Fake News Detection using Graph Neural Networks

Krishna Yemmanuru[*]
Department of Computer Science
University of Rochester
Rochester, NY 14620
kyemmanu@ur.rochester.edu

Amisha Dubey[†]
Department of Data Science
University of Rochester
Rochester, NY 14620
adubey04@ur.rochester.edu

1 Introduction

In today's digital age, the rapid proliferation of information on social media platforms has significantly complicated the task of distinguishing authentic news from deceptive misinformation. This paper delves into the application of sophisticated machine learning strategies, specifically Graph Neural Networks (GNNs), to tackle this issue. Utilizing the UPFD dataset—a comprehensive collection featuring detailed propagation networks of both fake and real news items on Twitter, authenticated by reputable sources such as Politifact and Gossipcop—this study aims to decipher the complex diffusion patterns that characterize the spread of misinformation. The dataset includes rich node representations derived from nearly 20 million tweets, offering a substantial foundation for not only understanding but also effectively countering the dissemination of fake news. Through a methodical analysis, this research explores how GNNs can leverage these vast data points to detect and mitigate the impact of false information, providing insights into the dynamics of news spread in social media ecosystems.

2 Abstract

The prevalence of fake news on social media is a growing concern, posing significant challenges to information veracity online. This study employs Graph Neural Networks (GNNs) to detect and analyze the spread of fake news using the UPFD dataset, which contains verified propagation networks of fake and real news on Twitter. The dataset, verified by sources like Politifact and Gossipcop, encompasses detailed interactions from nearly 20 million tweets, providing a granular view of how information, both true and false, proliferates across social networks. By applying GNNs, this research highlights the potential of advanced machine learning techniques in recognizing and understanding misinformation patterns, thereby aiding in the development of more robust systems to combat fake news. The findings demonstrate the effectiveness of GNNs in parsing complex network structures and offer valuable insights into the mechanisms of news distribution on social media, suggesting pathways for both technological and informational interventions.

3 literature review

This section discusses various studies related to the development of graph neural networks and their specific application for fake news detection.

[*]
[†]

3.0.1 Graph Neural Networks with Continual Learning for Fake News Detection from Social Media

In their 2020 study, Yi Han, Shanika Karunasekera, and Christopher Leckie explore the application of Graph Neural Networks (GNNs) with a continual learning framework for detecting fake news on social media platforms. The authors address the challenge of evolving news topics and the dynamic nature of social media by implementing a model that adapts to new information without forgetting previous knowledge. Their methodology involves using a graph-based approach to model the relationships between news items and their propagation paths, which is crucial for understanding the spread of information. The paper highlights significant improvements in detection accuracy compared to traditional models, emphasizing the potential of GNNs in real-time, adaptive systems for misinformation management.

3.1 Semi-supervised Classification with Graph Neural Networks

Thomas N. Kipf and Max Welling's influential 2016 paper introduces a novel approach to semi-supervised classification using Graph Convolutional Networks (GCNs), a variant of neural networks that operate directly on graphs. The study provides a foundational understanding of graph convolution operations, illustrating how these can be leveraged to efficiently process data structured as graphs. Their model, which utilizes a localized first-order approximation of spectral graph convolutions, demonstrates how learning can be effectively generalized across connected nodes, thereby enhancing the performance on tasks like node classification. The authors validate their approach on multiple citation network datasets, where GCNs achieve state-of-the-art classification results, underscoring the technique's efficacy and scalability.

3.2 Fake news detection using Deep Learning

This paper explores the application of deep learning techniques to distinguish between fake and real news on the Internet solely based on text. The authors propose three different neural network architectures, one of which is based on BERT, a modern language model that achieves state-of-the-art results in natural language processing

3.3 Fake news detection using machine learning: an adversarial collaboration approach

This paper explores adversarial collaboration methods in machine learning to detect fake news. It focuses on adversarial behaviors, debiasing techniques, and the challenges and opportunities in safeguarding business datasets against adversarial attacks in the realm of fake news.

4 Current Methodologies

Besides Graph Neural Networks (GNNs), several other methodologies are employed in the detection of fake news. These methods vary in approach and technology, each with its own strengths and limitations. Here are some prominent ones

4.1 Fact-Checking Algorithms

Methodology: These algorithms automate the process of checking claims made in news articles against verified facts stored in databases or available on fact-checking websites like Snopes, PolitiFact, or FactCheck.org.

Limitations: The effectiveness of these systems is highly dependent on the availability and comprehensiveness of the fact databases they rely on. They may also struggle to keep pace with the rapid dissemination of new misinformation or fabricated stories that have not yet been documented.

4.2 Machine Learning and Deep Learning Models

Methodology: Traditional machine learning models such as decision trees, logistic regression, and support vector machines (SVM), as well as more complex deep learning models like convolutional neural networks (CNNs) and recurrent neural networks (RNNs), are used to classify news articles as fake or real based on features extracted from the text.

Limitations: These models require substantial labeled datasets for training, which can be difficult to procure. They also risk inheriting biases present in the training data, and their decision-making processes can lack transparency, making it difficult to interpret why certain decisions are made.

4.3 Propagation-Based Techniques

Methodology: These techniques analyze how information spreads through networks, identifying patterns typical of false news dissemination, such as rapid spreading or spreading primarily through certain controversial nodes.

Limitations: They require access to data about the network of users, which may not always be available due to privacy concerns, and they can mistakenly flag viral but true stories as fake.

5 Problem Statement

Graph Neural Networks (GNNs) offer promising solutions by leveraging the relational information within these networks. However, the challenge of developing models that can continually learn and adapt to new information without forgetting previously learned knowledge remains largely unaddressed. This project aims to investigate the efficacy of continual learning strategies integrated with GNNs to detect and mitigate the spread of fake news on social media. The goal is to enhance the adaptability and accuracy of fake news detection systems in real-time, ensuring they remain effective as the nature of the news and the network evolves.

5.1 Motivation for Using Graph Neural Networks in Fake News Detection

Graph Neural Networks (GNNs) are particularly well-suited for analyzing data from social media platforms, offering several distinct advantages over traditional machine learning and deep learning models:

1. **Capturing Relational Information**: GNNs excel in capturing the dynamic interactions and relationships between entities, crucial for understanding the social contexts within which news content circulates.
2. **Dynamic Incorporation of Context**: These networks effectively incorporate the context of nodes dynamically, making them adept at assessing the credibility of news based on the user network involved in its dissemination.
3. **Efficient Handling of Sparse Data**: GNNs are designed to efficiently process the sparse but structured data typical of social networks, focusing on existing connections rather than all possible interactions.
4. **Transferability Across Different Structures**: The ability to apply trained GNNs to different graph structures without complete retraining facilitates their deployment across various platforms and evolving network architectures.
5. **Scalability and Flexibility**: The scalability and flexibility of GNNs enable them to manage and analyze large-scale data from social media, incorporating various data types directly into the graph.
6. **Improved Performance for Structured Data**: Empirical evidence shows that GNNs outperform traditional models in tasks involving structured graph data, leveraging connections and node features to enhance predictive accuracy and performance.

6 Model

Graph neural networks (GNNs) detect fake news by leveraging the relational data inherent in social media networks, where nodes represent users or articles and edges signify interactions like retweets or shares. In practice, GNNs initiate by embedding articles and user profiles into high-dimensional

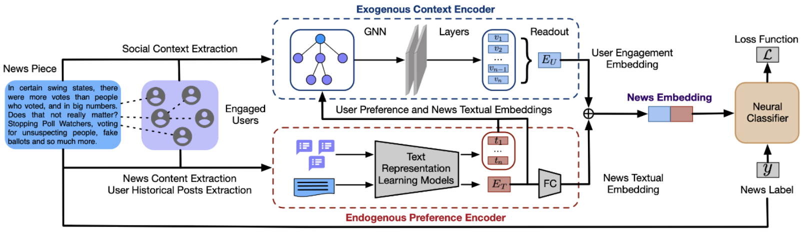

Figure 1: Example of a PNG image

space using pretrained models like BERT for textual content and node-specific metadata for user activities. The GNN layers then perform iterative aggregations of these features across the network's topology, where each node's updated state is a function of its neighbors' features, weighted by the edges connecting them.

This aggregation may incorporate attention mechanisms, allowing the model to learn which connections (e.g., user interactions) are most indicative of misinformation spread. For instance, a GNN can assign higher importance to edges connecting nodes frequently involved in spreading fake news. Once the network aggregates features through several layers, the final node embeddings capture not just individual article or user characteristics but also contextual information reflecting their broader network role. The proposed model architecture employs a sophisticated multi-component strategy to discern the authenticity of news articles by analyzing both their content and the surrounding social context. Below, we outline the key components of the model:

1. **Social Context Extraction**: This stage utilizes a Graph Neural Network (GNN) to derive embeddings that capture the dynamics of user interactions, such as likes and shares, providing insights into the social spread of news.

2. **Endogenous Preference Encoder**: Leveraging advanced text representation techniques, this encoder transforms news content into embeddings that reflect textual features, offering an understanding of the content independent of social interactions.

3. **Exogenous Context Encoder**: Integrates embeddings from user preferences and news content with those from social contexts, resulting in a comprehensive representation that encompasses both textual and relational information.

4. **Neural Classifier**: Receives combined embeddings and applies a classification mechanism to determine the news' authenticity, categorizing each piece as real or fake.

5. **Loss Function**: Utilized during training, this function optimizes the model to minimize discrepancies between predicted and actual labels, enhancing the detection accuracy.

For classification, these embeddings are fed into a softmax layer, predicting the likelihood of news being fake or real. The training process involves optimizing these predictions against a known dataset of labeled news items, using loss functions that penalize discrepancies between predicted and actual labels. By training on diverse examples of news propagation, GNNs learn to identify subtle patterns and structures typical of fake news dissemination, making them effective for real-time detection on evolving social media platforms.

Algorithm

Graph Neural Network Update Rule

The update rule in Graph Neural Networks (GNNs) is designed to integrate information from a node's neighbors within the graph to learn effective node representations. Below, we outline the components and the algorithmic steps involved in updating the node features in a GNN.

Components of the GNN Update Rule

- Node Feature Vector, $h_i^{(l)}$: Represents the features of node i at layer l.
- Neighbors, $N(i)$: The set of neighboring nodes of node i.
- Normalization Constant, c_{ij}: Used to normalize the influence of neighbor j on node i.
- Learnable Parameters: Weight matrix $W^{(l)}$ and bias vector $b^{(l)}$.
- Non-linear Activation Function, σ: Such as ReLU, applied to introduce non-linearity.

Update Rule Formula

The feature update for node i from layer l to $l+1$ is given by:

$$h_i^{(l+1)} = \sigma\left(\sum_{j\in N(i)} \frac{1}{c_{ij}} W^{(l)} h_j^{(l)} + b^{(l)}\right) \tag{1}$$

Algorithm Pseudocode

Algorithm 1 GNN Node Feature Update

Input: Graph $G(V, E)$, Initial node features $\{h_i^{(0)} \mid i \in V\}$, Depth L
Output: Node representations $h_i^{(L)}$ for all $i \in V$
for $l = 1$ to L **do**
    for each node $i \in V$ **do**
        $h_i^{(l)} \leftarrow \sigma\left(\sum_{j\in N(i)} \frac{1}{c_{ij}} W^{(l)} h_j^{(l-1)} + b^{(l)}\right)$
    end for
end for
return $\{h_i^{(L)} \mid i \in V\}$

7 Datasets

For our project, we have utilized the UPFD dataset. The dataset includes fake and real news propagation networks on Twitter built according to fact-check information from Politifact and Gossipcop. The news retweet graphs were originally extracted by FakeNewsNet. We crawled near 20 million historical tweets from users who participated in fake news propagation in FakeNewsNet to generate node features in the dataset.

Data	#Graphs	#Fake News	#Total Nodes	#Total Edges	#Avg. Nodes per Graph
Politifact	314	157	41,054	40,740	131
Gossipcop	5464	2732	314,262	308,798	58

Figure 2: Statistics of the dataset

7.1 Network Graph

The graph-based approach to detecting fake news on social media platforms is predicated on a sophisticated modeling of nodes and edges, which represent the entities and their interactions within the network.

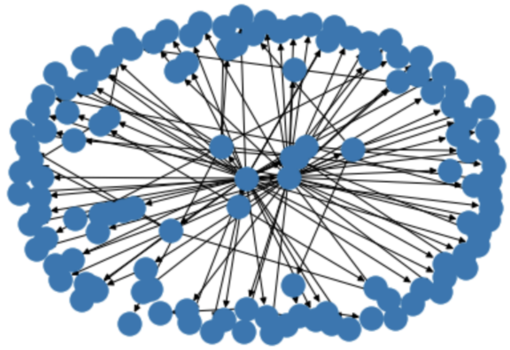

Figure 3: Statistics of the dataset

Nodes

- **User Nodes:** These nodes correspond to individual users who engage with news articles through activities such as posting, sharing, and commenting, each represented as a distinct node within the graph.
- **News Nodes:** In scenarios where the analysis focuses on the propagation patterns of specific articles, news articles themselves are also modeled as nodes. This allows for a granular analysis of information spread.

Edges

- **User-User Edges:** Represent social interactions among users, including followers/following dynamics and direct communications like retweets.
- **User-News Edges:** Capture the various interactions users have with news content, differentiated by type (e.g., posting, liking, sharing).
- **News-News Edges:** If news nodes are included, these edges depict thematic or contextual relationships between articles, such as shared topics or related events.

This structured representation of users and news content as nodes, connected by various types of edges, forms the backbone of our approach to understanding and mitigating the spread of fake news through social networks.

8 Model Training

Overview: The training methodology for our Graph Neural Network (GNN) leverages complex relational data from social networks and textual content from news articles. Implemented in a PyTorch-based framework and utilizing the `torch_geometric` library, our model is optimized for graph data processing, crucial for tasks such as fake news detection.

Model Structure and Initialization

Our GNN model consists of multiple Graph Attention Network (GAT) layers. These layers enable dynamic weighting of the importance of each node's neighbors, allowing the model to adapt to the varying influence of nodes across different contexts. Specifically, the model includes three GAT layers designed to capture deep relational patterns within the data. Additionally, a linear transformation layer adjusts the initial node features to suit the GAT layers' input requirements. The integration of node features from the graph, aggregated via global max pooling, with direct embeddings from news content, provides a holistic view of both the social and content-based elements of the news.

Forward Pass

1. **Graph Convolutions:** Node features are sequentially processed through three GAT layers, each followed by a ReLU activation to introduce non-linearity.
2. **Pooling:** Post convolution, node features are aggregated across each graph in the batch via global max pooling, resulting in a singular, comprehensive vector per graph.
3. **Readout:** The pooled graph-level representation is then enhanced by concatenating it with linearly transformed news content embeddings. This forms the comprehensive input for the final prediction.

Loss Computation and Optimization

The Binary Cross-Entropy (BCE) loss function is employed, appropriate for the binary classification nature of fake news detection. This function measures the discrepancy between the predicted labels and the actual labels, facilitating gradient computation. Model optimization is conducted using the Adam optimizer, with specific settings for the learning rate and weight decay to fine-tune performance during training.

Training Loop

The training loop operates as follows:

1. Each batch of data is loaded onto the designated computing device (GPU or CPU).
2. Before each forward pass, the optimizer's gradient buffers are cleared to ensure that gradient information from previous batches does not interfere.
3. A forward pass is executed with the batch data to compute the loss, followed by backpropagation to update model weights.
4. Loss values are aggregated across all batches to monitor and adjust the training process accordingly.

Performance Metrics

Post-training, the effectiveness of the model is evaluated using accuracy and F1 score metrics. These metrics are instrumental in assessing the model's capacity to accurately classify news articles as real or fake, considering the balance between precision and recall.

9 Simulation and Results

Our simulation shows 10 out 10 predictions being correctly made the trained GNN highlighting the successful convergence of the algorithm.

9.1 Metrics Table

Table 1: Performance Metrics of the GNN Model

Metric	Train Accuracy	Test Accuracy	Precision	Recall
Values	92.8%	95%	93%	88%
F1 Score	0.90			

9.2 Hyperparameter Sweep

We have utilized Wandb sweep to find the optimal set of parameters.

We have experimented with various batch sizes from 16 to 256, learning rates from 0.0001 to 0.01.

	pred_logit	pred	true
0	0.712935	1.0	1
1	0.546704	1.0	1
2	0.337851	0.0	0
3	0.572155	1.0	1
4	0.143253	0.0	0
5	0.337286	0.0	0
6	0.283256	0.0	0
7	0.805957	1.0	1
8	0.147197	0.0	0
9	0.272142	0.0	0

Figure 4: Statistics of the dataset

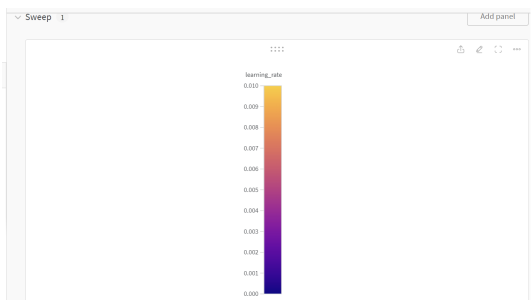

Figure 5: Orange zone indicates the optimal values and purple indicates undesirable values

10 Future Work

10.1 Enhancement of Model Architecture

The current architecture of our GNN has shown promise in initial tests; however, it requires further refinement to improve its efficacy in detecting fake news. One specific aspect that needs addressing is the integration of a dynamic learning component that allows the model to evolve with the ever-changing landscape of news. This will involve exploring advanced techniques in continual learning such as Elastic Weight Consolidation (EWC) or Progressive Neural Networks, which may provide the model with the ability to retain learned information and adapt to new data simultaneously.

10.2 Hyperparameter Optimization

To date, our model has been operating with an initial set of hyperparameters. The next stage will involve extensive hyperparameter tuning to optimize the model's performance. This will be accomplished using tools such as Weight Biases (wandb) to conduct a sweeping process across a range of hyperparameter configurations. This systematic approach is expected to identify the most effective combination that enhances the model's ability to discern complex patterns associated with misinformation.

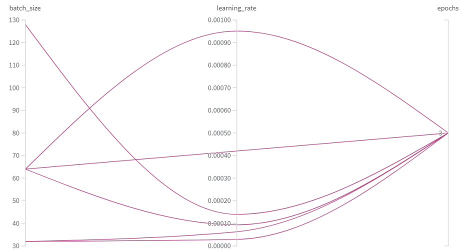

Figure 6: Orange zone indicates the optimal values and purple indicates undesirable values

10.3 Data Augmentation and Diversification

Although the UPFD dataset provides a solid foundation, it is crucial to incorporate additional datasets that include more varied and nuanced examples of fake and real news propagation. This expansion will likely improve the model's generalizability and robustness. Further data augmentation strategies will also be explored to artificially expand our dataset, providing the model with a richer learning experience.

10.4 Evaluation Metrics and Validation

We aim to introduce a more comprehensive set of evaluation metrics that go beyond accuracy, precision, and recall. These may include measures of robustness against adversarial attacks and the ability to detect deepfakes. Additionally, we will seek to validate our model through partnerships with fact-checking organizations to test its real-world efficacy.

11 References

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

Kipf, T. N., Welling, M. (2016). Semi-Supervised Classification with Graph Convolutional Networks. arXiv preprint arXiv:1609.02907.

Álvaro Ibrain Rodríguez, Lara Lloret Iglesias (2019) Fake news detection using Deep Learning

Karen M. DSouza, Aaron M. French (2017) Fake news detection using machine learning: an adversarial collaboration approach