Real-World Challenges in Fake News Detection: Dealing with Posts by Cold Users

Sai Keerthana Karnam[1], **Abhirup Kundu**[1], **Jashn Arora**[2], **Manish Jain**[2], **Animesh Mukherjee**[1]

[1]Indian Institute of Technology Kharagpur
[2]Google DeepMind
{saikeerthanakarnam.24@kgpian, animeshm@cse}.iitkgp.ac.in, {arorajashn, manishjn}@google.com.

Abstract

Social media serves as a primary source of information in the current digital era. Many people consume a vast range of information in a very short span, yet, amidst the stream of genuine information, fake news and rumors continue to spread. The need for effective detection models is becoming increasingly critical. Past user behavior and user engagement on a post are strong signals that SOTA approaches leverage for fake news detection and other post classification tasks. However, these approaches lean too heavily on knowing this past behavior, and thus suffer from a *cold user problem*, or users that are new or have minimal footprint on the platform. In this paper, we make three core contributions. We first establish the value of user behavior, both content and user-user interactions, in the task of fake news and rumor detection. We then establish the extensive prevalence of *cold users* in the real-world datasets, and show the need for newer algorithms considering cold users. We next propose a novel socially-aware context representation scheme – USER EVIDENCE NETWORK (UEN) – to detect the spread of misinformation and unverified information while efficiently navigating this cold user challenge. We introduce techniques that approximate missing / absent behavior data of a new user from existing users' interactions. By carefully addressing the cold user challenge, our work provides robust approaches targeting fake news and rumor detection for real-world platforms.

Code — https://github.com/saikeerthana00/UEN

Introduction

In the present digital age, social media platforms have emerged as a primary means by which people acquire information. Every day, consumers are exposed to a diverse range of content, from accurate news to unverified and misleading information. The continued rapid proliferation of fake news on platforms like Meta (Del Vicario et al. 2016; Allcott and Gentzkow 2017), X (formerly Twitter) (Grinberg et al. 2019; Vosoughi, Roy, and Aral 2018), and Reddit (Soliman, Hafer, and Lemmerich 2019) highlights the critical need for effective detection techniques.

Recently, Gong et al. (2023) presented a survey on fake news detection models and highlighted the effectiveness of graph neural networks (GNNs) in identifying fake news, because GNNs can simulate the social contexts and propagation patterns of information. Building upon GNNs, Xu et al. (2022); Yuan et al. (2019a); Bian et al. (2020) all present approaches that integrate both *local* and *global features* of the network. Local features include individual user interactions and immediate comment threads, while global features capture broader patterns across the network, such as overall communication trends and influence metrics.

Leveraging user behavior: Yang et al. (2012) and Castillo, Mendoza, and Poblete (2011) demonstrated the significance of incorporating user profile features, such as age, number of tweets, and followers, in enhancing the accuracy of fake news detection. However, this information is often incomplete, unavailable, or inaccurate, and metadata features can be biased. Further such information may lead to compromise of user privacy. To address these issues, we propose the USER EVIDENCE NETWORK (UEN) based framework which relies solely on the commenting and replying patterns of users to efficiently capture their behavioral traces.

Cold users: While past information on user behavior can bring in steady benefits to the GNN based fake news detection task (Shu, Wang, and Liu 2019), a major shortcoming is the absence of such information for a section of users who have just joined the platform or have done very little interactions so far. In this paper we call them the *cold users*. The lack of digital footprint in terms of prior posting and commenting patterns for these users can severely handicap the workings of the model and adversely affect the overall performance. To overcome this limitation, our work proposes novel heuristics to create (approximate) feature representations for the cold users by mapping their behavioral pattern to the existing users. This approach aims to improve the robustness of fake news detection models by mitigating the limitations associated with unavailability of traditional user profile and behavior features.

Temporal ordering: The interactions around a fake news post keep evolving in social media platforms. Thus, in the real world, the entire temporal context about a post might not be always available to ascertain whether that post is fake or not. The authors in Gong et al. (2023) highlights a key limitation in SOTA models where the complete future information of all posts are assumed to be available for detection. To maintain the realistic rigor in our work we take

non-overlapping training and test snapshots ordered in time and thereby do not rely on all the future data of a post. Our dataset creation approach mimics an actual implementation on a social media platform - models are developed on historical data, get deployed and provide value by classifying new unseen posts/comments.

The main contributions of this work are as follows.

- We propose the UEN framework that makes comprehensive use of the content of the source post, comments or reactions, comment pattern or tweeting pattern, and user information obtained from a global interaction-based graph to train GNN model variants to detect fake news or rumor.
- To improve the model's ability to generalize for samples involving cold users, we suggest multiple strategies for extracting feature representations from the global interaction-based graph. Even in the absence of prior data, we generate (approximate) meaningful feature representations by mapping the cold users to some of the existing users employing an array of novel heuristics.
- We show that incorporation of user evidence based features along with the textual features of the source post improves the accuracy of the prediction by $6-8\%$ for the different GNN variants. Further improvements are obtained in macro-F1 when feature representation of the cold users are acquired based on our proposed heuristics. Remarkably, the performance for the test users who have absolutely no footprint in the training set ('perfectly cold users') increases by around $9-10\%$ (macro-F1) when our proposed heuristics are employed.

Related Work

The early methods for fake news detection relied mainly on content-based features, such as linguistic cues and sentiment analysis, to identify misinformation (Rashkin et al. 2017). Miyazaki et al. (2023) used embedding features from sentenceBERT to train fake news classifiers using SVM and Random Forest, and to also finetune language models like BERT for the same task. Nakamura, Levy, and Wang (2020) integrated linguistic features using BERT embeddings, and visual features using InferSent and used a combination of both of these to create a fake news classifier. To improve detection accuracy, recent approaches have incorporated social context into their models. They tend to leverage post content features plus the content of the comments, the user profile descriptors (user's self-description, age of account, verified status), user connectivity (number of friends and followers), and structural and dispersion features of the post (post-comment propagation tree). Graph neural networks (GNNs) have emerged as a popular tool to model all these features of the post comment tree to build fake news classifiers. Han, Karunasekera, and Leckie (2020) trained GNNs incrementally for this task using the user profile and tweet timeline along with the structural features. On the other hand, Bian et al. (2020) proposed a bidirectional graph model, Bi-Directional GCN (Bi-GCN), and explored patterns of deep propagation and the structures of wide dispersion in rumor detection by operating on both the top-down and bottom-up propagation of rumors. Xu et al. (2023) proposed Hierarchically Aggregated GNN (HAGNN) and focused on different granularities of high-level representations of text content and fused the rumor propagation structure. Wei et al. (2024a) addressed the challenge of fake news detection for new posts by learning how to transfer structural knowledge learnt from existing propagation trees and classifying new posts with just their content. Specifically, they utilised both propagation and content features during training, but tested on samples that contain only content features.

Much of this body of work focuses on the 'local' social context of one particular post, and discards the 'global' social media network when addressing the task of detecting fake news. Global signals include a user's post/comment history, history of engagement on the platform, and the set of other users they interact with, and these global signals have significant value when considering the task of fake news detection. Sun et al. (2023) proposed a joint learning model named HG-SL, which is capable of catching the differences between true and fake news in the early stages of propagation through global and local user spreading behavior, but is blind to the content. Yuan et al. (2019b) presented a novel global-local attention network (GLAN) for rumor detection, which jointly encodes the local semantic and global structural information by modelling the global relationships among all source tweets, retweets, and users as a heterogeneous graph to capture the rich structural information for rumor detection. Su et al. (2023) proposed constructing an attributed hypergraph to represent non-textual and high-order relations for user participation in news spreading. However, all these approaches assume the availability of historical user data, making them ineffective in handling cold users, or users with minimal footprint on the given social media platform.

Several studies have explored the cold start problem in recommendation systems Ayub et al. (2020), online social networks Gong et al. (2021) and other domains Park and Chu (2009); Schein et al. (2002). These methods typically employ techniques like content-based filtering or collaborative filtering with side information to make predictions for new users. However, the cold user challenge in fake news detection presents unique complexities, as the users do not reveal whether a news is fake or not themselves. Table 1 summarizes prior works and the types of features they utilize: (a) Content – such as titles or news content; (b) Propagation patterns – including tweet-retweet or post-comment structures; (c) Historical user behavior – such as user-user interactions and posting habits; and (d) Cold user handling – the model's ability to generalize to users with limited (or no) historical data. Our work builds upon previous research in fake news detection and cold-start problems. We propose a novel approach that explicitly addresses the cold user challenge by leveraging existing user-user interactions to approximate the missing user's behavior data. By carefully integrating social context and user behavior, our model provides robust and practical solutions for fake news detection in real-world scenarios.

Paper	Content-based	Propagation patterns	Historical user behaviour	Handles cold users
Rashkin et al. (2017), Miyazaki et al. (2023), Nakamura, Levy, and Wang (2020)	✓	✗	✗	✗
Han, Karunasekera, and Leckie (2020), Bian et al. (2020) , Nakamura, Levy, and Wang (2020), Xu et al. (2023), Wei et al. (2024a)	✓	✓	✗	✗
Yuan et al. (2019b), Sun et al. (2023), Su et al. (2023)	✓	✓	✓	✗
Wei et al. (2024b)	✓	✓	✗	✗(Handles cold start problem but doesn't include user features)
Our work	✓	✓	✓	✓

Table 1: Prior works leverage content features, model propagation patterns, and incorporate historical user behaviour, but lack the ability to handle cold users. In contrast, our work addresses all four aspects, with a particular emphasis on cold user generalization.

Statistic	Fakeddit dataset	Gossipcop dataset
Total #samples	1,063,106	5,464
Text #samples	971,806	5,464
Text #samples ($\|\text{comments}\| \geq 1$)	583,305	5,464
Comments	9,291,298	294,288
Fake #samples	2,732	2,732
True #samples	402,753	2,732
Unique #users	1,592,037	76,356
Cold #users	430,439	14,273
Training set size	407,760	3,824
Training set timespan	1/6/2008 - 3/21/2019	5/13/2008 - 2/16/2018
Validation set size	58,315	546
Validation set timespan	3/21/2019 - 6/7/2019	2/16/2018 - 4/13/2018
Testing set size	116,639	1,094
Testing set timespan	6/7/2019 - 10/23/2019	4/13/2018 - 12/17/2018

Table 2: Dataset statistics: The time period between the training set and testing set do not overlap ensuring that there is no data leakage.

Dataset

For our experiments, we use the **Fakeddit** (Nakamura, Levy, and Wang 2020) and **Gossipcop**[1] datasets.

Fakeddit is derived from Reddit and consists of over 1 million submissions from 22 subreddits over a 10 year span. Each submission has a submission title and image, an author, comments made by users who engaged with the submission, and other metadata like the comments count and username of the author. For the classification, we considered a binary label, which represents whether the submission is `fake` or `true`. The original paper addresses the classification of multimodal samples, thus considering submissions with text, images, or a combination of both. In this paper, we will exclude submissions that have only images as the content since this number is substantially less. In our framework, we incorporate comments associated with each submission. Therefore, we consider samples with at least one comment. As depicted in Table 2, we use $583,305$ samples for our experiments.

Gossipcop dataset comprises source news articles along with associated tweets and retweets from X (formerly Twitter) conversations. Each tweet or retweet includes its text and the author information. Unlike the Fakeddit dataset, this does not have a user associated with the source post; therefore, we assume all source news items originate from a single, common user. For classification, we adopt a binary label indicating whether the source news is `fake` or `true`. Since only tweet IDs are publicly available and accessing tweet content via the X API is cost-prohibitive, we rely on pretrained BERT embeddings for tweet representations.

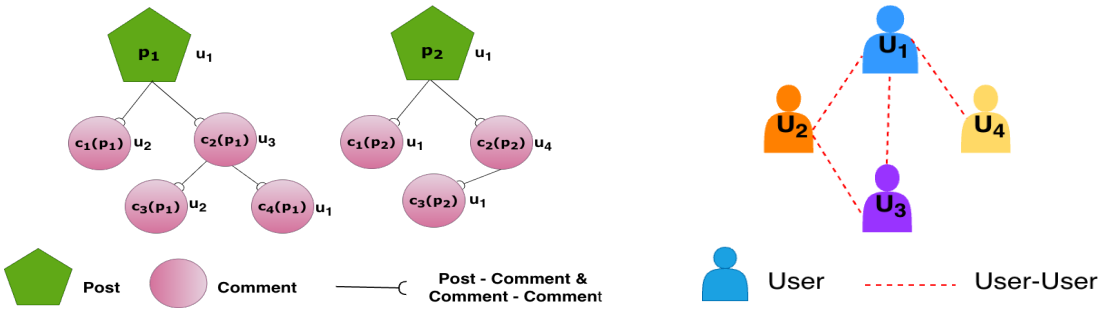
In summary, we utilize two datasets - Fakeddit and Gossipcop for our analysis. Each of our samples includes the submission or source news, author, comments or reactions associated with the submission or tweet-retweet, and the corresponding users. For comments (in Fakeddit), we include only text while omitting the images. In this paper, we use the terms submission (from Fakeddit) and source news (from Gossipcop) and post interchangeably, as well as comments/replies (from Fakeddit) and tweets/retweets (from Gossipcop), as they are treated similarly in our pipeline. We sort the data samples according to the timestamp and consider the first 70% as the training set, the next 10% as the validation set, and the remaining 20% as the test set. This ensures that the test samples are completely unknown and reflect real-world scenarios. We define cold/unknown users as those who are either non-existent or not present in the training samples. Table 2 notes the statistics of the dataset.

Problem Formulation

We use the following notations in the rest of the paper.

1. $P = \{p_1, p_2, \dots\}$ represents posts which have textual content and also have at least one textual comment associated with them.
2. $C(p_i) = \{c_1(p_i), c_2(p_i), \dots, c_n(p_i)\}$ represents the comments under the post p_i. Here n represents the num-

[1]https://github.com/safe-graph/GNN-FakeNews



(a) Pool of post-comment chains.

(b) Graph depicting the user-user interactions.

Figure 1: Graph representation of the social context.

ber of comments under the post p_i.

3. E_i is the list of edges which depicts the local relations as follows.
   (a) *Post-comment edges* $(p_i, c_j(p_i))$: Represents that a comment $c_j(p_i)$ is made on the post p_i.
   (b) *Comment-comment edges* $(c_j(p_i), c_k(p_i))$: Represents that a comment $c_k(p_i)$ is made in reply to the comment $c_j(p_i)$.
   Both types of edges are treated equally in the pipeline.
4. U_i is the set of social media users who either published the post p_i or made a comment on the post p_i.
5. $y_i = \{0, 1\}$ represents the ground truth assigned to source post p_i with 0 indicating that the post is fake and 1 indicating that the post is true. Similarly, $\overline{y}_i$ represents the predicted outcome.
6. $S_{\text{Train}} = \{s_1, s_2, s_3 \dots\}$ is the list of training data samples. Each data sample is denoted as $(p_i, C(p_i), E_i, U_i, y_i)$.
7. $S_{\text{Test}} = \{s_1, s_2, s_3 \dots\}$ is the list of test data samples. Each data sample is denoted as $(p_i, C(p_i), E_i, U_i)$.
8. U_G represents the global set of social media users.

$$U_G = \bigcup_{s_i \in S_{\text{Train}}} U_i \tag{1}$$

9. E_G is the list of edges which depicts the global relations between users U_G. Edge (u_i, u_j) exists if there is an interaction between u_i and u_j, where $u_i, u_j \in U_G$. The interactions could be of the following types.
   (a) User u_i commented on a post that was created by u_j.
   (b) User u_i replied to a comment made by u_j.

Our objective is to train a classifier capable of assigning a label 0 or 1 to the data samples S_{train} and S_{test}. Figure 1 represents the comprehensive details and interactions used to execute this task.

The UEN Framework

The proposed UEN framework consists of five main components. First, the **User history module** uses the global relations mentioned in the problem formulation section to generate a feature representation for each user. Next, the **Content and behavior representation module** generates detailed representation for posts and comments by concatenating features from textual content and user behavior. These combined features along with the local relations are then fed to **GNN module** to find an enhanced representation of the data sample. Then, this representation is passed through the **Fake news detection module** to determine whether the post is fake. For testing, we use the **Cold user behavior mapper module** to find the representation for cold users. Figure 2 represents the overall architecture of the model.

User History Module

We construct an undirected *global interaction-based graph* $G(U_G, E_G)$ where nodes are denoted by U_G and E_G represents the set of edges. We define $\mathbf{u_i} \in \mathbb{R}^{d_1}$ as a d_1-dimensional embedding representation for user $u_i \in U_G$. We use the `node2vec` (Grover and Leskovec 2016) model, which computes the node embedding by sampling its neighborhood and then minimizing the proximity loss function. Using this module, we represent user embedding, which captures historical evidence and global interactions between the users.

For this module, we consider nodes U_G as the set of users belonging to training data samples as shown in equation 1 and E_G as the set of edges representing the interactions between the users in U_G. We train the global graph using the `node2vec` (Grover and Leskovec 2016) model and generate the user embeddings of dimension 128 (d_1).

Content and Behavior Representation Module

We define $\mathbf{p_i} \in \mathbb{R}^{d_2}$ as the d_2-dimensional sentence embedding corresponding to post p_i. Similarly, $\mathbf{c_j(p_i)} \in \mathbb{R}^{d_2}$ is the d_2-dimensional sentence embedding corresponding to comment $c_j(p_i)$. Let $\mathbf{f_{p_i}}$ and $\mathbf{f_{c_j(p_i)}}$ be the representations for post p_i and comment $c_j(p_i)$, respectively, which aggregate both the textual content and user behavior features. These features are defined as follows.

$$\mathbf{f_{p_i}} = \mathbf{p_i} \,||\, \mathbf{u_p} \tag{2}$$

$$\mathbf{f_{c_j(p_i)}} = \mathbf{c_j(p_i)} \,||\, \mathbf{u_c} \tag{3}$$

where $\mathbf{u_p}$ and $\mathbf{u_c}$ represent the feature vectors for the users who created the post p_i and made the comment $c_j(p_i)$, respectively. The symbol $||$ is the concatenation operator.

We use the sentence transformer model (Reimers and Gurevych 2020) to represent each sentence in a 256 (d_2)

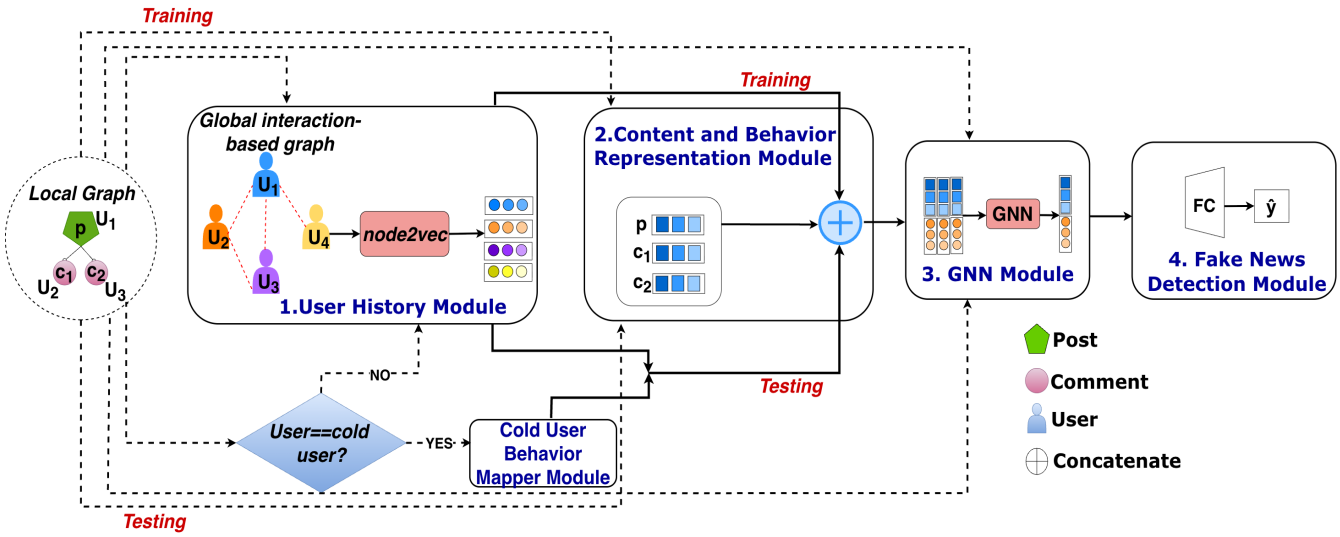


Figure 2: The overall architecture of the UEN framework. For **training**, first the *global interaction-based graph* is constructed and trained to generate user representation in the first module. These embeddings are passed to the second module, which captures content and user behavior features. These features are fed to the third module to obtain a robust graph representation, which is ultimately classified in the fourth module. For **testing**, each user in the sample is first checked to determine if they are a cold user. Cold user representation is obtained from the cold user behavior mapper module (shown in Figure 3); while other users' representation is obtained from the first module. Remaining steps follow the training phase.

dimensional representation. First, we use BERT to create 768 dimensional word embeddings. Then, we combine these word embeddings into a sentence embedding using a pooling layer. Here, we apply a $\tanh$ activation function and use a dense layer with 256 output features (i.e., d_2). Now, we concatenate the text and user embeddings and use this 384-dimensional embedding to represent the nodes that are further fed to the GNN module.

GNN Module

We consider the following standard GNN models for our experiments. These are GCN (Kipf and Welling 2017), GRAPHSAGE (Hamilton, Ying, and Leskovec 2018) and GAT (Veličković et al. 2018). For all these models we consider three layers ($l = 3$). The dimensions of each node's hidden feature vectors are 64. The activation function used is RELU. Now, we consider a graph L_i for each data sample d_i, with nodes represented by $\mathbf{f_{p_i}}$, $\mathbf{f_{c_j(p_i)}}$ and E_i representing the edges. The features obtained from the above module and the graph L_i are fed to GNN to learn graph-level representations. Let $\mathbf{l_{p_i}}$ and $\mathbf{l_c}$ denote the embeddings of the post and comment nodes, respectively, obtained from the GNN. The final graph-level representation $\mathbf{L_i}$ for the data sample d_i is computed as shown in the equation 4.

$$\mathbf{L_i} = \lambda * \mathbf{l_{p_i}} + (1 - \lambda) * \frac{1}{|C(p_i)|} \sum_{c \in C(p_i)} \mathbf{l_c} \qquad (4)$$

λ is obtained using hyperparameter optimization framework `optuna`[2] by minimizing the validation loss.

[2]https://optuna.org/

Fake News Detection Module

The final representation $\mathbf{L_i}$ obtained for each data sample d_i is then passed through a fully connected layer to classify the sample as either fake (F) or true (T). The model is trained using Adam optimizer with a learning rate of 0.01. The loss function utilized in our model is the cross-entropy loss.

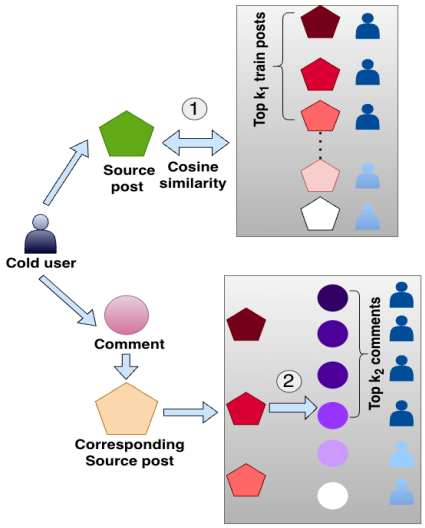
Cold User Behavior Mapper Module

Instead of random initialization, we propose a few heuristics to extract robust representation for cold users based on available data for more accurate approximation of the user behavior. These heuristics are built upon the collaborative filtering approach used in the recommendation system (Su and Khoshgoftaar (2009)).

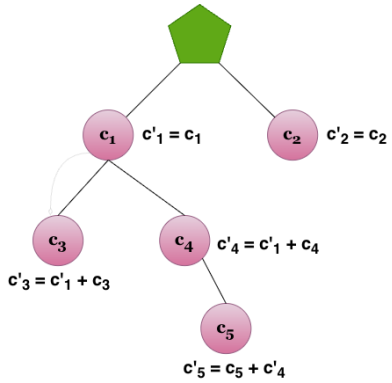
1. **Post similarity**: Users with similar interest and preferences tend to post comparable content.
2. **Reaction similarity**: Users often react similarly for similar content. This heuristic focuses on aligning user reactions to content.
3. **Historical reaction similarity**: The way users react depends on the historical context of the comments chain. This takes into account the sequence and context of previous reactions to infer the user's current behavior.

We apply the above three heuristics at different levels to find a representation to cold users. Figure 3 represents our approach to represent the cold users. In this figure, dark colors are used for posts (comments) to represent higher similarity with the current post (comment), while light colors indicate lower similarity.

- **Representation of cold users at the source post**: We utilize the text embedding of the cold user's source post,



(a) ①,② represent post similarity and reaction similarity heuristics respectively. In this example, we consider $k_1 = 3$ and $k_2 = 4$. The cold user is represented by the average of user embeddings of train users represented in blue.



(b) Representation of comments for comparison, following the ③ historical reaction similarity. Each comment is represented as the sum of text embeddings from first-level comment to the current comment along the comment chain.

Figure 3: Cold user behavior mapper module.

then compute the cosine similarity with the text embeddings of posts from the training dataset commensurate with the first heuristic. Now, we identify the top k_1 posts which are most similar and represent the cold user with the average of user embeddings of k_1 train users who posted these most similar posts. This is illustrated in Figure 3(a) where we consider $k_1 = 3$.

- **Representation of cold users who commented**: We first use the source post on which the comment of the cold user (c_{cold}) is made and find the top k_1 similar posts to this source post following the approach just mentioned above. We now collect all the comments to these k_1 posts (call this set $\mathcal{C}_{collect}$). We first represent c_{cold} as well as each of the comment in $\mathcal{C}_{collect}$ as the sum of the text embeddings from the first level comment (i.e., the one on the post directly) to the current comment (i.e., c_{cold} or one of the members of $\mathcal{C}_{collect}$) along the comment chain as shown in the Figure 3(b), commensurate with the third heuristic. This transformation in c_{cold} as well as the members of $\mathcal{C}_{collect}$ enables them to incorporate the characteristics of all the comments appearing before them in their respective chains. Now we compute the cosine similarity of the transformed c_{cold} with each of the transformed member of $\mathcal{C}_{collect}$ and return the top k_2 similar comments which is commensurate with the second heuristic. Finally, we represent the cold user as the average of the user embeddings of k_2 train users who made those top k_2 similar comments. This is illustrated in Figure 3(a) where we consider $k_2 = 4$.

The values of k_1 and k_2 are obtained by optimizing the valid loss function. We used `faiss`[3] for efficient similarity search. Note that, in this approach, the same user appearing multiple times in a particular data sample will be assigned different representations depending on the specific context of each occurrence.

Comparision with Other Methods

We choose three baselines for comparison with UEN. The first is the benchmark model proposed in the original paper introducing the Fakeddit dataset (Nakamura, Levy, and Wang 2020). In addition, we compare UEN with a recent method proposed in (Zhu et al. 2024), as well as with LLM-based baselines.

Nakamura, Levy, and Wang (2020): To generate fixed-length BERT embedding vectors, the authors employ the bert-as-service tool (Xiao 2018) for converting the variable-length text or sentences into a 768-dimensional array for each Reddit submission title. In their experiments, they use the pre-trained `bert-large-uncased` model. They pass these embeddings through a trainable dense layer to obtain the final classification output. For our purpose, we reproduce the same setup.

Zhu et al. (2024): We compare our UEN framework with **PSGT** – Propagation Structure-Aware Graph Transformer (Zhu et al. 2024), – a graph transformer based approach (Min et al. 2022) that enhances reliability and interpretability by filtering out noisy information, focusing on task-relevant features, and simultaneously capturing long-range structural dependencies. The key difference of this model from graph transformer is that in place of multi-headed self attention, it uses graph-sampled multi-headed self attention. The authors employ three distinct graph masks – a noise-filtered mask graph, designed to filter out noisy information among users; a structural positional encoding mask, aimed at capturing propagation depth and distance relationships between users; and a third mask focused on learning node interrelations while avoiding any structural biases. The input to the model has two parts – the source news and the engaged users who either commented or replied to

[3] https://github.com/facebookresearch/faiss

Dataset	Hyperparameters	GCN	GRAPHSAGE	GAT
Fakeddit	λ	0.62	0.89	0.87
	k_1	19	28	11
	k_2	72	96	51
Gossipcop	k	39	4	11

Table 3: Hyperparameters used for each of the GNN models.

Dataset	Method	GCN	GRAPHSAGE	GAT
Fakeddit	UEN $_{w/o\ \text{user}}$	0.77 / 0.73	0.81 / 0.77	0.79 / 0.75
	UEN $_{w/o\ \text{mapper}}$	0.85 / 0.81	0.87 / 0.83	0.86 / 0.81
	UEN	**0.87** / **0.84**	**0.87** / **0.85**	**0.87** / **0.83**
Gossipcop	UEN $_{w/o\ \text{user}}$	0.89 / 0.89	0.91 / 0.91	0.92 / 0.92
	UEN $_{w/o\ \text{mapper}}$	0.92 / 0.92	0.94 / 0.94	0.95 / 0.95
	UEN	**0.92** / **0.92**	**0.94** / **0.94**	**0.95** / **0.95**

Table 4: Evaluation of the UEN frameworks. Each cell shows Accuracy / Macro-F1. The variant with the best metric values are noted in **boldface**. Best accuracy is highlighted in blue and best macro-F1 is highlighted in orange.

the news and their interaction patterns (similar to our input format). They use the pre-trained BERT model to extract the initial features from this input.

LLM bases baseline: We prompt LLMs for fake news detection using the same approach mentioned in (Nan et al. 2025) to compare our model. Specifically, we use the instruction fine-tuned LLaMA 8B model checkpoint `meta-llama/Llama-3.1-8B-Instruct`, Qwen 8B model checkpoint `Qwen/Qwen3-8B` and GPT-OSS 20B model checkpoint `openai/gpt-oss-20b` for our experiments. We consider two variants of the LLM: LLM_{cnt}, which takes only the source content as input, and $\text{LLM}_{\text{cnt+cmt}}$, which incorporates both the source content and associated comments.

Results

Overall Performance

We evaluate the performance for the following variants of the framework using the three GNN models: GCN, GRAPHSAGE, and GAT. We report accuracy and a macro-averaged F1-Score.

- UEN: In this framework, we follow the same steps mentioned in the previous section.
- $\text{UEN}_{w/o\ \text{mapper}}$: For the cold users, we initialize them with the average of all user embeddings in the *global interaction-based graph*. Hence, in this framework, the cold user behavior mapper module is not utilized.
- $\text{UEN}_{w/o\ \text{user}}$: Here, we ignore the user features and represent every node using only the textual content, i.e. $\mathbf{f_{p_i}} = \mathbf{p_i}$ and $\mathbf{f_{c_j(p_i)}} = \mathbf{c_j(p_i)}$

Model	Fakeddit (p-value)	Gossipcop (p-value)
GCN	1.8342×10^{-7}	2.1947×10^{-54}
GRAPHSAGE	7.0411×10^{-96}	3.284×10^{-19}
GAT	4.6118×10^{-12}	6.2534×10^{-90}

Table 5: p-values comparing the prediction outcomes of $\text{UEN}_{w/o\ \text{mapper}}$ with UEN using the Mann-Whitney U test.

Category	Fakeddit			Gossipcop		
	Total	True	Fake	Total	True	Fake
Total	116,639	86,443	30,196	1,094	557	537
(0.5,1]	31,327	23,080	8,247	838	509	329
(0,0.5]	63,915	50,758	13,157	240	44	196
0	21,397	12,605	8,792	16	4	12

Table 6: Distribution of test samples based on overlap ratio between the users in the sample and the *global interaction-based graph*. Even the last row, representing the existence of solely *cold users*, has a significant number of samples for Fakeddit.

Table 3 displays the hyperparameters selected for our framework. The models have been trained on a NVIDIA A100 40GB GPU and implemented by `Pytorch`. Most models completed the training within a day. We trained the models for 20 epochs. Table 4 shows the performance of different variants of the proposed framework.

Importance of User Evidence

Table 4 clearly shows the advantage of including the user behavior based features ($\text{UEN}_{w/o\ \text{mapper}}$) over just using text based features ($\text{UEN}_{w/o\ \text{user}}$). For the fakeddit dataset, the accuracy improves by 8%, 6% and 7% for the models GCN, GRAPHSAGE, and GAT, respectively. Similarly, on the Gossipcop dataset, the inclusion of user evidence leads to a 3% improvement in accuracy. Hence, the user evidence plays an important role in fake news detection. The complete UEN framework including the mapper module produces the best results across all the three GNN models and for both the metrics. Table 5 shows the p-values comparing the prediction outcomes of $\text{UEN}_{w/o\ \text{mapper}}$ with UEN using the Mann-Whitney U test. For all the GNN models the prediction outcomes are significantly different demonstrating that the macro-F1 improvements from $\text{UEN}_{w/o\ \text{mapper}}$ to UEN are not by chance.

Investigation of Cold Users

Fakeddit dataset contains approximately 1.5 million users, of which 430,439 are cold users. Among all users, 11% engage in fake posts per user (general), while cold users contribute 7% engagement in fake posts per cold user. This demonstrates that cold users contribute a considerable vol-

Category	UEN$_{w/o\ user}$	UEN$_{w/o\ mapper}$	UEN
GCN			
(0.5,1]	0.83 / 0.79	0.93 / 0.91	0.93 / 0.92
(0,0.5]	0.78 / 0.72	0.89 / 0.84	0.89 / 0.84
0	0.66 / 0.65	0.63 / 0.61	**0.72** / **0.71**
GRAPHSAGE			
(0.5,1]	0.86 / 0.83	0.93 / 0.92	0.93 / 0.91
(0,0.5]	0.83 / 0.77	0.90 / 0.86	0.88 / 0.84
0	0.69 / 0.68	0.69 / 0.66	**0.76** / **0.75**
GAT			
(0.5,1]	0.84 / 0.81	0.93 / 0.91	0.93 / 0.90
(0,0.5]	0.80 / 0.75	0.89 / 0.84	0.89 / 0.83
0	0.68 / 0.67	0.65 / 0.62	**0.73** / **0.71**

Table 7: Results for **Fakeddit** dataset, based on overlap ratio based buckets. The variant with the best metric values for the zero overlap buckets are noted in **boldface**. Best accuracy for this bucket is highlighted in blue and best macro-F1 for this bucket is highlighted in orange.

ume to the overall fake post engagement. This observation highlights the importance of carefully assessing model performance in the presence of cold users. To better understand the challenge they pose, we divide the test data into buckets based on the extent of overlap between the users in the test data and the *global interaction-based graph*. In particular we compute the overlap ratio as the number of unique users in the test sample who also exist in the *global interaction-based graph* to the total number of unique users in the test sample. Based on this value we divide the test data into three buckets as follows.

1. $(0.5, 1]$: Samples with a user overlap ratio greater than 50%. This category represents scenarios where the model is relatively familiar with the majority of the users in the test dataset.
2. $(0, 0.5]$: Samples with a user overlap ratio less than 50% but not zero. This category includes scenarios where the model has less prior knowledge of the users in the test dataset.
3. (0): Samples with no user overlap. This category represents the most challenging scenario, where the model encounters cold users who have no prior footprint on the platform. Analyzing this category is crucial for evaluating the model's robustness and its ability to generalize to completely new user data. For the Gossipcop dataset we omitted this set for our experiments as it has a very small number of instances.

Table 6 shows the distribution of samples in each category in the test dataset. We observe that there is a substantial number of cold users in the test dataset. The performance of the framework UEN $_{w/o\ mapper}$ across the three categories is reported in Table 7 and Table 8. We observe that the framework performs better in the first bucket, achieving 93% accuracy irrespective of the GNN model used for the Fakeddit dataset, and $95 - 96\%$ for the Gossipcop dataset. In the second bucket, there is a slight drop in performance on Fakeddit, with accuracies of 89%, 90%, and 89% for GCN, GRAPHSAGE, and GAT, respectively. For Gossipcop, the drop is more pronounced, with a reduction of approximately $5 - 10\%$ in accuracy and $11 - 21\%$ in macro-F1. For the third category (applicable only to Fakeddit, as the number of samples is insufficient in Gossipcop), we observe a significant decline in performance, with accuracy dropping by approximately $20 - 25\%$. This clearly demonstrates that the presence of cold users adversely affects model performance. However, this impact is mitigated when using the full UEN framework. Compared to the UEN $_{w/o\ mapper}$ variant, the accuracy improves by 9%, 7%, and 8% for GCN, GRAPHSAGE, and GAT, respectively, on Fakeddit. The macro-F1 score also increases by $9 - 10\%$. Similarly, for the second category on the Gossipcop dataset, the full framework leads to a 2% increase in accuracy and improvements of 1%, 6%, and 2% in macro-F1 for GCN, GRAPHSAGE, and GAT, respectively.

Category	UEN$_{w/o\ user}$	UEN$_{w/o\ mapper}$	UEN
GCN			
(0.5,1]	0.92 / 0.91	0.95 / 0.94	0.95 / 0.94
(0,0.5]	0.80 / 0.70	0.85 / 0.78	**0.85** / **0.79**
GRAPHSAGE			
(0.5,5]	0.94 / 0.93	0.96 / 0.96	0.96 / 0.96
(0,0.5]	0.83 / 0.73	0.88 / 0.75	**0.90** / **0.81**
GAT			
(0.5,1]	0.94 / 0.94	0.96 / 0.96	0.96 / 0.96
(0,0.5]	0.86 / 0.76	0.91 / 0.85	**0.93** / **0.87**

Table 8: Results for **Gossipcop** dataset, based on overlap ratio based buckets. The variant with the best metric values for the (0,0.5] category are noted in **boldface**. Best accuracy for this bucket is highlighted in blue and best macro-F1 for this bucket is highlighted in orange.

Ablation study on heuristics used in the cold user mapper module

Fakeddit: Table 9 presents the accuracy and macro-F1 scores obtained when different combinations of heuristics are used in the Cold User Mapper Module. The heuristics considered are: (H1) Post Similarity, (H2) Reaction Similarity, and (H3) Historical Reaction Similarity.

For each combination of heuristics, we report the best results achieved. As seen in the table, different combinations contribute variably to performance, and including all heuristics

Category	UEN$_{H1}$	UEN$_{H1+H2}$	UEN$_{H1+H2+H3}$
GCN			
Overall	0.86 / 0.82	0.86 / 0.83	0.87 / 0.84
(0.5,1]	0.93 / 0.92	0.93 / 0.92	0.93 / 0.92
(0,0.5]	0.88 / 0.85	0.89 / 0.83	0.89 / 0.84
0	0.67 / 0.64	0.71 / 0.70	**0.72** / **0.71**
GRAPHSAGE			
Overall	0.87 / 0.83	0.87 / 0.83	0.87 / 0.85
(0.5,1]	0.93 / 0.91	0.93 / 0.91	0.93 / 0.91
(0,0.5]	0.89 / 0.84	0.88 / 0.82	0.88 / 0.84
0	0.71 / 0.69	0.75 / 0.74	**0.76** / **0.75**
GAT			
Overall	0.86 / 0.82	0.87 / 0.82	0.87 / 0.83
(0.5,1]	0.93 / 0.91	0.93 / 0.91	0.93 / 0.90
(0,0.5]	0.88 / 0.84	0.88 / 0.82	0.89 / 0.83
0	0.68 / 0.65	0.71 / 0.70	**0.73** / **0.71**

Table 9: Results for **Fakeddit** dataset, based on overlap ratio based buckets. The variant with the best metric values for the (0) category are noted in **boldface**. Best accuracy for this bucket is highlighted in blue and best macro-F1 for this bucket is highlighted in orange.

typically yields the highest scores.

Gossipcop: Table 10 presents the accuracy and macro-F1 scores for two heuristics: (H1) Post Similarity and (H2) Reaction Similarity. In the case of the GossipCop dataset, each source news item is unique and accompanied by associated tweets and retweets. Therefore, H1 is not applicable, as there are no repeated posts to compare for similarity. For this reason, we have excluded H1 from the evaluation on Gossipcop and focused solely on H2. As shown in the table, H2 alone performs well on this dataset.

Parameter sensitivity: To evaluate the sensitivity of our models to the key parameter k_1, we computed the mean and standard deviation of the validation accuracy for each setting on the Fakeddit dataset. The results are summarized in Table 11. We observe that all three models show stable performance across different k_1 values, indicating that our results are robust with respect to this parameter.

Baseline Results

Nakamura, Levy, and Wang (2020):We recompute the results on our dataset (see Table 2) using text-only features and the same hyperparameter settings as in their paper, and obtain an accuracy of 77% for Fakeddit and 73% for Gossipcop. For Fakeddit, this performance is comparable to that of UEN$_{w/o\ \text{user}}$, which also uses text-only features. However, for Gossipcop, the performance is worse.

Category	UEN$_{H1}$	UEN$_{H2}$
GCN		
Overall	0.92 / 0.92	0.92 / 0.92
(0.5,1]	0.95 / 0.94	0.95 / 0.94
(0,0.5]	0.84 / 0.76	**0.85** / **0.79**
GRAPHSAGE		
Overall	0.94 / 0.93	0.94 / 0.94
(0.5,1]	0.96 / 0.96	0.96 / 0.96
(0,0.5]	0.88 / 0.79	**0.90** / **0.81**
GAT		
Overall	0.95 / 0.95	0.95 / 0.95
(0.5,1]	0.96 / 0.96	0.96 / 0.96
(0,0.5]	0.92 / 0.86	**0.93** / **0.87**

Table 10: Results for **Gossipcop** dataset, based on overlap ratio based buckets. The variant with the best metric values for the (0,0.5] category are noted in **boldface**. Best accuracy for this bucket is highlighted in blue and best macro-F1 for this bucket is highlighted in orange.

Model	$k_1 = 5$	$k_1 = 10$	$k_1 = 15$
GCN	74.88 ± 0.33	76.54 ± 0.45	76.63 ± 0.42
GraphSAGE	75.84 ± 0.48	77.02 ± 0.50	77.29 ± 0.53
GAT	77.29 ± 0.25	77.89 ± 0.34	77.63 ± 0.29

Table 11: Validation accuracy (%) of different models for varying k_1 values on the Fakeddit dataset. Values are reported as mean ± standard deviation.

Zhu et al. (2024): We train the PSGT model for 20 epochs, initialising the source post and engaged users with the text features obtained in the Content and Behavior Representation module. Table 12 presents the results of the comparison between the UEN framework and PSGT. Our findings indicate that the UEN framework achieves a 6% higher accuracy compared to PSGT for both the datasets. Notably, for the zero-overlap case, UEN reports 9% improvement (GRAPHSAGE) over PSGT for the Fakeddit dataset and in $(0, 0.5]$ case, UEN reports 13% improvement (GAT) for the Gossipcop dataset.

LLM-based baselines: Both LLM variants perform worse than our model. Overall, for the Fakeddit dataset our model achieves 24% and 14% higher accuracy compared to the two LLM baselines, respectively. For zero-overlap cases, it outperforms the LLMs by 21% and 20%, respectively. We are unable to report results for the Gossipcop dataset, as the raw text is not available.

Dataset	Category	UEN			PSGT	Llama		Qwen		GPT-OSS	
		GCN	GraphSage	GAT		LLM_{cnt}	$LLM_{cnt+cmt}$	LLM_{cnt}	$LLM_{cnt+cmt}$	LLM_{cnt}	$LLM_{cnt+cmt}$
Fakeddit	Overall	**0.87** / 0.84	**0.87** / **0.85**	**0.87** / 0.83	**0.81** / **0.76**	**0.63** / **0.52**	**0.73** / 0.54	0.60 / 0.51	0.65 / **0.55**	0.61 / 0.45	0.64 / 0.44
	(0.5,1]	0.93 / 0.92	0.93 / 0.91	0.93 / 0.90	0.86 / 0.83	0.63 / 0.52	0.73 / 0.53	0.62 / 0.55	0.67 / 0.59	0.60 / 0.44	0.64 / 0.43
	(0,0.5]	0.89 / 0.84	0.88 / 0.84	0.89 / 0.83	0.83 / 0.76	0.68 / 0.52	0.79 / 0.57	0.60 / 0.52	0.65 / 0.57	0.65 / 0.46	0.69 / 0.45
	0	0.72 / 0.71	**0.76** / **0.75**	0.73 / 0.71	**0.67** / **0.64**	0.54 / **0.48**	**0.56** / 0.50	**0.55** / **0.48**	**0.56** / **0.51**	0.51 / 0.42	0.51 / 0.39
Gossipcop	Overall	0.92 / 0.92	0.94 / 0.94	**0.95** / **0.95**	**0.89** / **0.89**	-	-	-	-	-	-
	(0.5,1]	0.95 / 0.94	0.96 / 0.96	0.96 / 0.96	0.93 / 0.92	-	-	-	-	-	-
	(0,0.5]	0.85 / 0.79	0.90 / 0.81	**0.93** / **0.87**	**0.80** / **0.63**	-	-	-	-	-	-

Table 12: Comparison of results of **UEN** with other baselines. Each cell shows Accuracy / Macro-F1. Best accuracy is highlighted in blue and best macro-F1 is highlighted in orange.

Analysis of the Results

We analysed the performance of UEN further on *zero overlap* cases in the Fakeddit dataset based on the size and depth of the propagation tree. We also conducted analyses based on token length and subreddit, but no clear patterns observed in those dimensions.

Based on tree size: The tree size (number of comments, reactions, and source posts) within the bucket of size [1–10] accounts for 80% of the data in this dataset. In this bucket, both PSGT and our model performed equivalently, achieving an accuracy of 78%. For the remaining 20% of the samples, where the tree size is greater than 10, PSGT's accuracy dropped to the range of $14 - 17\%$, while our model (GAT) maintained an accuracy between $60 - 70\%$.

Based on depth: Depth-based analysis revealed that accuracy of PSGT model was 32% at depth 3, while our model achieved 68%. The drop in PSGT performance at depth 3 can be attributed to the fact that 74% of the data at this depth had tree size greater than 10.

Conclusion

In this paper, we first show the importance of content from posts, comments, and user behavior from the user-user interactions in fake news detection task. Then, we showed that there is a significant drop in performance for real-world datasets due to the presence of cold users. Next, we proposed UEN framework capable of approximating the behavior of cold users. Our carefully crafted heuristics enables us to significantly improve the detection performance especially when the cold users have absolutely no footprint in the training data. Our approach is generic and can be easily extended to other similar datasets.

Ethics Statement

This study does not aim to identify or trace individual users involved in the dissemination of fake news. Our intent is not to harm any individuals or target specific communities. All experiments were conducted on publicly available or previously published datasets. The primary objective of this research is to enhance model generalization, particularly for scenarios involving cold-start users.

References

Allcott, H.; and Gentzkow, M. 2017. Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives*, 31: 211–236.

Ayub, M.; Ghazanfar, M. A.; Mehmood, Z.; Alyoubi, K. H.; and Alfakeeh, A. S. 2020. Unifying user similarity and social trust to generate powerful recommendations for smart cities using collaborating filtering-based recommender systems. *Soft Comput.*, 24(15): 11071–11094.

Bian, T.; Xiao, X.; Xu, T.; Zhao, P.; Huang, W.; Rong, Y.; and Huang, J. 2020. Rumor Detection on Social Media with Bi-Directional Graph Convolutional Networks. arXiv:2001.06362.

Castillo, C.; Mendoza, M.; and Poblete, B. 2011. Information credibility on twitter. In *The Web Conference*.

Del Vicario, M.; Bessi, A.; Zollo, F.; Petroni, F.; Scala, A.; Caldarelli, G.; Stanley, H.; and Quattrociocchi, W. 2016. The spreading of misinformation online. *Proceedings of the National Academy of Sciences*, 113.

FORCE11. 2020. The FAIR Data principles. https://force11.org/info/the-fair-data-principles/.

Gebru, T.; Morgenstern, J.; Vecchione, B.; Vaughan, J. W.; Wallach, H.; Iii, H. D.; and Crawford, K. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92.

Gong, Q.; Chen, Y.; He, X.; Xiao, Y.; Hui, P.; Wang, X.; and Fu, X. 2021. Cross-site Prediction on Social Influence for Cold-start Users in Online Social Networks. *ACM Trans. Web*, 15(2).

Gong, S.; Sinnott, R. O.; Qi, J.; and Paris, C. 2023. Fake News Detection Through Graph-based Neural Networks: A Survey. arXiv:2307.12639.

Grinberg, N.; Joseph, K.; Friedland, L.; Swire-Thompson, B.; and Lazer, D. 2019. Fake news on Twitter during the 2016 U.S. presidential election. *Science*, 363: 374–378.

Grover, A.; and Leskovec, J. 2016. node2vec: Scalable Feature Learning for Networks. arXiv:1607.00653.

Hamilton, W. L.; Ying, R.; and Leskovec, J. 2018. Inductive Representation Learning on Large Graphs. arXiv:1706.02216.

Han, Y.; Karunasekera, S.; and Leckie, C. 2020. Graph neural networks with continual learning for fake news detection from social media. *arXiv preprint arXiv:2007.03316*.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR)*.

Min, E.; Chen, R.; Bian, Y.; Xu, T.; Zhao, K.; Huang, W.; Zhao, P.; Huang, J.; Ananiadou, S.; and Rong, Y. 2022. Transformer for Graphs: An Overview from Architecture Perspective. arXiv:2202.08455.

Miyazaki, K.; Uchiba, T.; Tanaka, K.; An, J.; Kwak, H.; and Sasahara, K. 2023. " This is Fake News": Characterizing the Spontaneous Debunking from Twitter Users to COVID-19 False Information. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 17, 650–661.

Nakamura, K.; Levy, S.; and Wang, W. Y. 2020. r/Fakeddit: A New Multimodal Benchmark Dataset for Fine-grained Fake News Detection. arXiv:1911.03854.

Nan, Q.; Sheng, Q.; Cao, J.; Zhu, Y.; Wang, D.; Yang, G.; and Li, J. 2025. Exploiting user comments for early detection of fake news prior to users' commenting. *Front. Comput. Sci.*, 19(10).

Park, S.-T.; and Chu, W. 2009. Pairwise preference regression for cold-start recommendation. In *Proceedings of the third ACM conference on Recommender systems*, 21–28.

Rashkin, H.; Choi, E.; Jang, J. Y.; Volkova, S.; and Choi, Y. 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*, 2931–2937.

Reimers, N.; and Gurevych, I. 2020. Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Schein, A. I.; Popescul, A.; Ungar, L. H.; and Pennock, D. M. 2002. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 253–260.

Shu, K.; Wang, S.; and Liu, H. 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*, WSDM '19, 312–320. New York, NY, USA: Association for Computing Machinery. ISBN 9781450359405.

Soliman, A.; Hafer, J.; and Lemmerich, F. 2019. A Characterization of Political Communities on Reddit. In *Proceedings of the 30th ACM Conference on Hypertext and Social Media*, HT '19, 259–263. New York, NY, USA: Association for Computing Machinery. ISBN 9781450368858.

Su, X.; and Khoshgoftaar, T. 2009. A Survey of Collaborative Filtering Techniques. *Adv. Artificial Intellegence*, 2009.

Su, X.; Yang, J.; Wu, J.; and Qiu, Z. 2023. Hy-DeFake: Hypergraph Neural Networks for Detecting Fake News in Online Social Networks. *arXiv preprint arXiv:2309.02692*.

Sun, L.; Rao, Y.; Lan, Y.; Xia, B.; and Li, Y. 2023. Hg-sl: Jointly learning of global and local user spreading behavior for fake news early detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 5248–5256.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. arXiv:1710.10903.

Vosoughi, S.; Roy, D.; and Aral, S. 2018. The spread of true and false news online. *Science*, 359: 1146–1151.

Wei, L.; Hu, D.; Zhou, W.; and Hu, S. 2024a. Transferring Structure Knowledge: A New Task to Fake News Detection towards Cold-Start Propagation. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8045–8049. IEEE.

Wei, L.; Hu, D.; Zhou, W.; and Hu, S. 2024b. Transferring Structure Knowledge: A New Task to Fake News Detection towards Cold-Start Propagation. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8045–8049.

Xiao, H. 2018. bert-as-service. https://github.com/hanxiao/bert-as-service.

Xu, S.; Liu, X.; Ma, K.; Dong, F.; Riskhan, B.; Shunzhi, X.; and Bing, C. 2022. Rumor detection on social media using hierarchically aggregated feature via graph neural networks. *Applied Intelligence*, 53.

Xu, S.; Liu, X.; Ma, K.; Dong, F.; Riskhan, B.; Xiang, S.; and Bing, C. 2023. Rumor detection on social media using hierarchically aggregated feature via graph neural networks. *Applied Intelligence*, 53(3): 3136–3149.

Yang, F.; Liu, Y.; Yu, X.; and Yang, M. 2012. Automatic detection of rumor on Sina Weibo. In *MDS '12*.

Yuan, C.; Ma, Q.; Zhou, W.; Han, J.; and Hu, S. 2019a. Jointly embedding the local and global relations of heterogeneous graph for rumor detection. arXiv:1909.04465.

Yuan, C.; Ma, Q.; Zhou, W.; Han, J.; and Hu, S. 2019b. Jointly embedding the local and global relations of heterogeneous graph for rumor detection. In *2019 IEEE international conference on data mining (ICDM)*, 796–805. IEEE.

Zhu, J.; Gao, C.; Yin, Z.; Li, X.; and Kurths, J. 2024. Propagation Structure-Aware Graph Transformer for Robust and Interpretable Fake News Detection. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, 4652–4663. New York, NY, USA: Association for Computing Machinery. ISBN 9798400704901.

Paper Checklist

1. For most authors...
   (a) Would answering this research question advance science without violating social contracts, such as violating privacy norms, perpetuating unfair profiling, exacerbating the socio-economic divide, or implying disrespect to societies or cultures? Yes
   (b) Do your main claims in the abstract and introduction accurately reflect the paper's contributions and scope? Yes, please see the sections Result and Discussion supporting the claims through rigorous experiments.
   (c) Do you clarify how the proposed methodological approach is appropriate for the claims made? Yes, it is discussed in the section UEN framework.
   (d) Do you clarify what are possible artifacts in the data used, given population-specific distributions? Yes, please see the section Dataset description.
   (e) Did you describe the limitations of your work? Yes.
   (f) Did you discuss any potential negative societal impacts of your work? NA
   (g) Did you discuss any potential misuse of your work? NA
   (h) Did you describe steps taken to prevent or mitigate potential negative outcomes of the research, such as data and model documentation, data anonymization, responsible release, access control, and the reproducibility of findings? NA
   (i) Have you read the ethics review guidelines and ensured that your paper conforms to them? Yes
2. Additionally, if your study involves hypotheses testing...
   (a) Did you clearly state the assumptions underlying all theoretical results? NA
   (b) Have you provided justifications for all theoretical results? NA
   (c) Did you discuss competing hypotheses or theories that might challenge or complement your theoretical results? NA
   (d) Have you considered alternative mechanisms or explanations that might account for the same outcomes observed in your study? NA
   (e) Did you address potential biases or limitations in your theoretical framework? NA
   (f) Have you related your theoretical results to the existing literature in social science? NA
   (g) Did you discuss the implications of your theoretical results for policy, practice, or further research in the social science domain? NA
3. Additionally, if you are including theoretical proofs...
   (a) Did you state the full set of assumptions of all theoretical results? NA
   (b) Did you include complete proofs of all theoretical results? NA
4. Additionally, if you ran machine learning experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? The code is available on Github.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? Yes, please see the section Results.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? NA
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? Yes
   (e) Do you justify how the proposed evaluation is sufficient and appropriate to the claims made? Yes. The results that we present sufficiently support the claim.
   (f) Do you discuss what is "the cost" of misclassification and fault (in)tolerance? NA
5. Additionally, if you are using existing assets (e.g., code, data, models) or curating/releasing new assets, **without compromising anonymity**...
   (a) If your work uses existing assets, did you cite the creators? Yes
   (b) Did you mention the license of the assets? NA
   (c) Did you include any new assets in the supplemental material or as a URL? NA
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? NA
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? NA
   (f) If you are curating or releasing new datasets, did you discuss how you intend to make your datasets FAIR (see FORCE11 (2020))? NA
   (g) If you are curating or releasing new datasets, did you create a Datasheet for the Dataset (see Gebru et al. (2021))? NA
6. Additionally, if you used crowdsourcing or conducted research with human subjects, **without compromising anonymity**...
   (a) Did you include the full text of instructions given to participants and screenshots? NA
   (b) Did you describe any potential participant risks, with mentions of Institutional Review Board (IRB) approvals? NA
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? NA
   (d) Did you discuss how data is stored, shared, and deidentified? NA