superior results of our model in both settings reveal the effectiveness of the proposed method.

2 RELATED WORK

2.1 Fake News Datasets

Fake news detection has attracted increasingly more attention and many fake news detection datasets are developed and publicly released. Many of them only contain news contents. For example, **BuzzFeedNews** [3] specializes in political news published on Facebook during the 2016 U.S. Presidential Election. **LIAR** [44] collects 12.8K short statements with manual labels from the political fact-checking website. **FA-KES** [32] consists of 804 articles around Syrian war. In addition to news contents, several datasets containing social context such as user comments and reposts on the social media platforms. **CREDBANK** contains about 1000 news events and 60 million tweets, labeled by Amazon mechanical Turk. **Twitter15** [24] contains 778 reported events between March 2015 to December 2015, with 1 million posts from 500k users. **FakeNewsNet** [36] is a data repository with news content and related posts, containing political news and entertainment news which are checked by politifact and gossiocop. **FakeHealth** [8] is collected from healthcare information review website Health News Review, it contains over 2000 news articles, 500k posts and 27k user profiles, along with user networks. Due to the quickly spread of COVID-19 virus, many related fake news datasets are also constructed [7, 21, 34]. **COAID** [7] collects 1,896 news, 183,654 related user engagements, 516 social platform posts about COVID-19, and ground truth labels. **FakeCovid** [34] is a multilingual cross-domain dataset of 5,182 fact-checked news article for COVID-19 from 92 different fact-checking websites. **MM-COVID** is a multilingual and multidimensional COVID-19 fake news data repository, containing 3,981 pieces of fake news content and 7,192 trustworthy information from 6 different languages. These datasets pushed forward the development of fake news detection in recent years. Among all these datasets, **FakeHealth** is the only one publishing the social network of involved users, but it only contains news on health. In summary, it is necessary to construct a new fake news detection datasets which contains multiple news topics and the involved user social network.

[2]https://github.com/qwerfdsaplking/MC-Fake
[3]https://github.com/BuzzFeedNews/2016-10-facebook-fact-check

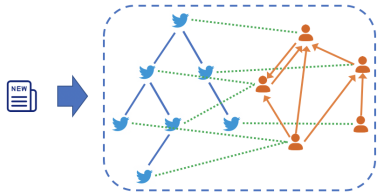


Figure 2: An illustration of a news event, which can be considered as a heterogeneous graph.The user-user following relations shall be directional since one can not infer B-follow-A from A-follow-B.

2.2 Social Context-based Fake News Detection

We briefly introduce existing work on fake news detection on social media. Fake news detection methods generally focus on using news contents and social contexts. News content based models generally include 1) style-based: capturing specific writing styles such as deception [10, 31] and non-objectivity [29]; 2) Knowledge-based: using external sources to fact-check claims in news content [27, 49]. Some work also make use of multi-modal features such as pictures or videos [46]. Apart from news contents, social context related to news pieces contains rich information to help detect fake news. Social context based approaches generally include post-based and user-based approaches. Post-based methods attempt to exploit the stance information and propagation patterns in the post propagation trees. Previous models can be categorized into three groups: Sequential Modeling [20, 24, 30, 52], Explicit responding path modeling [4, 19, 26, 47] and Implicit attention modeling [17, 43]. Sequential Modeling methods treat the posts as a sequence in chronological order, ignoring the propagation structure. Explicit responding path modeling methods use topology-aware models like graph neural networks to model the explicit responding paths, ignoring the implicit relations between posts. Implicit attention modeling methods use self-attention mechanisms [41] to capture the relations between all posts, but has the potential risk of introducing extra noise, and the computational complexity is quadratic to the post count. The limitations of these methods stimulate us to explore more principled method for propagation tree modeling. In terms of user-based approaches, Yang et al [50] proposes an unsupervised fake news detection algorithm by utilizing users' opinions on social media and estimating their credibilities. TriFN [39] is a tri-relationship embedding framework, which models publisher-news relations and user-news interactions simulataneously for fake news classification. FANG [28] is a graphical model which model the relations among user, source and news simultaneously. However, as far as we know, none of previous work jointly consider all of post propagation tree, user social graph and post-user interaction graph at the same time.

3 PROBLEM STATEMENT

We define a fake news dataset with social contexts as

$$\mathbf{D} = \{\mathbf{T}, G^U, G^{UP}\},$$

where $\mathbf{T} = \{T_1, T_2, ...T_{|\mathbf{T}|}\}$ is the set of news events, and $T_i = \{p_1^i, p_2^i, ...p_{M_i}^i, G_i^P\}$ is the related post set of i-th news event, with M_i denoting the number of posts, p_j^i denoting the j-th post. G_i^P is the propagation structure of posts, which can be a set of tree structures. Specifically, G_i^P is defined as a graph $\{V_i^P, E_i^P\}$, where $V_i^P = \{p_1^i, p_2^i, ...p_{M_i}^i\}$, and $E_i^P = \{e_{i(st)}^P | s, t = 1., , , .M_i\}$ represents the set of edges from responded posts to responsive posts. $G^U = \{V^U, E^U\}$ is the user network, where $V^U = \{u_1, u_2, ...u_N\}$ is the set of all users involved in the entire dataset, and $E^U = \{e_{st}^U | s, t = 1, 2, , , N\}$ represents the set of following edges from users to their followings. $G^{UP} = \{V^U \cup V^P, E^{UP}\}$ is the the bipartite graph between all involved users V^U and all involved posts V^P in $\mathbf{T}$, with $E^{UP} = \{e_{st}^{UP} | s = 1, ...N, t = 1, ..., M\}$ denotes the set of is-author edges from users to their published posts. Based on the global graph G^U and G^{UP}, we can generate two induced graphs G_i^U and G_i^{UP} for each news event T_i, according to its involved posts. Therefore, we reformulate T_i as a heterogeneous graph with $T_i = \{p_1^i, p_2^i, ...p_{M_i}^i, G_i^P, u_1^i, u_2^i, ...u_{N_i}^i, G_i^U, G_i^{UP}\}$, where N_i denotes the number of involved users. As illustrated in Figure 2, each news event T_i can be considered as a heterogeneous graph, with two types of nodes: post and user, and three types of edges: post-post, user-user and user-post. Each event T_i is associated with a ground-truth veracity label $y_i^V \in \{F, R\}$ (i.e. Fake, news or Real news). In our dataset, we also associate T_i with a topic label $y_i^C \in$ {Politics, Entertainment, Health, Covid-19, Sryia War}. We formulate the social context based fake news detection task as a supervised classification problem:

Problem 1. *Given the training set $\mathcal{T}_{train} = \{\mathbf{T}_{train}, Y_{train}^V, Y_{train}^C\}$, and the testing set $\mathcal{T}_{test} = \{\mathbf{T}_{test}\}$, how to learn a classifier $f : T_i \rightarrow y_i^v$ from $\mathcal{T}_{train}$ and then predict the veracity label Y_{test} for $\mathcal{T}_{test}$.*

Note that the connection between news events via shared users might also contain useful evidences, but considering them would make the definition and modeling much more complicated, so we will take this part as future work. Naturally, we do not assume the topic labels of test data are available as novel topics are continuously emerging in real world. This constitutes the distribution shift issue.

4 DATASET CONSTRUCTION AND STATISTICS

To obtain reliable ground truth labels on veracity of news, the most common solution is to utilize fact-checking websites such as PolitiFact, Snope and so on. There are already many previous work along this line of research. Therefore, we do not repeat this actions, instead, we collect labelled fake news from existing datasets and collect real news from reliable sources. We conduct a uniform filter criterion to select reliable instance. Then we retrieve related tweets, retweets and replies, with the corresponding users on the Twitter platform. The "following" relationship between users are also retrieved to obtain the induced user network. We elaborate the detailed construction process in Appendix A due to the space limits.

After the data retrieval and construction, we obtain a fake news dataset with 27,155 news events from five topics: Politics, Entertainment, Health, Covid-19 and Syria War, 5 million posts, 2 million users and an induced user social graph with 0.2 billion edges. Table 1 illustrates the statistics of collected datasets. Here we compare the distributions of tweets count, retweets count, reply count and user count to illustrate the difference between fake and real news. For

Politics and Covid-19 topics, the posts and users of fake news are significantly larger than real news. However, we find the opposite phenomenon for topics Entertainment, Health and Sryia War. These observations reveal the challenge of fake news detection, especially for out-of-topic news detection. We conduct more detailed sentiment analysis, bot score analysis and network analysis in Appendix B, illustrating the discrepancy between fake news and real news.

5 METHODOLOGY

In this section, we introduce our proposed Post-User Interaction Network (PSIN). Although the posts and users constitute a heterogeneous graph, their intrinsic characteristics limit the effectiveness of off-the-shelf heterogeneous graph models such as HGAN [45], HetGNN [51], HGT [13], etc. For example, the tree structures of post-post subgraph is quite different from the directed user-user subgraph. To solve these issues, we design an organized learning mechanism based on divide-and-conquer strategy to integrate different aspects while maintaining their intrinsic characteristics. Generally, we decompose the original graph into three parts: post propagation tree, user social graph and post-user interaction graph and process them individually, then, we perform the integration process at the end. As illustrated in Figure 3, our model generally includes five parts: Hybrid Node Feature Encoder (HNFD) for node representation, Tree Graph Attention Network (Tree-GAT) for post tree modeling, Relational Graph Attention Network (R-GAT) for user graph modeling, a post-user fusion layer for information interaction based on user behaviours, and a veracity classifier with an additional adversarial topic discriminator for topic-agnostic model learning.

5.1 Hybrid Node Feature Encoder

In order to learn the high-level features in the complex propagation structure and social graph, it is crucial to extract the nodes' features effectively at first. The purpose of the HNFD module is to generate the unified vector by exploiting textual and meta features simultaneously and highlighting the salient features that are likely to reveal the veracity. For news event T_i, we have nodes set $\{p_1^i, p_2^i, ...p_{M_i}^i, u_1^i, u_2^i, ...u_{N_i}^i\}$, and we discard the superscript i for simplicity in the following sections. Since each node has two part features: textual features and meta features, we have $p_j = \{\mathbf{t}_j^p, \mathbf{m}_j^p\}$ and $u_k = \{\mathbf{t}_k^u, \mathbf{m}_k^u\}$. The post meta features $\mathbf{m}^p$ consist of features such as like count, retweet count, reply count, sentiment score, etc, and the user meta features $\mathbf{m}^u$ includes verified flag, follower count, following count, etc. The full list of meta features are list in Table 6 in Appendix.

5.1.1 Text Content Encoding. Textual content is a strong indicator for discovering potential deception. Compared with real news, posts spreading fake news or rumour tend to exhibit certain patterns: the malicious spreader would apply misleading or exaggerated expressions to attract attention or stimulate the public mood while the normal users tend to express negative stance such as query or oppose [6]. In terms of user-related texts, i.e., user description, some bot-like flag or political stance could also implies the credibility of the users. There are many methods to represent text in fake news detection, such as TF-IDF [3], CNN [16], LSTM [12], Transformer [41] and BERT [48]. In our work, we apply word embeddings with CNN as our textual feature extractor, which shows the best performance and efficiency in our experiments. Let $\mathbf{c}_j$ be the extracted text embedding for the j-th node.

5.1.2 Meta feature based Gate Mechanism. The text embeddings compress important semantic information which is crucial for veracity detection. However, the importance of each node is different. Intuitively, the meta features like retweet count or follower count implies the popularity and social attention, which can be used to infer the importance of the given node. Therefore, we design a gate mechanism based on meta features to filter the text features as illustrated in Figure 4. Specifically, given meta features $\mathbf{m}_j$ of j-th node, we calculate its contribution score to measure how important the text features of j-th node will be:

$$g_j = \sigma(\mathbf{W}^m \mathbf{m}_j + \mathbf{b}^m),$$

where σ is an activation function which maps the input into $[0, 1]$, $\mathbf{W}^m$ and $\mathbf{b}^m$ are trainable parameters. Finally, the representation of j-th node is denoted as follows:

$$\mathbf{n}_j = g_j \mathbf{c}_j \oplus \mathbf{m}_j,$$

where $\oplus$ is the concatenation operator. Therefore, given input sequence $\{p_1, p_2, ...p_M, u_1, u_2, ...u_N\}$ for the i-th news event, we obtain the post feature matrix $\mathbf{P} = \{\mathbf{h}_1^P, \mathbf{h}_2^P, ..., \mathbf{h}_M^P\}$ and the user feature matrix $\mathbf{U} = \{\mathbf{h}_1^U, \mathbf{h}_2^U, ...\mathbf{h}_N^U\}$.

5.2 Post Tree Modeling

The natural characteristics of post trees make it unsuitable to directly apply graph models. When directly using graph neural networks to represent the post trees, each post node in the tree structure can only aggregate information from their direct neighbors in each layer, and k-hop information aggregation requires k layers of network. However, as illustrated in Figure 5, given the target post node, intuitively, all its children nodes might discuss about it, not limited to its directed respondents. For example, when a controversial post is published, discussion on it would construct a deep sub-tree, and replies after even ten rounds could also contain relevant information about it. On the other hand, the post is likely to be relevant to the nodes in its replying conversation, i.e., the path from the root node to the target node. This situation is more obvious when two users have multiple interaction through response. The unique characteristics of post tree poses a challenge for applying GNNs. For these sake, we propose Tree-GAT to effectively model the propagation structure. Tree-GAT includes two modules: Edge Augmentation and Depth-aware Graph Attention.

- **Edge Augmentation:** In order to bridge connections between all relevant posts in the propagation trees, we firstly augment the propagation trees through connecting a node to all its children nodes and conversation nodes. Let A^P be the adjacent matrix of the propagation trees G^P of i-th news event, with $A_{ij}^P = 1$ denoting that the i-th post is the respondent of j-th post. We calculate the augmented adjacent matrix $\widehat{A}^P$ as follows:

$$A_{\mathrm{BU}}^P = \sum_{d=1}^{d_{\max}} (A^P)^d, A_{\mathrm{TD}}^P = A_{BU}^{P\ \top},$$
$$\widehat{A}^P = A_{BU}^P + A_{TD}^P, \tag{1}$$

Table 1: The statistics of the dataset

Topics	Politics		Entertainment		Health		Covid		Syria War	
Labels	Fake	Real	Fake	Real	Fake	Real	Fake	Real	Fake	Real
News Count	225	1026	2587	8846	590	5120	843	5393	194	2230
Tweet Count (Sum)	51343	140940	128109	504936	75465	695225	33201	285511	9532	227663
Tweet Count (Avg)	228.19	137.37	49.52	57.08	127.91	135.79	39.38	52.94	49.13	102.09
Retweet Count (Sum)	71143	221364	190851	788937	27142	547610	178777	455269	5433	245316
Retweet Count (Avg)	316.19	215.75	73.77	89.19	46	106.96	212.07	84.42	28.01	110.01
Reply Count (Sum)	39342	162108	99362	490452	5682	188730	157835	297559	465	123279
Reply Count (Avg)	174.85	158	38.41	55.44	9.63	36.86	187.23	55.18	2.40	55.28
User Count (Sum)	135338	400815	362195	1504381	91924	1262745	315739	888650	13517	517419
User Count (Avg)	601.50	390.66	140.01	170.06	155.80	246.63	374.54	164.78	69.68	232.03

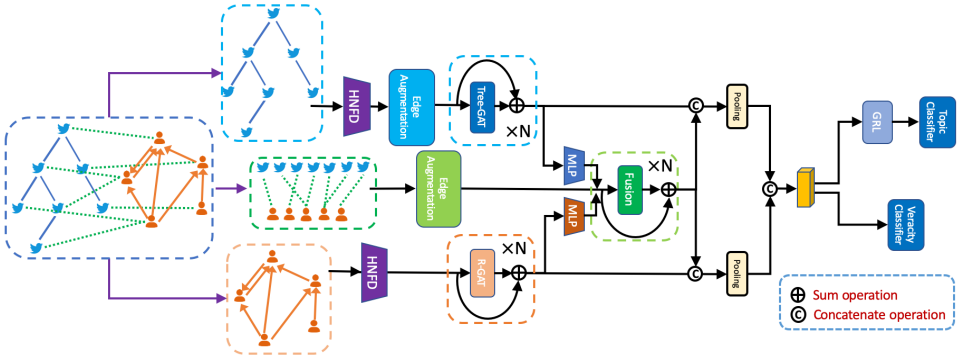


Figure 3: The architecture of PSIN. We decompose the original propagation tree with user information as three subgraphs: post propagation tree, user social graph, and user-post interaction graph. We first use the HNFD module to represent post and user nodes. Afterwards, We use Tree-GAT, R-GAT to represent the augmented post tree and directed user graph individually, and then integrate them via a GAT-based fusion network on the augmented user-post interaction graph. The concatenated and pooled representation is fed into a veracity classifier and an adversarial topic classifier.

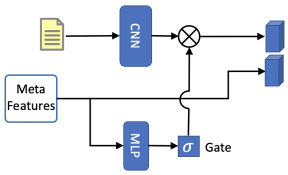


Figure 4: The structure of the hybrid node feature encoder

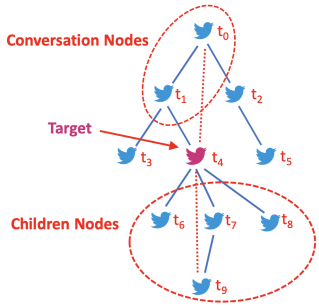


Figure 5: The structure of a post tree. The blue solid lines denote the responsive relations, and the red dash lines denote the augmented edges.

where $d_{\max}$ is the maximum depth of propagation trees in the news event.

- **Depth-aware Graph Attention:** Given the augmented adjacent matrix $\widehat{A}^P$ and post node feature matrix $\mathbf{H}^0 = \{\mathbf{h}_1^0, \mathbf{h}_2^0, ..., \mathbf{h}_M^0\}$, we adopt a modified Graph Attention Network (a.k.a GATv2) [5] as our backbone to represent the graph, which fixes the static attention problem of the standard Graph Attention Network [42] and show more robust performance. It adopts the attention mechanism that assigns different importance scores to the neighbors. The attention score between i-th node and j-th node can be formulated as:

$$e_{ij} = \mathbf{a}^\top \text{LeakyReLU}(\mathbf{W} \cdot [\mathbf{h}_i || \mathbf{h}_j]),$$
$$\alpha_{ij} = \text{Softmax}(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \in \mathcal{N}(i)} \exp(e_{ik})}, \quad (2)$$

where $\mathbf{a} \in \mathbb{R}^d$ is a parameter vector, $\mathbf{W} = [\mathbf{W}_s || \mathbf{W}_d]$ with $\mathbf{W}_s$ and $\mathbf{W}_d$ are parameter matrices to project source nodes and target nodes, e_{ij} and α_{ij} are unnormalized and normalized attention between the adjacent nodes i and j. After calculating the attention scores for all neighbor nodes, the central node's representation is updated by the aggregating features weighted by the attention scores:

$$\mathbf{h}_i' = \sigma(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} \mathbf{W}_d \mathbf{h}_j),$$

where σ denotes any non-linear functions. GAT treats all edges equally, however, in a propagation tree, there is semantic drift in distant nodes. Therefore, we propose depth-aware graph attention, which integrates the relative depth information into the attention by modifying the calculation of attention score as:

$$e_{ij} = \mathbf{a}^\top \text{LeakyReLU}(\mathbf{W} \cdot [\mathbf{h}_i || \mathbf{h}_j] + \mathbf{v}[d(i,j)]),$$

where $d(i,j) = d_i - d_j + d_{\max}$ with d_i being the depth of i-th node and $d_{\max}$ is the maximum depth of all trees. $\mathbf{v}[d(i,j)] \in \mathbb{R}^d$ are also trainable position vectors, enabling the network being aware

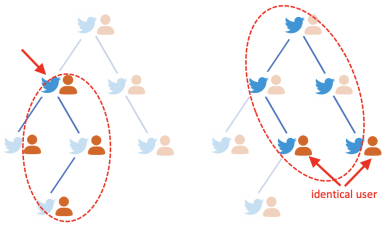


Figure 6: An illustration of post-user relations in our interaction graph. The left part shows that the relevant users of a post are its spreaders, and the right part shows that the relevant posts of a user are the posts he/she spreads.

of relative positions between nodes (relative temporal order and relative depth). Additionally, we also add residual connections to the update equation:

$$\mathbf{h}_i' = \sigma(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} \mathbf{W}_d \mathbf{h}_j) + \mathbf{h}_i. \tag{3}$$

Let $\mathbf{H}^0 = \mathbf{P}$, we have $\widehat{\mathbf{P}} = \mathbf{H}^K = \{\widehat{\mathbf{h}}_1^P, \widehat{\mathbf{h}}_2^P, \ldots, \widehat{\mathbf{h}}_M^P\}$ after K layers of Tree-GAT representation.

5.3 User Social Graph Modeling

In this section, we introduce our method for modeling the social graph of involved users. Given the user representations $\mathbf{U} = \{\mathbf{h}_1^U, \mathbf{h}_2^U, ..., \mathbf{h}_N^U\}$ and the induced local user graph G^U. We attempts to obtain the graph-aware user representations. We generate the adjacent matrix $\mathbf{A}^U$ based on G^U, where $\mathbf{A}_{ij}^U = 1$ denotes there is a follow relation from the i-th user and j-th user. Since the followers and followings reveal different aspects to represent a target node, we divide the neighbours of a user as three groups: only follow relation, only followed, friend (follow and followed), and thus the corresponding adjacent matrices are:

$$\mathbf{A}^{\text{friend}} = \mathbf{A}^U \cdot \mathbf{A}^{U\top}, \mathbf{A}^{\text{follow}} = \mathbf{A}^U - \mathbf{A}^{\text{friend}},$$
$$\mathbf{A}^{\text{followed}} = \mathbf{A}^{U\top} - \mathbf{A}^{\text{friend}}.$$

In order to distinguish different edges in the message passing process, we propose Relational Graph Attention Network (R-GAT), which calculates the attention score between nodes as follows:

$$e_{ij} = \mathbf{a}_{r(i,j)}^\top \text{LeakyReLU}(\mathbf{W} \cdot [\mathbf{h}_i || \mathbf{h}_j]),$$

where $r(i,j) \in \{0,1,2\}$ denotes the edge type, with $\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2 = \mathbf{a}_0 + \mathbf{a}_1$ as different parameter vectors for follow relations, followed relations and friend relations respectively. The calculation of normalized score α_{ij} and update equations are the same with Equation 2 and Equation 3. Given the user feature matrix $\mathbf{U}$, we obtain $\widehat{\mathbf{U}} = \{\widehat{\mathbf{h}}_1^U, \widehat{\mathbf{h}}_2^U, \ldots, \widehat{\mathbf{h}}_N^U\}$ after multiple R-GAT layers.

5.4 Post-User Interaction

The interaction between users and posts also provides clues for veracity detection. For example, there are some abnormal accounts which might publish hundreds of posts in a news event. These accounts can be bots who aim to spread the information for certain purposes, or fact checking accounts who hope to interrupt the spreading process. Neither post propagation tree modeling nor user network modeling can capture such patterns. For this sake, we propose a user-post fusion layer to enrich the representations of both user and post nodes. We construct a user-post graph according to users' behaviors. As illustrated in Figure 6, we assume that the spreaders of a given post can express a pattern of social effect of it, and the posts a user spreads describe the characteristics of the user. Based on this assumption, we calculate the adjacent matrix $\widehat{\mathbf{A}}^{UP} \in \mathbb{R}^{N \times M}$ of the bipartite user-post graph as:

$$\widehat{\mathbf{A}}^{UP} = \mathbf{A}^{UP}(\sum_{d=1}^{d_{\max}} (\mathbf{A}^P)^d),$$

where $\mathbf{A}^{UP} \in \mathbb{R}^{N \times M}$ is the adjacent matrix of the *is-author* graph G^{UP}, and $\mathbf{A}^P$ is the responding matrix as used in Equation 1. In order to represent the user-post graph using GNN, we firstly project their representations into a unified space using two projection matrices:

$$\mathbf{H}^P = \mathbf{W}^P \widehat{\mathbf{P}}, \mathbf{H}^U = \mathbf{W}^U \widehat{\mathbf{U}}.$$

Then, we treat the graph as homogeneous graph and obtain $\mathbf{H} = \text{Concat}(\mathbf{H}^P, \mathbf{H}^U)$. The adjacent matrix is defined as:

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A}^{UP^T} & 0 \\ 0 & \mathbf{A}^{UP} \end{bmatrix} \tag{4}$$

We use the standard GATv2 to represent the nodes, and the update rule in each layer is:

$$\mathbf{H}' = \text{GATv2}(\mathbf{H}, \widetilde{\mathbf{A}}) + \mathbf{H}.$$

We obtain $\widetilde{\mathbf{H}} = \{\widetilde{\mathbf{h}}_1^P, \widetilde{\mathbf{h}}_2^P, \ldots, \widetilde{\mathbf{h}}_M^P, \widetilde{\mathbf{h}}_1^U, \widetilde{\mathbf{h}}_2^U, \ldots, \widetilde{\mathbf{h}}_N^U\}$ after the post-user interaction layers. Then we obtain the final representation of posts and users as $\mathbf{P}' = \{\mathbf{h}_1^{P\prime}, \mathbf{h}_2^{P\prime}, \ldots, \mathbf{h}_M^{P\prime}\}$,$\mathbf{U}' = \{\mathbf{h}_1^{U\prime}, \mathbf{h}_2^{U\prime}, \ldots, \mathbf{h}_N^{U\prime}\}$, with $\mathbf{h}_i^{P\prime} = \text{Concat}(\widehat{\mathbf{h}}_i^P, \widetilde{\mathbf{h}}_i^P)$ and $\mathbf{h}_i^{U\prime} = \text{Concat}(\widehat{\mathbf{h}}_i^U, \widetilde{\mathbf{h}}_i^U)$.

5.5 Aggregation

Given the representation of posts and users: $\mathbf{P}' \in \mathbb{R}^{M \times d}, \mathbf{U}' \in \mathbb{R}^{N \times d}$, we adopt three Global Attention layers [22] to transform them into two fix-sized vectors respectively. The Global Attention layer is formulated as:

$$\mathbf{r} = \sum_{k=1}^{K} \text{Softmax}(f(\mathbf{h}_k)) \odot \mathbf{h}_k,$$

where $f : \mathbb{R}^d \to \mathbb{R}$ is a two-layer MLP. Finally, we obtain the two pooled vectors $\mathbf{p}, \mathbf{u}$, and we concatenate them to obtain the final representation of i-th news event as $\mathbf{z} = \text{Concat}(\mathbf{p}, \mathbf{u})$.

5.6 Topic-agnostic Fake News Classification

As illustrated in Table 1, the propagation characteristics across different topics vary significantly, we propose an auxiliary adversarial module and a veracity classifier to learn both class discriminative and domain invariant node representations. The overall objective is as follows:

$$\mathcal{L}(\mathbf{Z}, \mathbf{Y}^V, \mathbf{Y}^C) = \mathcal{L}_V(\mathbf{Z}, \mathbf{Y}^V) + \gamma \mathcal{L}_C(\mathbf{Z}, \mathbf{Y}^C). \tag{5}$$

The γ is the balance parameters. The $\mathcal{L}_V$ and $\mathcal{L}_C$ represent the veracity classifier loss and topic classifier loss, respectively. $\mathbf{Z}$ is the extracted feature matrix of the whole dataset, $\mathbf{Y}^V$ is the veracity labels and $\mathbf{Y}^C$ is the topic labels. The details are introduced as follows:

5.6.1 Veracity Classifier Loss. The veracity classifier loss $\mathcal{L}_V(\mathbf{Z}, \mathbf{Y}^V)$ is to minimize the cross-entropy loss of the veracity classification:

$$\mathcal{L}_V(\mathbf{Z}, \mathbf{Y}^V) = -\frac{1}{N_t} \sum_{i=1}^{N_t} y_i^V \log(f_V(\mathbf{z}_i)), \tag{6}$$

where $f_V : \mathbb{R}^d \to \mathbb{R}$ is an MLP classifier, $\mathbf{z}_i$ is the features of i-th news event, $y_i^V \in \{0,1\}$ is the corresponding veracity label, N_t is the number of instances in the training set.

5.6.2 Topic Classifier Loss. The topic classifier loss $\mathcal{L}_C(\mathbf{Z}, \mathrm{Y}^C)$ enforces that the representation after the feature extraction process of different topics are similar. To achieve this, we learn a topic classifier $f_C(\mathbf{Z}; \theta_C)$ parameterized by θ_C with an adversarial training scheme. On the one hand, we wish f_V can classify each news event into the correct veracity label via minimizing Equation 6. On the other hand, we would like the features from different topics are similar, so that the topic classifier cannot differentiate the topic of the news event. In our paper, we use Gradient Reversal Layer (GRL) [11] for adversarial training. Mathematically, GRL is defined as $Q_\lambda(x) = x$ with a reversal gradient $\frac{\partial Q_\lambda(x)}{\partial x} = -\lambda I$. θ_C is optimized by minimizing the cross-entropy topic classifier loss:

$$\mathcal{L}_C(\mathbf{Z}, \mathrm{Y}^t) = -\frac{1}{N_t} \sum_{i=1}^{N_t} y_i^C \log(f_C(\mathbf{z}_i)), \tag{7}$$

where y_i^C denotes the topic label for i-th news event. $\mathcal{L}_V(\mathbf{Z}, \mathrm{Y}^V)$ and $\mathcal{L}_C(\mathbf{Z}, \mathrm{Y}^C)$ are jointly optimized via the objective function in Equation 5, and all parameters are optimized using the standard backpropagation algorithms.

6 EXPERIMENTS

6.1 Baselines

In order to demonstrate the effectiveness of our proposed model, we employ the following methods as baselines.

- **PPC_RNN+CNN [23]**: A fake news detection approach combining RNN and CNN, which learns the fake news representations through the characteristics of users in the news propagation path.
- **RvNN [25]**: A tree-structured recursive neural network with GRU units that learn the propagation structure.
- **Bi-GCN [4]**: A GCN-based rumour detection model using bi-directional GCN to represent the propagation structure.
- **PLAN [17]**: A post-level attention model that incorporates tree structure information in the Transformer network.
- **FANG [28]**: A graphical fake news detection model based on the interaction between users, news, and sources. We remove the source network modeling part for fair evaluation.
- **RGCN [33]**: The relational graph convolutional network keeps a distinct linear projection weight for each edge type.
- **HGT [13]**: Heterogeneous Graph Transformer leverages node- and edge-type dependent parameters to characterize the heterogeneous attention over each edge.
- **PSIN** : Our proposed Post-User Interaction Model.
- **PSIN(-T)**: PSIN without the adversarial topic discriminator. We compare it with other baselines to demonstrate the superiority of our network architecture.

6.2 Settings

We implement PPC_RNN+CNN with Keras; RvNN, Bi-GCN, PLAN, FANG and our method with Pytorch. For PPC_RNN+CNN, RvNN, Bi-GCN and PLAN, we concatenate post features with corresponding user features to generate the node features to fit their architectures. For RGCN and HGT, we treat posts and users as two groups of nodes, which is the same with PSIN. We evaluate the methods in two settings: in-topic Split and out-of-topic Split. In in-topic split setting, we split the dataset into training set, validation set and testing set with ratio 6:2:2. We generate the split three times for more stable results. In out-of-topic split setting, we split the dataset according to topics as illustrated in Table 2, we split the data in training&validation set with ratio 8:2 to construct training set and validation set. Since the labels in the dataset is unbalanced, We adopt the widely-used AUC [9] and F1 Score as the evaluation metric for evaluation. We limit the number of posts of each event to 2000, the optimizer is Adam with learning rate selected from $\{10^{-3}, 10^{-4}, 10^{-5}\}$, the batch size is set to 32, the dimension of word embedding and hidden size of network is set to 100, the dropout rate is select from 0.1 to 0.9, the number of neural network layers for each part is selected from $\{2, 3, 4\}$, γ is selected from $\{0.01, 0.1, 0.5, 1.0\}$ and λ is select from $\{0.01, 0.1, 1.0\}$.

Table 2: Details of the out-of-topic split

ID	Training&Validation set	Testing set
1	Politics, Entertainment, Syria War	Health, Covid-19
2	Health, Covid-19	Politics, Entertainment, Syria War
3	Politics, Entertainment, Health	Covid-19, Sryia War

Table 3: The results of all methods in the in-topic setting.

Methods	Average	
	AUC	F1
SVM	0.7459	0.5210
GRU	0.8539	0.5458
PPC_RNN+CNN	0.8548	0.5419
BiGCN	0.8748	0.5482
PLAN	0.8635	0.5584
FANG	0.8235	0.5084
RGCN	0.8790	0.5930
HGT	0.8856	0.6166
PSIN (-T)	0.9039	0.6213
PSIN	**0.9063**	**0.6267**

Table 4: The results of all methods in the out-of-topic setting.

Methods	Average		Split 1		Split 2		Split 3	
	AUC	F1	AUC	F1	AUC	F1	AUC	F1
SVM	0.5593	0.1859	0.5737	0.1920	0.5012	0.1273	0.6031	0.2384
GRU	0.6012	0.2118	0.6150	0.2001	0.5298	0.1678	0.6589	0.2675
PPC_R+C	0.6001	0.1984	0.6151	0.1994	0.5344	0.1382	0.6507	0.2576
BiGCN	0.6087	0.2608	0.6201	0.2302	0.5245	0.2086	0.6815	0.3436
PLAN	0.6013	0.1883	0.6133	0.1923	0.5271	0.0283	0.6635	0.3442
FANG	0.6129	0.2371	0.6229	0.2134	0.5381	0.2029	0.6837	0.2949
RGCN	0.6138	0.1949	0.6194	0.2345	0.5400	0.2001	0.6880	0.1501
HGT	0.6147	0.2424	0.6215	0.2357	0.5372	0.2239	0.6913	0.2677
PSIN (-T)	0.6277	0.2693	0.6391	0.2721	0.5469	**0.2459**	0.6971	0.2898
PSIN	**0.6367**	**0.3094**	**0.6571**	**0.2722**	**0.5480**	0.2432	**0.7051**	**0.4120**

6.3 Overall Performance

Tables 3 and 4 show the performance of the proposed method and all the compared methods in in-topic setting and out-of-topic setting. From the results, we can make the following observations: (1) The results of all models on out-of-topic split mode are obviously inferior to that of in-topic split, demonstrating that the distribution shift issue makes the detection of newly-emerged news challenging. (2) Deep learning methods perform significantly better than SVM with hand-crafted features. It is reasonable as deep learning methods are capable to learn high-level representations of news stories to capture valid features. (3) Bi-GCN and PLAN have better performance than GRU and PPC_RNN+CNN in both settings. This is because GRU and PPC_RNN+CNN only utilize sequential information, while Bi-GCN and PLAN make use of propagation structure. (4) FANG has obviously worse performance than Bi-GCN and PLAN in the in-topic split setting, which is because it does not effectively utilize the post content and structures. However, this observation is the opposite in out-of-topic settings, which implies that post tree modeling methods are more likely to overfit

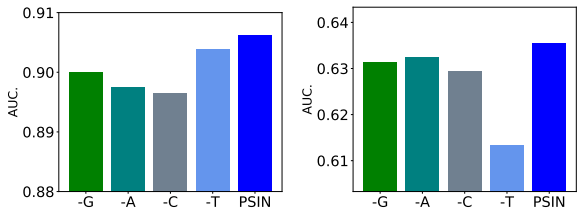
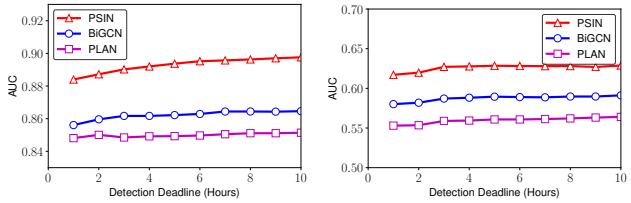


Figure 7: The performance of PSIN and its variants. Left: in-topic performance; Right: out-of-topic performance.



(a) In-topic Split (b) Out-of-topic Split

Figure 8: Results of Early Detection

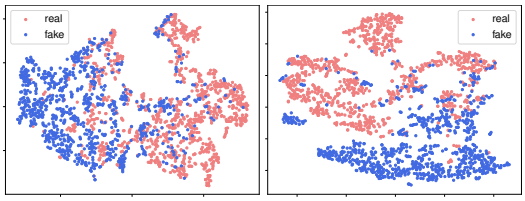
the training data, and thus impair their generalization capability to events of new topics. (5) RGCN and HGT based on the post-user heterogeneous graph have better performance than previous methods, which demonstrates the importance of jointly considering posts and users. (5) We can observe that our proposed PSIN(-T) and PSIN achieve state-of-the-art performance on both settings. Our model jointly model post-post, user-post and user-user graphs based on their own intrinsic properties and fuse them effectively, strengthening the capability of feature representation. Note that PSIN outperforms PSIN(-T) in both settings, and the gap is more salient in out-of-topic settings, showing that the adversarial topic classifier mitigates the overfitting issue and enables our model to learn more generalizable features for veracity detection.

6.4 Ablation Study

In order to analyze the contribution of each component of PSIN, we compare the proposed approach with the variants of PSIN. The empirical results of the model variants in terms of AUC on both settings are summarized in Figure 7. (-G) denotes our model with the gated mechanism for text feature extractor. (-A) represents our model without the edge augmentation techniques in both post network and post-user network. (-C) represents our model without the post-user Interaction network. (-T) represents our model without the adversarial topic classifier. We can observe that in both settings, the performance of PSIN decreases without any one of the four parts, which indicates that they are all vital to PSIN. Thirdly, the performance of -T decreases most in out-of-topic setting, which indicate that the adversarial topic classifier is indispensable when detecting news from new topics.

6.5 Early Detection

Early detection performance is another important metric to evaluate the method, which aims to detect fake news at the early stage of propagation. We set up a series of detection deadlines and only utilize the posts released before the deadlines with induced user networks to evaluate the performance of PSIN and other baselines.



(a) PSIN(-T) (b) PSIN

Figure 9: Visualization of learned feature representations of news events on the testing data.

Figure 8 shows the performances of our PSIN method versus BiGCN and PLAN at various deadlines in the two settings. One can observe that the proposed PSIN method achieves relatively high AUC at a very early stage. Additionally, its performance is stably superior to other models at all deadlines, which demonstrates the advantages of making use of both post and user structures.

6.6 Visualization of Effects of the Adversarial Topic Discriminator

To further analyze the effectiveness of the adversarial topic discriminator, we qualitatively visualize the final features learned by the feature extractor of PSIN on the testing set with tSNE [40] shown in Figure 9. The label for each event is real or fake. From Figure 9(a), we can observe that for the approach PSIN-T, it can learn distinguishable features, but the learned features are still twisted together. In contrast, the feature representations learned by the proposed model PSIN are more discriminative, and there are larger segregated areas among samples with different labels shown in Figure 9(b). This is because in the training stage, the adversarial topic discriminator attempts to suppress the dependencies between feature representations and specific topics.

7 CONCLUSION

In this paper, we firstly curate a new fake news detection dataset with multiple topics, spreading posts, users and their social relations, to facilitate further research in this field. Secondly, we formulate social context based fake news detection as a heterogeneous graph classification problem, and propose a novel Post-User Interaction Network (PSIN) to jointly model the heterogeneous post-user graph. Additionally, we also use an adversarial topic discriminator to enforce the model to learn topic-agnostic features, in order to improve its generalizability to newly-emerged incidents. Experiments in both in-topic and out-of-topic settings show that our approach outperforms all state-of-the-art baselines significantly.

ACKNOWLEDGMENTS

Erxue Min acknowledges the support from The University of Manchester - China Scholarship Council joint scholarship.

REFERENCES

[1] Mohammad-Ali Abbasi and Huan Liu. 2013. Measuring user credibility in social media. In *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*. Springer, 441–448.
[2] Koirala Abhishek. 2021. COVID-19 Fake News Dataset. https://data.mendeley.com/datasets/zwfdmp5syg/1
[3] Akiko Aizawa. 2003. An information-theoretic perspective of tf–idf measures. *Information Processing & Management* 39, 1 (2003), 45–65.

[4] Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 549–556.

[5] Shaked Brody, Uri Alon, and Eran Yahav. 2021. How Attentive are Graph Attention Networks? *arXiv preprint arXiv:2105.14491* (2021).

[6] Juan Cao, Junbo Guo, Xirong Li, Zhiwei Jin, Han Guo, and Jintao Li. 2018. Automatic rumor detection on microblogs: A survey. *arXiv preprint arXiv:1807.03505* (2018).

[7] Limeng Cui and Dongwon Lee. 2020. Coaid: Covid-19 healthcare misinformation dataset. *arXiv preprint arXiv:2006.00885* (2020).

[8] Enyan Dai, Yiwei Sun, and Suhang Wang. 2020. Ginger cannot cure cancer: Battling fake health news with a comprehensive data repository. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 14. 853–862.

[9] Tom Fawcett. 2006. An introduction to ROC analysis. *Pattern recognition letters* 27, 8 (2006), 861–874.

[10] Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 171–175.

[11] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Francois Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research* 17, 1 (2016), 2096–2030.

[12] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[13] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. 2020. Heterogeneous graph transformer. In *Proceedings of The Web Conference 2020*. 2704–2710.

[14] Hong Huang, Zhexue Chen, Xuanhua Shi, Chenxu Wang, Zepeng He, Hai Jin, Mingxin Zhang, and Zongya Li. 2021. China in the eyes of news media: a case study under COVID-19 epidemic. *Frontiers of Information Technology & Electronic Engineering* (2021), 1–15.

[15] Clayton Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 8.

[16] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188* (2014).

[17] Ling Min Serena Khoo, Hai Leong Chieu, Zhong Qian, and Jing Jiang. 2020. Interpretable Rumor Detection in Microblogs by Attending to User Interactions. *arXiv preprint arXiv:2001.10667* (2020).

[18] Pang Wei Koh, Shiori Sagawa, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, et al. 2021. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*. PMLR, 5637–5664.

[19] Sumeet Kumar and Kathleen M Carley. 2019. Tree lstms with convolution units to predict stance and rumor veracity in social media conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 5047–5058.

[20] Quanzhi Li, Qiong Zhang, and Luo Si. 2019. Rumor detection by exploiting user credibility information, attention and multi-task learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 1173–1179.

[21] Yichuan Li, Bohan Jiang, Kai Shu, and Huan Liu. 2020. MM-COVID: A Multilingual and Multimodal Data Repository for Combating COVID-19 Disinformation. arXiv:2011.04088 [cs.SI]

[22] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. 2015. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493* (2015).

[23] Yang Liu and Yi-Fang Brook Wu. 2018. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Thirty-second AAAI conference on artificial intelligence*.

[24] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. (2016).

[25] Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor detection on twitter with tree-structured recursive neural networks. Association for Computational Linguistics.

[26] Jing Ma, Wei Gao, and Kam-Fai Wong. 2019. Detect rumors on twitter by promoting information campaigns with generative adversarial learning. In *The World Wide Web Conference*. 3049–3055.

[27] Amr Magdy and Nayer Wanas. 2010. Web-based statistical fact checking of textual documents. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents*. 103–110.

[28] 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*. 1165–1174.

[29] 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).

[30] Yu Rong, Hong Cheng, and Zhiyu Mo. 2015. Why it happened: Identifying and modeling the reasons of the happening of social events. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1015–1024.

[31] Victoria L Rubin and Tatiana Lukoianova. 2015. Truth and deception at the rhetorical structure level. *Journal of the Association for Information Science and Technology* 66, 5 (2015), 905–917.

[32] Fatima K Abu Salem, Roaa Al Feel, Shady Elbassuoni, Mohamad Jaber, and May Farah. 2019. Fa-kes: A fake news dataset around the syrian war. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 13. 573–582.

[33] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*. Springer, 593–607.

[34] Gautam Kishore Shahi and Durgesh Nandini. 2020. FakeCovid–A multilingual cross-domain fact check news dataset for COVID-19. *arXiv preprint arXiv:2006.11343* (2020).

[35] Chengcheng Shao, Giovanni Luca Ciampaglia, Onur Varol, Alessandro Flammini, and Filippo Menczer. 2017. The spread of fake news by social bots. *arXiv preprint arXiv:1707.07592* 96 (2017), 104.

[36] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2020. Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. *Big data* 8, 3 (2020), 171–188.

[37] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter* 19, 1 (2017), 22–36.

[38] Kai Shu, Suhang Wang, and Huan Liu. 2018. Understanding user profiles on social media for fake news detection. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, 430–435.

[39] Kai Shu, Suhang Wang, and Huan Liu. 2019. Beyond news contents: The role of social context for fake news detection. In *Proceedings of the twelfth ACM international conference on web search and data mining*. 312–320.

[40] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[42] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[43] Amir Pouran Ben Veyseh, My T Thai, Thien Huu Nguyen, and Dejing Dou. 2019. Rumor detection in social networks via deep contextual modeling. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. 113–120.

[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] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous graph attention network. In *The World Wide Web Conference*. 2022–2032.

[46] 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*. 849–857.

[47] Penghui Wei, Nan Xu, and Wenji Mao. 2019. Modeling conversation structure and temporal dynamics for jointly predicting rumor stance and veracity. *arXiv preprint arXiv:1909.08211* (2019).

[48] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv* abs/1910.03771 (2019).

[49] You Wu, Pankaj K Agarwal, Chengkai Li, Jun Yang, and Cong Yu. 2014. Toward computational fact-checking. *Proceedings of the VLDB Endowment* 7, 7 (2014), 589–600.

[50] 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 *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 5644–5651.

[51] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V Chawla. 2019. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 793–803.

[52] Kaimin Zhou, Chang Shu, Binyang Li, and Jey Han Lau. 2019. Early rumour 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)*. 1614–1623.

A DATA CONSTRUCTION

In this section, we introduce the dataset construction process. We elaborate how we collect news contents with reliable ground truth labels, how we obtain additional social context and induced social graph.

A.1 News with ground truth

We collect fake news with ground truth labels from 7 existing datasets: FakeNewsNet [36], FakeHealth [8],FA-KES [32], FakeCovid [34], MM-COVID [21], COAID [7] and CFND [2]. We randomly sample real news from the all-the-news corpus [4] and a Covid-19 news corpus[14], which contain news collected from credible sources. We use the most frequent keywords in the collect fake news to retrieve real news of similar topics from this corpus. Similar to previous work, we choose to use the news title to retrieve related social context, so the titles should be informative enough to represent the contents of the news, and should not be too general. Therefore, we filter news whose titles are too short and do not express a specific event. For example, we would filter news with titles like "U.S. Imports & Exports" or "The early catastrophe".

A.2 Social Context Retrieval

We retrieve the social context of the source news from the Twitter, which is one of the most popular social media platform. Twitter provides several APIs to collect tweets and user engagement. We adopted Twitter's Academic Search API [5], which provides full-archive access to all historical data, to search tweets that directly post the news. Following [8], we used the titles as queries to search tweets. Then, we remove meaningless stop words and special tokens in the title to generate more general queries to search tweets. We filter tweets which contains overlapped words fewer than 5 with the original queries. Finally, we search tweets with the URLs of news if available. Note that we filter the URLs who refers to the fact checking articles instead of source news, which is a common noise in several datasets (i.e., FakeNewsNet, COAID, etc.). With the three search strategies, we obtained 2,150,000 tweets in total. After obtaining the directly posted tweets, we further scraped the retweets and replies. We limit the maximum number of replies for a tweet as 5000 for efficiency. We construct the propagation tree structures of each news based on the retweet relations and replying relations. Note that each news could associate several propagation trees.

A.3 User Social Network Construction

After we acquire all the tweets related to the news, we use a user crawler to obtain all involved users who tweet, retweet and reply in these tweets. Different from previous works [2, 7, 36], we further retrieve the users followings list and followers list to construct a large user social network, which is a induced graph of Twitter users' social relation graph. Due to the rate limits of Twitter APIs, we limit the user numbers in both followings list and followers list as 5000 for efficiency.

[4] https://components.one/datasets/all-the-news-2-news-articles-dataset/
[5] https://developer.twitter.com/en/products/twitter-api/academic-research

Table 5: The average sentiment scores of both real news and fake news

	Compound	Negative	Neutral	Positive
Real	0.080	0.079	0.771	0.150
Fake	0.065	0.092	0.776	0.122

B DATA ANALYSIS

In this section, we perform an exploratory study of posts, user profiles, network structures.

B.1 An Overview of Text Contents

In order to analyze the topic distribution of social context of fake and real news. We construct word cloud figure to visualize it. From Figures 10(a) and 10(b), we can observe that the general topics of them are similar, covering topics of politics, entertainment, health, coronavirus and Syria wars. Furthermore, we also visualize the word distribution in user descriptions. As illustrated in Figure 11(a) and 11(b), we have that user description usually express a user's appetite for news and political preference, which are useful to describe the social context of news.

(a) Real (b) Fake

Figure 10: The word cloud of post texts for real and fake news

(a) Real (b) Fake

Figure 11: The word cloud of user descriptions for real and fake news

B.2 Post and Response

Users express their opinions or emotions towards fake news through social media posts, such as sensational reactions or skeptical opinions, which are crucial signals to assess news credibility in general. To obtain statistical information from these, we perform sentiment analysis on replies to reveal the difference on sentiment polarity

between fake and real news. We adopt the widely-used model for social media text sentiment analysis, i.e., VADER [15]. The polarity scores contains four parts: compound score, negative score, neutral score and positive socre. The compound score is computed by summing the valence scores of each word in the lexicon, adjusted according to the rules, and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive). The negative, neutral and positive scores are ratios for proportions of text that fall in each category. Table 5 lists the average polarity scores of the replies. As expected, the compound scores of real news are significant higher than fake one. Besides, the negative scores of real news are significantly less. This implies that replies provides import information for identifying fake news.

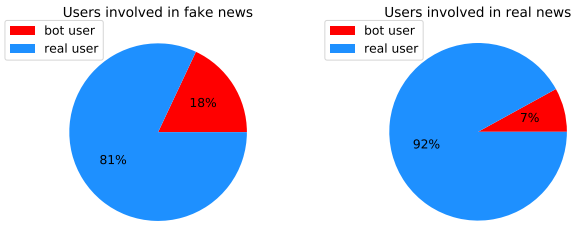


Figure 12: Bot scores on users related to fake news and real news.

B.3 User Credibility

Users profiles on social media are found to be correlated with fake news detection. [1, 38]. Abbasi et al. [1] shows that users with low credibility tends to spread misinformation. Besides, Research has shown that fake contents are more likely to be created and spread by non-human accounts, such as social bots or cyborgs. First, we explore whether the verified proportion of user accounts for fake news and true news are different or not. Specifically, we compare users who have engaged in over three real news and have never engaged in fake news, with users who have engaged in fake news, on the proportion of verified. We obtain 3.08% and 2.43% for the two group respectively, showing that the user accounts of fake news are more likely to be unverified. Secondly, we contrast fake news and real news in the bot-likelihood scores of involved users. We randomly sample 10000 users who are only involved in fake news and 10000 users who are only involved with real news. We obtain the bot-likelihoods of the users through the BotoMeter API [6], which is a commercial bots detection tools based on the features of network, user, followers, followings, content, sentiment and so on. We set the threshold of 0.5 on the bot score from the Botometer results to assign bot labels. Figure 12 show the ratio of the bot and real users involved in tweets related to fake and real news. We can see that bots are more likely to engage in fake news than real users, which is consistent with the observation in [8, 35, 36].

B.4 User Network

Users tend to form different network patterns on social media in terms of topics, interests and friendships, which comprise of the fundamental paths for information diffusion [37]. The different propagation patterns between fake news and real news highlights the importance of utilizing network-based features. We compare the average followers and followings for users who have engaged in fake news and only engaged in real news. Moreover, we also compare the same statistics based on our induced user graph, which our models are based on. As we can see in Figure 13, in both real social graph and induced graph, the numbers of both followers and followings for fake news involved users are significantly larger than that of users only spread real users. This observation demonstrates the importance of making use of user network features.

[6] https://botometer.iuni.iu.edu/

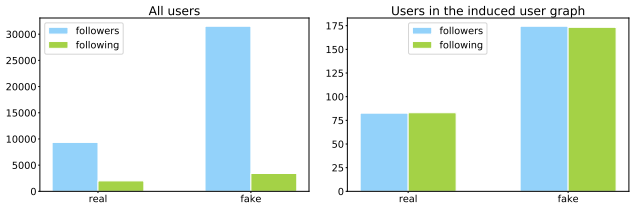


Figure 13: Average number of followers and followings for users who are involved in real news and fake news respectively.

Table 6: Detailed meta features of post and user nodes

Node type	Meta features	Example
Post	Post type	0/1/2*
	Retweet Count	10
	Reply Count	10
	Like Count	10
	Quote Count	10
	Created time	1501143981
	Sentiment Score	0.8
User	is_verified	1
	Following Count	100
	Followers Count	1000
	Tweet Count	1000
	List Count	10
	Account created time	1458483921
	Description length	20

*0 denotes tweet, 1 denotes retweet, 2 denotes reply.