Journal of Physics: Complexity

ACCEPTED MANUSCRIPT • OPEN ACCESS

Advancing Fake News Detection with Graph Neural Network and Deep Learning

To cite this article before publication: Haji Gul *et al* 2024 *J. Phys. Complex.* in press https://doi.org/10.1088/2632-072X/ad744d

Advancing Fake News Detection with Graph Neural Network and Deep Learning

Haji Gul[1], **Feras Al-Obeidat**[2], **Muhammad Wasim**[3], **Adnan Amin**[1*], **Fernando Moreira**[4]

[1]Center for Excellence in Information Technology, Institute of Management Sciences, Peshawar 25000, Pakistan
[2]Zayed University, Abu Dhabi 51133, UAE
[3] City University of Science and Information Technology, Peshawar 25000, Pakistan
[4]REMIT, IJP, Universidade Portucalense Porto, Portugal & IEETA, Universidade de Aveiro, Aveiro, Portugal

E-mail: [1]hajigul1993@gmail.com, [2]feras.al-obeidat@zu.ac.ae, [3]muhammadwasim443@gmail.com, [4]fmoreira@uportu.pt

E-mail: adnan.amin@imsciences.edu.pk *

Abstract. In the modern era of digital technology, the rapid distribution of news via social media platforms substantially contributes to the propagation of false information, presenting challenges in upholding the accuracy and reliability of information. This study presents an updated approach that utilizes Graph Neural Networks (GNNs) alongside with advanced deep learning techniques to improve the identification of false information. In contrast to traditional approaches that primarily rely on analyzing text and assessing the credibility of sources, our methodology utilizes the structural information of news propagation networks. This allows for a detailed comprehension of the interconnections and patterns that are indicative of misinformation. By analyzing the intricate, graph-based connections between news items, our approach not only overcomes the constraints of conventional fake news detection methods but also demonstrates significant enhancements in detection accuracy. This paper emphasizes the revolutionary nature of utilizing GNNs in the field of fake news detection. It also examines the potential consequences of our research in reducing the propagation of false information. Our model achieved an impressive accuracy rate of 97%, demonstrating a significant improvement in its ability to identify and classify fake news.The findings highlight the substantial improvement in the ability to detect fake news provided by GNNs in comparison to traditional methods, demonstrating promising growth in the struggle against false information.

Keywords: Natural Language Processing (NLP), Fake News Detection, Text Classification, Deep Learning, Graph Neural Network, Machine Learning, Text Complexity Monitoring

1. Introduction

Fake news on social media and various other media is widespread and is a matter of serious concern due to its ability to cause a lot of social destructive impacts [1, 2]. There has been a rapid increase in the spread of fake news in the last decade [3]. Such a spread of sharing articles online that do not comply with facts has led to many problems covering various domains like politics, sports, health, and science [4, 5]. One such area affected by fake news is the financial markets [6], where a rumor can have disastrous consequences and may bring the market to a standstill. Previously, in the field of fake news detection, there were multiple methods to detect fake news. NLP is one of the methods that were previously used for fake news detection [7]. The NLP rating of an algorithmic system enables the combination of speech understanding and speech generation. Additionally, naive bayes uses probabilistic reasoning to determine whether news stories are likely to be authentic or fraudulent based on feature independence assumptions [8]. By combining the predictions of several decision trees, Random Forest, an ensemble learning technique, improves accuracy and robustness and successfully separates false information from true information [9]. Support vector machines (SVMs) are useful for classifying news articles by extracting pertinent features, since they are good at generating appropriate decisions [10]. By combining the interpretability of logistic regression (LR) with the feature learning power of neural networks (NN), LR paired with NN offers a sophisticated method of detecting fake news[11]. Lastly, because Recurrent Neural Networks (RNNs) are adept at processing sequential data, they can better detect minor patterns that point to fake news by capturing the temporal dynamics of news stories and user interactions [12]. Conventional approaches to identifying fake news have predominantly relied on NLP and machine learning methods, with a particular focus on analyzing text and assessing the credibility of sources. Although these methods have laid a solid foundation for initial endeavors to eliminating false information, they typically fail to effectively tackle the complex and continually evolving nature of news dissemination on social media. The constraints of current methods emphasize the necessity for inventive solutions that can unwrap the complex network of connections within data, which is crucial for accurately differentiating between authentic and deceptive information [13].

This study introduces an innovative methodology for identifying fake news through the utilization of GNNs, a state-of-the-art deep learning technique known for its proficiency in analyzing data that possesses inherent graph structures. Our contribution encompasses two main aspects. Firstly, we present the utilization of GNNs in the domain of fake news detection, which represents a notable departure from conventional text-based analysis techniques. Additionally, we utilize the relational information pertaining to news items, their paths of dissemination, and the interactions among users in order to improve the process of detection. This methodology not only facilitates a more intricate comprehension of the dissemination of false information but also enhances the precision of identification by capturing patterns that may not be readily apparent exclusively through textual analysis. Our approach overcomes these challenges by employing an advanced feature extraction procedure that encompasses both the substance and the circumstances of news articles. The utilization of GNNs when combined with deep learning principles, allows for the exploitation of graph-based representations to effectively analyze the interconnected inherent in social media data. This novel methodology not only differentiates our research from current approaches but also establishes a fresh standard for the utilization of GNNs in the domains of information verification and fake news detection. The key technical challenge faced in our proposed methodology pertains to the efficient depiction and utilization of graph-based data for the purpose of detecting fake news. Conventional deep learning models lack inherent architectural capabilities to effectively process graph structures, thereby posing a significant obstacle in capturing the intricate interconnections and dynamics inherent in social media networks. In response to this issue, we have created an enhanced Graph Neural Network (GNN) structure that can effectively handle graph-structured data. This allows the model to acquire knowledge and recognize the distinct patterns associated with the spread of fake news.

The uniqueness of our study lies in the utilization and modification of GNNs to address the particular issue of identifying fake news. The area has traditionally been dominated by text-based analysis and traditional machine learning models. We expand the limits of what can be accomplished in identifying misinformation by providing a strong framework that can adjust to the changing nature of news distribution in digital environments. The findings of our study illustrate the unparalleled effectiveness of our approach compared to conventional methods, presenting a hopeful alternative for future research and practical implementations in the ongoing struggle against false information. The main objectives of our work are highlighted below:

- We Performed a comparison of GNN and traditional classifiers such as Decision Trees, Naive Bayes, Random Forest, SVM, LR, NN, and RNN.
- We investigated the ability of GNNs to capture intricate connections among news articles, which are depicted as nodes in a graph.
- We highlighted the potential of GNN to comprehend complex connections within news data, resulting in enhanced precision and forecasting.

2. Literature Review

Social media, as an autonomous platform, is the main source of fake news circulation. Users can spread false information by themselves or using bots [14, 15, 16]. Bots are algorithms that use matching input and associated response patterns to carry out particular tasks. Bots distribute bogus news in large quantities to make it seem credible because consumers typically believe everything they read online.5 Because those tales typically garner more attention than other stories, users are also more likely to share them. Additionally, those stories typically have more likes or comments. An additional factor in determining a user's liking to a topic is their emotions and feelings toward it[17]. According to MIT researchers, since people are just as interested in this activity as the bots, fake news spreads more quickly than genuine news [18, 19]. Within the research focusing on enhancing the identification of fake news using Graph Neural Networks (GNN) and deep learning, it is crucial to distinguish between 'fake news' and 'biases,' since they both have important functions within the disinformation ecosystem, although in distinct ways [20]. Fake news refers to deliberately invented material that is spread to deceive or mislead. It is characterized by a total absence of factual accuracy and is generally created to manipulate public opinion or generate financial gain by using sensationalism. Biases, in contrast to fake news, are predispositions or subjective leanings that may impact the perception and spread of news items. Biases, while not intrinsically involving the dissemination of erroneous information, may result in a distorted portrayal of facts, where certain elements are emphasized to support specific perspectives or agendas. The correlation between fake news and biases adds complexity to the misinformation ecosystem. Fake news erodes public discourse and trust by disseminating falsehoods, while biases can distort the public's comprehension of factual information, potentially bolstering the credibility of false narratives or exacerbating societal divisions. Recognizing the complex nature of the situation, the research highlights the need to identify false information by not just confirming its accuracy but also comprehending the biased ways it is presented and the structural patterns through which it spreads. This method emphasizes the complex task of differentiating truth from disinformation, taking into account both the immediate effects of fake news and the gradual effects of biases on public perception [21].

2.1. Combating Fake News

Misinformation is a long-existing problem, and its impact spans across technological and political spheres. It's crucial to address this issue because the increasing dependence on social networking sites for daily news is a continuous upward trend due to technological availability, and there's no indication of a decline soon. The tech giant Facebook has already taken steps to combat the circulation of misinformation on its website in some countries, partnering with third-party fact-checkers to review articles and posts and assess their accuracy. Identified fake news content is pushed down in the news feed, and action is taken against repeat offenders[22]. Previous approaches to halting the dissemination of false information have primarily concentrated their investigation on fake news articles spread by automated systems. Social media accounts that publish false material more frequently than real accounts are generally referred to as bots; they have a strong tendency to share unrelated content. Users who exhibit power over others—especially their social media followers—are their main targets. It was noted that the bots used to spread false information to the intended audience were usually active during the initial phases of the spread of fake news, drawing in like-minded individuals who subsequently shared the same content on social media[23]. Similarly,

research indicates that social bots often populate the social space to inflict harm and deceive social media users. They have also been employed to induce political disruptions, negatively impact the stock market, engage in personal information theft, and propagate misinformation.[24].
The main issue arises with the dynamic nature of the network [16]. Since the network deals with real-time data, it is necessary to control the diffusion of rumors early in the process[25]. One of the approaches to halting the dissemination of rumors was identifying their source[26]. The identification of a proper diffusion model led to an analysis of the pace at which fake news spreads. Then, based on the source, the anti-rumor–based approach (for a single source) or the approximation-based approach (for multiple sources) was employed to tackle the spread of rumors in the network.

2.2. Machine Learning Approachs

Several algorithms have been put to the test to see how well they identify false reports from unreliable sources. These algorithms, which have achieved higher accuracy, consist of the following:

2.2.1. Decision Tree A flexible supervised machine learning technique for both regression and classification applications is the decision tree [27]. Starting with the full dataset at the root node, it chooses the optimal feature for partitioning according to factors like Gini impurity or information gain. Recursive splitting continues in this manner until certain conditions are satisfied, like reaching a maximum depth or a minimum number of samples. Final predictions are stored in leaf nodes, and the resulting tree gives understandable if-else rules. Decision trees are useful in ensemble methods like random forests, where they improve robustness and performance even if they are prone to overfitting, which can be reduced via approaches like pruning. The decision tree can be represented as a set of rules

$$\text{Decision at node } n : \begin{cases} \text{If } X \leq T, & \text{then go to node } L, \\ \text{If } X > T, & \text{then go to node } R. \end{cases} \tag{1}$$

Xj represents the value of a specific feature in the input vector X, Tj is the threshold for the feature Xj at a particular node and The process continues recursively in the left or right sub-tree until a leaf node is reached. At a leaf node, the predicted output is denoted as y

2.2.2. Naive Bayes Naive Bayes is a probabilistic machine learning algorithm commonly used for classification tasks, especially in text-related applications like spam detection[28] or sentiment analysis[29]. Using prior knowledge about relevant conditions[30], the Bayes theorem is utilized to compute the probability of an event. To make the computation of probabilities easier, Naive Bayes posits conditional independence between features given the class label in the context of classification. For a given observation, the algorithm calculates the probability of each class and assigns the class with the highest probability to the observation. Despite its simplicity and the "naive" assumption, Naive Bayes often performs well, particularly in text classification tasks, making it a popular choice for various applications. The Naive Bayes can be represented as

$$y = argmax(y)P(Y = y|X1 = x1, X2 = x2, ..., Xn = xn) \quad (2)$$

where *P(Y=y)* is the prior probability of each class, *P(X1=x1— Y=y)* is the Class-conditional probabilities of each feature given the class and *argmax(y)* is the function used for selecting the class with the highest estimated probability among the possible class labels.

2.2.3. Random Forest Random Forest, an ensemble learning technique for classification and regression, constructs numerous decision trees during training, each based on a bootstrap sample of the original dataset[31]. Utilizing random feature selection at each node introduces diversity. These trees, grown to a specified depth or until a stopping criterion is met, collectively form the "forest." In prediction, for classification tasks, individual tree predictions are combined through majority voting, while for regression tasks, predictions are averaged. Random Forest is well known for its capacity to manage outliers, guarantee high accuracy, and lessen overfitting while providing insights into the significance of individual features. It's become a widely used tool in the field of machine learning, with many different applications, including image classification[32], medical diagnosis[33], and financial prediction[34]. The Random Forest can be represented as

$$Yensamble(X) = \frac{1}{T}\left(\sum t = 1^T Y_t(X)\right) \quad (3)$$

where Yt(X) is the prediction of Dt and T is the total number of trees.

2.2.4. Support Vector Machine (SVM) The supervised machine learning technique known as Support Vector Machine was developed to handle tasks related to regression and classification[35]. The principle of operation is determining the best hyperplane and optimizing the margin between classes to create a clear division. Support vectors are crucial to this procedure because they define this ideal boundary by showing which data points are closest to the hyperplane. The kernel trick demonstrates SVM's ability to handle non-linear decision boundaries, allowing implicit operations in higher-dimensional domains. The algorithm's performance is good even in high-dimensional areas and it is resistant to overfitting, which makes it useful in a variety of contexts. On the other hand, it can be computationally demanding for big datasets and susceptible to noise in the data. SVM is used in many different fields, including text classification[36], image recognition[37], and bioinformatics[38], because of its capacity to manage challenging classification issues. SVM can be represented as:

$$y_i(w * X_i + b) \geq 1 \tag{4}$$

where yi is the class label of the *i-th* data point, Xi is the feature vector of the *i-th* data point.

2.2.5. Logistic Regression with Neural Network Using LR as the output layer in a neural network intended for binary classification tasks is known as LR with NN.[39]. An input layer, hidden layers with activation functions, and an output layer with a single LR unit employing a sigmoid activation function are the three layers of a neural network. The output is compressed by the sigmoid function and limited to a range of 0 to 1, which represents the probability of falling into the positive class. Using binary cross-entropy loss and optimization techniques, the network modifies its weights and biases throughout training to reduce the difference between expected probability and actual class labels. For situations involving binary classification, integrating LR—typically a stand-alone algorithm—into a neural network works well. It is possible to identify complex patterns in the data thanks to this integration. It can be represented as

$$z = w * X + b \tag{5}$$

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{6}$$

where X represents the input features, w is the weight vector, b is the bias vector, z is the weighted sum of inputs, and $\sigma(z)$ is the sigmoid activation function

2.2.6. Graph Neural Network (GNN) An instance of a unique neural network designed specifically for handling graph-structured data is a Graph Neural Network (GNN). In graphs, entities are represented as nodes and their relationships as edges. GNNs aim to obtain embeddings for each node, which capture inherent characteristics as well as relationships with neighboring nodes.[40]. Through message-passing protocol, nodes enable the sharing of data with their neighbors and subsequently aggregate those neighbors' representations. Graph convolutional layers are used for these functions in notable architectures such as Graph Convolutional Networks (GCNs). Applications for GNNs can be found in many different fields, including social network analysis[41], recommendation systems[42], and molecular chemistry[43]. Tasks including node classification, graph classification, link prediction, and recommendation demonstrate their efficacy and emphasize how well they can capture complex dependencies in graph-structured data. GNN can be represented as

$$m_i^{(l+1)} = \sum_{j \in \mathcal{N}(i)} h_j^{(l)} \cdot W^{(l)} \tag{7}$$

$$h_i^{(l+1)} = \sigma\left(m_i^{(l+1)} + h_i^{(l)} \cdot W_{\text{self}}^{(l)}\right) \tag{8}$$

where $mi^{(l+1)}$ represents m_i in layer *l+1*, $hi^{(l+1)}$ represents hi in layer *l+1*, $W^{(l)}$ represents W for layer l, $W^{(l)}_{\text{self}}$ represents W_{self} for layer l, $\mathcal{N}(i)$ denotes the set of neighboring nodes of v_i and $\sigma(\cdot)$ is the placeholder for choose activation function
The study [44], centers on the integration of linguistic patterns with GNNs to improve the explainability of fake news detection. This approach presents a distinctive perspective that could potentially enhance the analytical capabilities of your method. However, it encounters certain limitations pertaining to the intricacy of its methodology, reliance on the quality of linguistic data, difficulties in generalizing findings, potential trade-offs between explainability and accuracy, the necessity for adaptability to evolving online behaviors, and unresolved ethical and privacy considerations. The model's practicality and impact could be significantly enhanced by addressing these limitations. The manipulation of news in order to attack fake news detectors As a means of evaluating the robustness of GNN-based detectors, Wang et al.'s Social Engagement presents an adversarial attack framework [45]. This framework brings to light the necessity of developing methods that are able to withstand adversarial behaviors.The paper may not fully address the real-world feasibility and cost of executing the proposed manipulation strategies, as large-scale attacks may require significant resources and coordination. Ethical and legal implications arise from manipulating social engagement metrics to test the vulnerability of fake news detectors. The paper's findings may become outdated if it does not consider the adaptability of social media platforms and detection systems to new types of attacks. The research could inadvertently affect public trust in these systems, so it is crucial to balance disclosure of vulnerabilities with recommendations for strengthening detection systems. Mitigation strategies should be developed to protect against manipulation.
The study conducted by Hiremath et al. [46], investigates the application of graph embeddings and GNNs in the detection of fake news. They achieved a classification accuracy of 94% in their analysis. Your approach can be likened to your method, which has the potential to incorporate supplementary features or algorithms beyond graph analysis to improve the accuracy of identifying fake news. . However, this research encounters certain constraints, such as the possibility of excessive dependence on complex data-intensive models that may not effectively apply to a wide range of datasets or real-life situations due to the ever-changing nature of news distribution and consumption. The effectiveness of the models is contingent upon the abundance and caliber of the training data and may be compromised in settings characterized by limited data availability, obsolescence, or inadequate representation of diverse news categories. Moreover, the paper may not comprehensively tackle the computational expenses linked to these sophisticated models, potentially restricting their suitability in settings with limited resources. In addition, the study may fail to consider the changing strategies employed by misinformation disseminators, which raises concerns about the adaptability of the models and the need for regular updates to ensure their continued effectiveness. Furthermore, the ethical implications concerning bias, privacy, and the possibility of these technologies being misused to suppress valid information

or viewpoints may not be adequately scrutinized. This underscores the necessity for comprehensive ethical principles in the creation and implementation of AI-powered fake news detection systems.

In the article [47], a novel framework named SAFER is introduced. This framework aims to identify instances of fake news on various social media platforms. Although the methodology combines data from various sources such as content, content-sharing behavior, and social networks, it does possess certain constraints. Significantly, the mentioned approach fails to acknowledge the insufficiency of appropriate extensive resources for pre-training NN, and the utilization of transfer learning does not yield substantial enhancements in performance within this particular field. Furthermore, the paper fails to investigate the potential influence of diverse social contexts on the detection of fake news. Notwithstanding these constraints, SAFER attains cutting-edge outcomes on counterfeit news datasets across various domains.

2.2.7. Recurrent Neural Network (RNN) A neural network that is specifically made for processing data sequentially is called a Recurrent Neural Network (RNN).[48]. Diverging from other NN, RNNs possess the capability to retain a hidden state, which allows them to record data from previous time steps and operate well on problems that involve temporal dependencies. This hidden state acts as a memory and is updated at every step based on the input that is received at that moment and the hidden state that came before it. One interesting feature of RNNs is that they share parameters between time steps, which makes information easier to generalize over the entire sequence. Nevertheless, traditional RNNs have problems, such as disappearing gradients. To overcome these issues, advanced variations such as Long Short-Term Memory (LSTM)[49] and Gated Recurrent Unit (GRU) networks have been introduced[50]. RNNs are widely applied in various domains, including time series analysis[51], NLP[52], and speech recognition[53], due to their capacity to handle sequential data effectively. RNN can be represented as

$$h_t = \tanh(W_{hx} \cdot x_t + W_{hh} \cdot h_{t-1} + b_h) \tag{9}$$

where h_t is the hidden state at time step t, x_t is the input at time step t, W_{hx} and W_{hh} are the weight matrices, b_h is the bias term and tanh is the activation function.

When contrasting conventional machine learning methods like Random Forests, Decision Trees, Naive Bayes, SVM, LR, and NN Maroco et al. (2011); Nhu et al. (2020), including RNNs, with GNNs, certain limitations become apparent. Traditional approaches encounter challenges in comprehending intricate dependencies and relationships within data structured as graphs, thereby constraining their effectiveness in tasks like node classification, graph classification, and link prediction. A noteworthy constraint is their innate struggle to grasp and utilize the inherent structure of graphs. GNNs, however, possess a tailored design for managing such graph-structured information and excel in capturing complex dependencies within networks. GNNs frequently surpass traditional techniques in scenarios where the data exhibits intrinsic interconnectedness. Their adeptness at leveraging the graph structure results in superior outcomes, rendering

GNNs particularly advantageous in domains where relationships among entities hold paramount importance. While traditional methods have their utility, GNNs truly stand out in tasks involving graph-structured data , offering a more potent and efficient solution for specific problems.

3. Material and Method

This method uses mathematical formulas to represent the steps that are carried out in the GNNs architecture to systematically detect fake news. The process uses layers of NN and graph-based representation learning to modify node properties such that they match the graph structure and categorize articles as bogus or real. An incremental process is involved in integrating GNN to identify bogus news. The methodology execution diagram is given in Figure 1.

Node Characteristic X_i is a node feature that represents the content of each article in

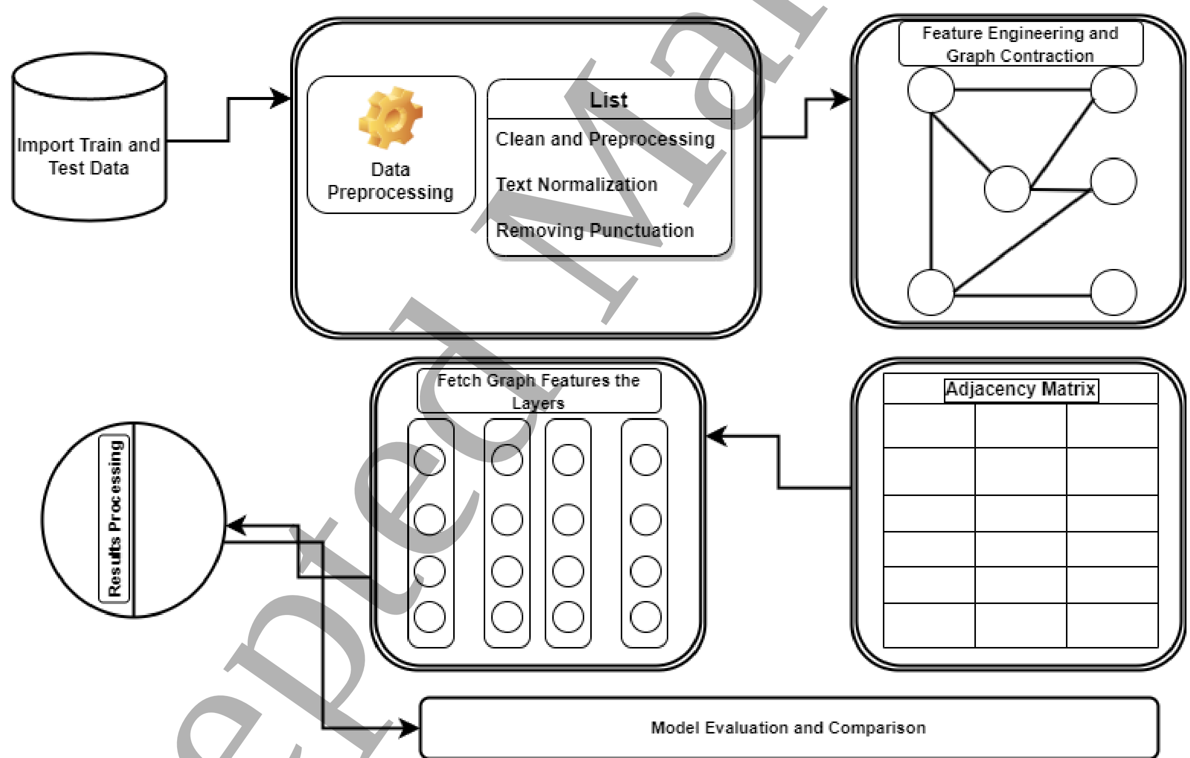

Figure 1. Methodology Execution Diagram

creation. This could involve using other numerical representations, such as embeddings. then make a graph. G can be described as $G = (V, E)$, where V represents the set of nodes (articles) and E represents the set of edges (relationships between articles). Take the following actions to construct the adjacency matrix: Let A be a matrix, and let A_{ij} represent the connection in the network between articles i and j. This could be ascertained by examining related factors, content similarities, and similar sources. The transformation rule that updates node representations in a graph convolutional network

(GCN) is represented by equation 1. Here, $H^{(l)}$ stands for the layer one node characteristics $l, W^{(l)}$ represents the weight matrix applied to these features, $\hat{A}$ refers to the normalized adjacency matrix of the graph, and $\hat{D}^{-\frac{1}{2}}$ denotes the symmetrically normalized degree matrix of $\hat{A}$. The procedure entails combining data from adjacent nodes, denoted as $\hat{A}$, and modifying these characteristics using the weights $W^{(l)}$. The node characteristics $H^{(l+1)}$ obtained from the previous layer $l+1$ are subsequently subjected to a non-linear activation function (such as ReLU or Sigmoid) on an element-wise basis. Updated representations are produced by this process and used in downstream activities or further levels within the GCN.

$$H^{(l+1)} = \sigma\left(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \tag{10}$$

The adjacency matrix A and the identity matrix I are added up in equation 2 to produce $\hat{A}$. Every node in the network has its self-loop connections preserved by the identity matrix I, ensuring that each node carries over its characteristics during message delivery. In addition, $\hat{D}$ represents the degree matrix of $\hat{A}$, containing information on each node's degree (count of connections) in the modified graph. LaTeX format is used to express the equation:

$$\hat{A} = A + I \tag{11}$$

The input layer of a graph neural network is in charge of initially designating node features. $H^{(0)} = X$, where X is the initial node feature matrix, represents the process. It depicts the process of setting the initial node representations in the network by allocating the feature matrix X to the first hidden layer $H^{(0)}$. The neural network's initial state is represented by this equation, where the input hidden layer receives the initial node features right away. A graph neural network's hidden layer is computed by applying the graph convolution process, which is typically represented as:

$$H^{(l+1)} = \sigma(D^{\sim -\frac{1}{2}}\tilde{A}^{\sim}D^{\sim -\frac{1}{2}}H^{(l)}W^{(l)}) \tag{12}$$

The updated node representations in layer $l+1$ are represented in this context by $H(l+1)$. These are obtained by integrating the adjacency matrix A, the node features $H^{(l)}$, and the weights $W^{(l)}$, followed by the application of a non-linear activation function . The output layer is denoted by, signifying the complete node representation.:

$$Z = H^{(L)}W^{(L)} \tag{13}$$

The equation defines the variables L and Z. L indicates the number of layers in the network, while Z represents the output node representations obtained by multiplying the final hidden layer $H^{(L)}$ with the weights $W^{(L)}$. During the training process of a graph neural network, the loss function, typically referred to as the cross-entropy loss L, quantifies the difference between the predicted labels Y^{pred} and the true labels Y^{true}. This function calculates the discrepancy between the predicted and actual labels by evaluating the negative logarithm of the likelihood of the predicted labels. The optimization method aims to minimize the loss function to update the parameters of the model. This update is accomplished by utilizing an optimizer, such as Adam, which modifies the weights $W^{(l)}$ in the network based on the computed gradients of the loss function concerning the weights. The weights are adjusted in the direction that minimizes the loss, as determined by the learning rate . The mathematical formulation is here:

$$L = -\frac{1}{N}\sum_{i=1}^{N} Y_{true}(i)\log(Y_{pred}(i)) \tag{14}$$

$$W_{new}^{(l)} = W_{old}^{(l)} - \alpha\frac{\partial W_{old}^{(l)}}{\partial L} \tag{15}$$

Finally, we utilized evaluation indicators to assess the effectiveness of each approach. At the end of the methodology, we also expressed the procedure of GNN in Algorithm 1.

Algorithm 1 Graph Neural Network for Fake News Detection

Require: Adjacency matrix $\mathbf{A}$, Feature matrix $\mathbf{X}$, Labels $\mathbf{Y}$

1: Initialize weights $\mathbf{W}$ and biases $\mathbf{b}$
2: Define GNN architecture:
3: Implement message passing function: $\mathbf{h}_i^{(l+1)} = \sigma\left(\sum_{j\in\mathcal{N}(i)} \mathbf{W}^{(l)}\cdot\mathbf{h}_j^{(l)}\right)$
4: Define graph convolutional layers: $\mathbf{H}^{(l+1)} = \sigma\left(\mathbf{A}\cdot\mathbf{H}^{(l)}\cdot\mathbf{W}^{(l)}\right)$
5: Split dataset into train, validation, and test sets
6: Initialize optimizer and loss function
7: **Train GNN model**:
8: **for** each epoch **do**
9: Forward pass:
10: Compute predictions: $\hat{\mathbf{Y}} = \text{GNN}(\mathbf{X},\mathbf{A};\mathbf{W},\mathbf{b})$
11: Calculate loss: $\mathcal{L}(\mathbf{Y},\hat{\mathbf{Y}})$
12: Backward pass:
13: Update weights using backpropagation: $\mathbf{W}\leftarrow\mathbf{W}-\alpha\cdot\nabla_{\mathbf{W}}\mathcal{L}(\mathbf{Y},\hat{\mathbf{Y}})$
14: **end for**

Our method based on GNN (Graph Neural Network), is specifically designed to identify fake news. It uses graph-based data structures to accurately represent the extensive network of links between news items, their propagation patterns, and user interactions. This method effectively captures the nuanced and complex relationships using node and edge representation, allowing our Graph Neural Network (GNN) model to detect patterns of misinformation propagation that are often disregarded by models that do not use relational data. In addition to content analysis, our approach incorporates sophisticated feature extraction techniques, such as node embeddings, to enhance the detection process by incorporating both content-based and structural insights. We improve this approach by using advanced preprocessing and feature engineering techniques to convert unprocessed news articles into organized graph data that accurately represents the complex relationships present in news stories. The GNN architecture we have developed is designed mainly to address the specific issues associated with detecting false news. It optimizes the layers for processing graph-structured input. In our work, we combine GNN with deep learning techniques to create a hybrid model. This model effectively identifies patterns in textual content and the spread of fake news.

Decision Trees and Naive Bayes, which rely on feature independence and statistical analysis, provide a simple approach but face difficulties when dealing with complex misinformation patterns. On the other hand, Random Forest and Support Vector Machine aim to enhance predictions by using ensemble learning and handling high-dimensional feature spaces. However, they may fail to consider the intricate relational dynamics that exist in news propagation networks. Although LR and NN are powerful in identifying patterns, they mostly concentrate on textual content, potentially disregarding important structural intricacies that are crucial for comprehending the spread of fake news. Recurrent Neural Networks (RNN) excel at handling sequential data and are useful for analyzing the chronological order of news stories. However, they may not fully consider the broader network of connections. On the other hand, the Graph Neural Network (GNN) utilizes the structural and relational data of news items and their distribution channels. This allows it to effectively detect complex patterns of false information by examining the relationships between articles, which conventional approaches might ignore. The computational complexity of GNNs is very efficient because they can process nodes (representing news items) and edges (showing connections) in parallel. This efficiency depends on the number of edges and the depth of the network design. The efficiency of the algorithm is denoted by $O(E+D)$, where E represents the number of edges and D represents the network depth. In order to handle the computing requirements of large datasets, graph sampling methods are used, which also enhance the efficiency of processing tasks. However, GNNs successfully handle the space complexity by using sparse matrix representations and message-passing methods to store graph structures such as adjacency matrices and node characteristics. This approach reduces redundant data storage. The space complexity, denoted in Big O notation, is given by the expression $O(V + E)$, where V is the number of vertices or nodes and E is the number of edges. This expression emphasizes the effective use of memory to store the graph data structure that is fundamental to GNNs.

3.1. Datasets

The detection of fake information is a significant and complex effort, especially in the modern era of social media, when users are able to share and flow material without verification occurring. It is possible for the transmission of false information to have substantial repercussions, such as the manipulation of public perceptions, the proliferation of wrong data, and an overall loss of faith in the profession of journalism by the general public. The identification of fake news may be accomplished by the employment of a variety of approaches, such as the application of NLP, machine learning, and deep learning systems. In general, these methodologies are dependent on a variety of factors, such as the substance of the news story, the news source, the social context, and the temporal dynamics [54] that are involved. The databases come from a wide variety of fields and sources, and they include both fake and genuine news stories.the dataset is collected from ‡ Listed below are the columns that they contain:

- Title: The Headline of the News Article
- Content: the main text of the news article
- Subject: The classification of the news article, such as politics, world news, etc.
- Date: The publication date of the news article.
- Target: The classification of the news article as either false or accurate.

In comparison to the real data, which has 21,417 rows, the fake dataset has 23,481 rows overall. The disparity is shown by the datasets, which show that there is a bigger quantity of fake news stories in comparison to real news items. In light of this, some machine learning algorithms would encounter a challenge, as they might acquire the ability to classify the vast majority of news stories as false. This would result in a high degree of accuracy, but a low level of completeness. In addition to including a wide variety of topics, writing styles, and news sources, the databases are heterogeneous. In the process of identifying the semantic and syntactic similarities and differences between fake news sources and real news articles, some NLP approaches, such as word embeddings, may encounter difficulties. Several years' worth of data are included in the dynamic datasets. Certain characteristics, such as the publishing date, the subject matter, and the origin of the news component, may be subject to change over time, which may have an effect on the relevance and accuracy of such characteristics. Please refer to Figure 2 for clarification.

3.1.1. Feature Engineering The procedure initially acquires two separate CSV files, $'fake'$ and $'true,'$ into distinct DataFrames named $'data_false'and'data_true.'$ Before merging into a unified DataFrame called $'data,'$ each DataFrame is modified by adding a column called $'target.'$ For the $'data_false'$ DataFrame, this column is labeled $'fake'$, and for the $'data\_true'$ DataFrame, $'true'$. Then, to guarantee unforeseen circumstances, the data entries are randomized. This leads to the removal of the $'date'and'title'$ columns because these characteristics might not significantly affect the classification process. Text preprocessing includes a variety of techniques, such as converting text to lowercase, eliminating punctuation, and eliminating commonly used stop words from the textual data. In the investigation stage, articles are categorized based on their "subject" to

‡ https://www.kaggle.com/code/maxcohen31/nlp-fake-news-detection-for-beginners

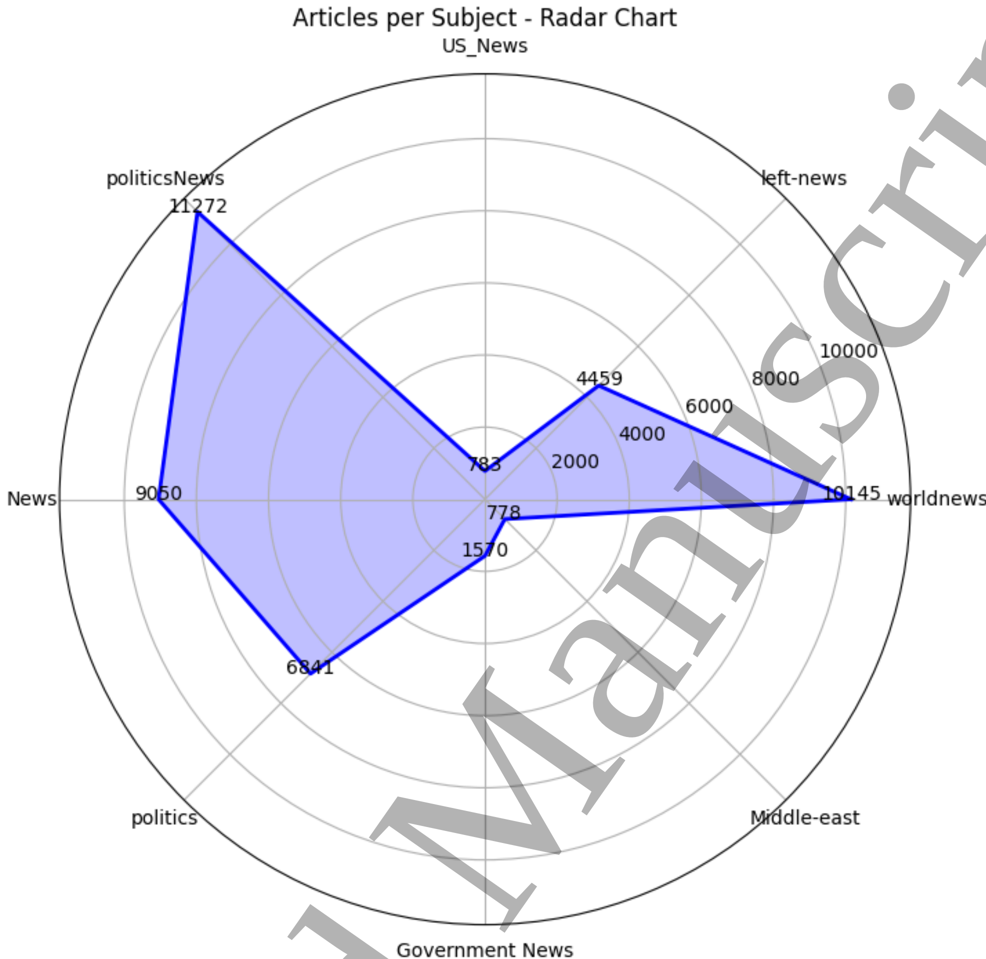

Figure 2. Dataset descriptions

produce a bar chart that efficiently displays the distribution of articles among various subjects. In addition, articles are grouped according to their 'goal' to give an illustration of how fake news is distributed with real ones. Different word clouds are also created for fictitious and real news stories, which help to visually represent the terms that are used the most in each category. A function is implemented to ascertain and graphically depict the terms that are most frequently used throughout the compilation of texts. After that, the dataset is split using the $'train\_test\_split'$ function from the $'sklearn'$ library into training and testing sets. To train and evaluate the future machine learning model, this is an essential stage. Ultimately, a function for creating and presenting a confusion matrix is developed [55], this aids in assessing how well the model can predict in comparison to the test set's actual labels. To put it simply, the algorithm manages a comprehensive set of steps that include loading data, preprocessing, exploring, and preparing the information needed to build a trustworthy fake news detection model.

Table 1. Fake News Classification Model Comparison

Methods	TP	TN	FP	FN	TTP	TFP	Accuracy
DTC	4090	4593	155	142	8683	297	0.966927
NBC	4116	4443	284	137	8559	421	0.953118
RFC	3900	4676	157	247	8576	404	0.955011
SVM	4238	4301	226	215	8539	441	0.950891
LRNNM	4227	4403	224	126	8630	350	0.961024
NN	4316	4219	331	114	8535	445	0.950445
RNN	3900	4719	151	220	8619	371	0.958732
GNN	**4316**	**4419**	231	**14**	**8735**	**245**	**0.9727**

3.2. Evaluation Criteria

Performance evaluation in classification models requires the use of assessment criteria. The percentage of accurately anticipated instances (true positives and true negatives) relative to all instances is known as accuracy. The method of computation requires figuring out the number of True Positives (TP), which represents the accurately predicted positive results. True Negatives (TN) indicate the accurate prediction of negative instances. False Positives (FP) refer to occurrences that are incorrectly forecasted as positive when they are negative, whereas False Negatives (FN) refer to instances that are incorrectly predicted as negative when they are truly positive. Mathematical equations:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{16}$$

It provides valuable information about a model's ability to forecast both positive and negative instances, helping to a thorough evaluation.

4. Results and Discussions

Table 1, presents the performance metrics of various classification models, such as Decision Tree, Naive Bayes, Random Forest, SVM, LR with NNM, Neural Network, and the Graph Neural Network (GNN) method we incorporated in this paper for fake news detection. These models were evaluated on a specific dataset. Generally, models demonstrate an accuracy range of about 95% to 97%, indicating their overall effectiveness in making predictions. The GNN, which has been utilized as an unusual approach in this study, shows impressive performance with numerous true positive and true negative results and significantly reduced false positive and false negative predictions. This illustrates GNN's ability to classify data accurately while reducing inaccurate predictions. Given that GNN's performance is on par with or higher than that of traditional classifiers, it appears that GNN holds great promise as a cutting-edge tool for knowledge graph predictive modeling. The noteworthy findings indicate that

GNN is a promising method for enhancing prediction accuracy and applying knowledge graph analysis, and that it is worthy of more research and development. In Table 1 TTP: Total True Prediction, TFP: Total False Prediction, DTC: Decision Tree Classifier, NBC: Naive Bayes Classifier, RFC: Random Forest classifier, SVM: SVM, LRNNM: LR with a Neural Network Mindset, NN: Neural Network, RNN: Recurrent Neural NEtwork, and GNN expressed Graph Neural Network.

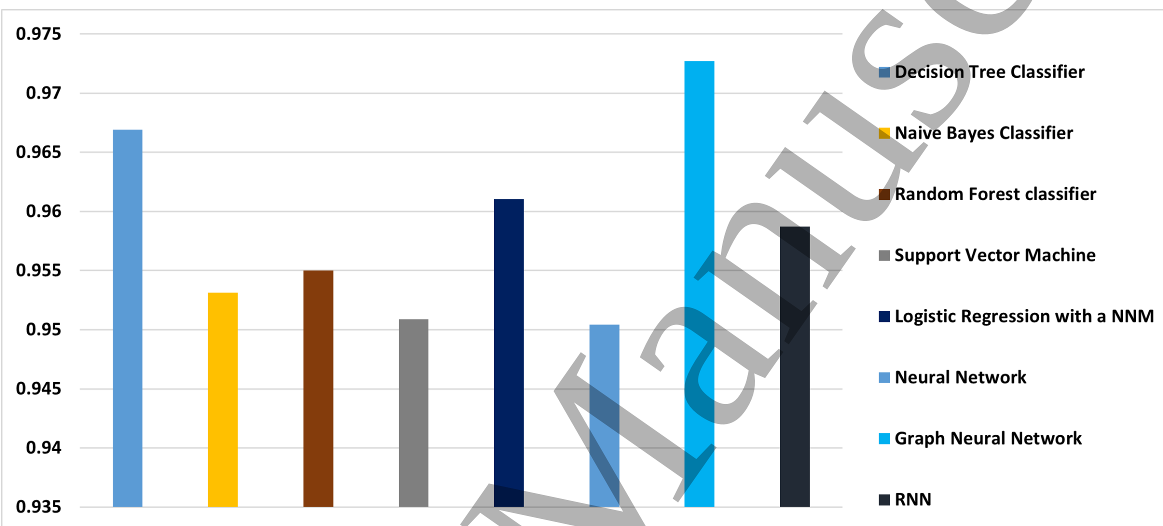

Figure 3. Accuracies of Different Methods

The variation in precision among various methods for identifying false information originates from their basic ability to handle complex relationships in the data. When news stories are represented as nodes in a graph structure, GNN is quite good at understanding the relationships between them, which allows for broad pattern detection. On the other hand, less complex models such as decision trees or naive Bayes might have trouble understanding these complex relationships, which could affect their accuracy. SVMs and LR models perform well in situations with well-defined class borders, but they may struggle with data that is very overlapping or complex, which could have a conflicting impact on their accuracy. Inadequate training or insufficient architectural depth can lead to inferior performance in NN and RNNs. More generally, GNN does exceptionally well in understanding complex graph topologies and intricate interactions, leading to increased accuracy. On the other hand, the effectiveness of other approaches depends on how well they can handle the complexities of the data and the unique features of the dataset. These factors include model complexity, data quality, and hyperparameter tuning accuracy. These factors deserve a careful investigation for more in-depth understanding, see Figure 3.

The model's ability to correctly categorize situations is measured by the metrics TP (true positives), TN (true negatives), FP (false positives), and FN (false negatives). true Negatives (TN) are instances that are accurately labelled as negative (e.g., correctly

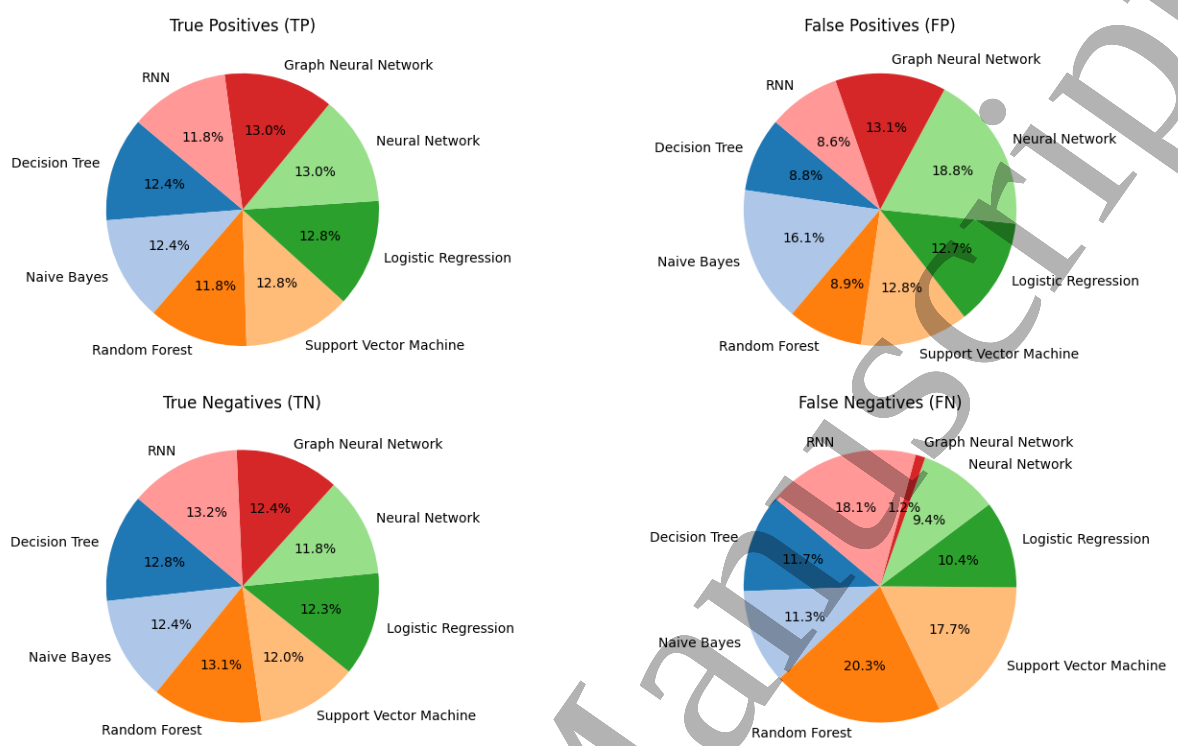

Figure 4. Detail assessment of different method

labelling truthful news as true), whereas True Positives (TP) indicate examples that are accurately identified as positive (e.g., correctly detecting fake news as fake). When the model incorrectly classifies a sample as positive—for example, by mislabeling real news as fake—this is known as a False Positive (FP).On the other hand, False Negatives (FN) refer to situations that are mistakenly labeled as bad, such as mistakenly accepting bogus news as true. The ”Total True Predicted” indicator effectively captures both positive and negative occurrences that the system accurately identified, demonstrating the overall prediction accuracy. On the other hand, the Total False Predicted metric combines cases that were wrongly classified and exposes the overall prediction errors of the model. Together, these measures provide a comprehensive insight of the model's performance in differentiating between classes as well as its limitations when it comes to making precise predictions. For more detail, see Figures 4 and 5.

4.1. Ablation Study

The integration of an ablation study into our research clarifies the importance of certain elements inside our Graph Neural Network (GNN) model, therefore enhancing our comprehension of its effectiveness in identifying fake news. This research systematically excludes multiple elements of the model, including graph features, GNN layers, preprocessing approaches, and the classifier head, in order to evaluate how they affect performance measures such as accuracy, precision, recall, and F1 score. Our objective

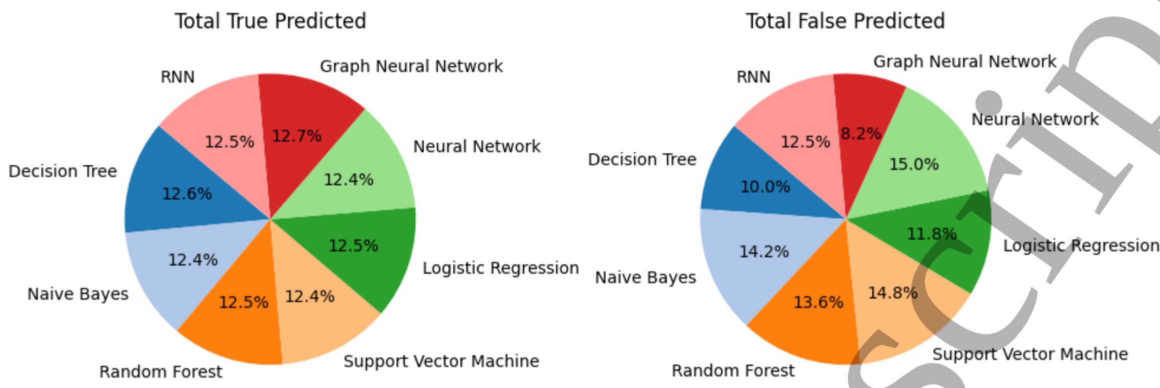

Figure 5. Total positive and negative prediction

is to identify the fundamental elements for successfully categorizing news items by retraining the model for each modification and comparing the outcomes with the initial configuration. Early projections indicate that this assessment will uncover vital observations, including the essentiality of particular node and edge characteristics in capturing the variations of false information, the ideal depth of GNN layers for efficiently representing the data, and the substantial impact of preprocessing on improving the model's ability to generalize. Furthermore, the research aims to identify the optimal classifier head for the given problem. The results of this ablation study not only confirm the selected components of the original model, but also provide guidance for future research by identifying areas that may be optimized and improved. In the end, comprehending the individual contribution of each component will allow the creation of more efficient and powerful models for preventing the dissemination of false information.

5. Conclusion

Nowadays, with social media making it simple to spread unconfirmed information, spotting fake news gets harder. False information has the power to sway public opinion, spread erroneous information, and damage journalism's reputation. Technologies like deep learning, machine learning, and NLP make it easier to identify fake news. These methods usually depend on elements such as news article content, sources, social context, and temporal dynamics. By analyzing different classification models according to measures such as accuracy, false positives (FP), false negatives (FN), true positives (TP), and false negatives (TN), the Graph Neural Network (GNN) has proven to be quite effective in identifying fake news. When compared to previous approaches, the GNN model shows impressive accuracy, demonstrating how well it can differentiate between real and fake news pieces. The precision of the system is derived from its capacity to deftly identify complex connections and trends among data arranged in a graph style. Better true positive (TP) and true negative (TN) values are also shown by GNN, demonstrating its dependability in predicting both positive and negative cases.On the other hand, while partially effective, alternative methods show limits in encapsulating the complex relationships present in news data. In summary, GNN proves to be a reliable technique that uses graph-based architectures to distinguish real news stories from fakes. Despite these promising results, our study has limitations. For model training, labelled data availability and quality are constraints. GNNs, like all machine learning methods, depend on the diversity and representativeness of the training dataset. The model's performance may be affected by datasets that don't capture all misinformation tactics, including fake news's latest strategies. GNNs' computational complexity, especially with large datasets, hinders scalability and real-time processing. Addressing these limitations in future there are many research avenues. Developing more advanced data augmentation techniques for fake news could improve model robustness against emerging misinformation strategies. Second, investigating more efficient GNN architectures or hybrid models that balance accuracy and computational efficiency could enable real-time fake news detection. Finally, integrating user behavioral data and network dynamics into the detection process may reveal fake news propagation patterns, improving countermeasures.

References

[1] Z. Khanam, B. Alwasel, H. Sirafi, and M. Rashid, "Fake news detection using machine learning approaches," in *IOP conference series: materials science and engineering*, vol. 1099, p. 012040, IOP Publishing, 2021.

[2] M. Wasim, F. Al-Obeidat, F. Moreira, H. Gul, and A. Amin, "Forecasting networks links with laplace characteristic and geographical information in complex networks," *Procedia Computer Science*, vol. 224, pp. 357–364, 2023.

[3] A. Holan, "The media's definition of fake news vs. donald trump's," *First Amend. L. Rev.*, vol. 16, p. 121, 2017.

[4] D. M. Lazer, M. A. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, M. J. Metzger, B. Nyhan, G. Pennycook, D. Rothschild, *et al.*, "The science of fake news," *Science*, vol. 359, no. 6380, pp. 1094–1096, 2018.

[5] M. Wasim, F. Al-Obeidat, A. Amin, H. Gul, and F. Moreira, "Enhancing link prediction efficiency with shortest path and structural attributes," *Intelligent Data Analysis*, no. Preprint, pp. 1–17, 2023.

[6] S. Kogan, T. J. Moskowitz, and M. Niessner, "Fake news: Evidence from financial markets," *Available at SSRN*, vol. 3237763, 2019.

[7] S. A. Alkhodair, S. H. Ding, B. C. Fung, and J. Liu, "Detecting breaking news rumors of emerging topics in social media," *Information Processing & Management*, vol. 57, no. 2, p. 102018, 2020.

[8] M. Granik and V. Mesyura, "Fake news detection using naive bayes classifier," in *2017 IEEE first Ukraine conference on electrical and computer engineering (UKRCON)*, pp. 900–903, IEEE, 2017.

[9] J. Antony Vijay, H. Anwar Basha, and J. Arun Nehru, "A dynamic approach for detecting the fake news using random forest classifier and nlp," in *Computational Methods and Data Engineering: Proceedings of ICMDE 2020, Volume 2*, pp. 331–341, Springer, 2020.

[10] N. F. Baarir and A. Djeffal, "Fake news detection using machine learning," in *2020 2nd International workshop on human-centric smart environments for health and well-being (IHSH)*, pp. 125–130, IEEE, 2021.

[11] K. Poddar, K. Umadevi, *et al.*, "Comparison of various machine learning models for accurate

detection of fake news," in *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*, vol. 1, pp. 1–5, IEEE, 2019.

[12] J. A. Nasir, O. S. Khan, and I. Varlamis, "Fake news detection: A hybrid cnn-rnn based deep learning approach," *International Journal of Information Management Data Insights*, vol. 1, no. 1, p. 100007, 2021.

[13] J. Yi, T. Nasukawa, R. Bunescu, and W. Niblack, "Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques," in *Third IEEE international conference on data mining*, pp. 427–434, IEEE, 2003.

[14] S. Kumar, R. Asthana, S. Upadhyay, N. Upreti, and M. Akbar, "Fake news detection using deep learning models: A novel approach," *Transactions on Emerging Telecommunications Technologies*, vol. 31, no. 2, p. e3767, 2020.

[15] H. Gul, F. Al-Obeidat, A. Amin, M. Tahir, and K. Huang, "Efficient link prediction model for real-world complex networks using matrix-forest metric with local similarity features," *Journal of Complex Networks*, vol. 10, no. 5, p. cnac039, 2022.

[16] H. Gul, F. Al-Obeidat, A. Amin, M. Tahir, and F. Moreira, "A systematic analysis of community detection in complex networks," *Procedia Computer Science*, vol. 201, pp. 343–350, 2022.

[17] E. F. Dos Santos, D. Silva de Carvalho, and J. Oliveira, "Pattern identification of bot messages for media literacy," in *Proceedings of the Brazilian Symposium on Multimedia and the Web*, pp. 121–128, 2021.

[18] S. Lohr, "It's true: False news spreads faster and wider. and humans are to blame," *The New York Times*, vol. 8, 2018.

[19] Z. Zainab, F. Al-Obeidat, F. Moreira, H. Gul, and A. Amin, "Comparative analysis of machine learning algorithms for author age and gender identification," in *Proceedings of International Conference on Information Technology and Applications: ICITA 2022*, pp. 123–138, Springer, 2023.

[20] B. Gawronski, "Partisan bias in the identification of fake news," *Trends in Cognitive Sciences*, vol. 25, no. 9, pp. 723–724, 2021.

[21] S. Van der Linden, C. Panagopoulos, and J. Roozenbeek, "You are fake news: political bias in perceptions of fake news," *Media, culture & society*, vol. 42, no. 3, pp. 460–470, 2020.

[22] T. Lyons, "Hard questions: What's facebook's strategy for stopping false news," *Facebook newsroom*, vol. 23, p. 2018, 2018.

[23] C. Shao, G. L. Ciampaglia, O. Varol, K.-C. Yang, A. Flammini, and F. Menczer, "The spread of low-credibility content by social bots," *Nature communications*, vol. 9, no. 1, pp. 1–9, 2018.

[24] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, "The rise of social bots," *Communications of the ACM*, vol. 59, no. 7, pp. 96–104, 2016.

[25] Z. Zhao, P. Resnick, and Q. Mei, "Enquiring minds: Early detection of rumors in social media from enquiry posts," in *Proceedings of the 24th international conference on world wide web*, pp. 1395–1405, 2015.

[26] S. Shelke and V. Attar, "Source detection of rumor in social network–a review," *Online Social Networks and Media*, vol. 9, pp. 30–42, 2019.

[27] Y.-Y. Song and L. Ying, "Decision tree methods: applications for classification and prediction," *Shanghai archives of psychiatry*, vol. 27, no. 2, p. 130, 2015.

[28] K. V. Kumar and M. Ramamoorthy, "Naive bayes classifier algorithm for spam detection of email to improve accuracy and in comparison with decision tree algorithm," *Journal of Pharmaceutical Negative Results*, pp. 49–55, 2022.

[29] M. Abbas, K. A. Memon, A. A. Jamali, S. Memon, and A. Ahmed, "Multinomial naive bayes classification model for sentiment analysis," *IJCSNS Int. J. Comput. Sci. Netw. Secur*, vol. 19, no. 3, p. 62, 2019.

[30] M. M. Saritas and A. Yasar, "Performance analysis of ann and naive bayes classification algorithm for data classification," *International journal of intelligent systems and applications in engineering*, vol. 7, no. 2, pp. 88–91, 2019.

[31] H. Pang, A. Lin, M. Holford, B. E. Enerson, B. Lu, M. P. Lawton, E. Floyd, and H. Zhao, "Pathway analysis using random forests classification and regression," *Bioinformatics*, vol. 22, no. 16, pp. 2028–2036, 2006.

[32] A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forests and ferns," in *2007 IEEE 11th international conference on computer vision*, pp. 1–8, Ieee, 2007.

[33] E. Elyan and M. M. Gaber, "A fine-grained random forests using class decomposition: an application to medical diagnosis," *Neural computing and applications*, vol. 27, pp. 2279–2288, 2016.

[34] Z. Rustam and G. S. Saragih, "Predicting bank financial failures using random forest," in *2018 International Workshop on Big Data and Information Security (IWBIS)*, pp. 81–86, IEEE, 2018.

[35] W. S. Noble, "What is a support vector machine?," *Nature biotechnology*, vol. 24, no. 12, pp. 1565–1567, 2006.

[36] A. Basu, C. Walters, and M. Shepherd, "Support vector machines for text categorization," in *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*, pp. 7–pp, IEEE, 2003.

[37] O. Okwuashi and C. E. Ndehedehe, "Deep support vector machine for hyperspectral image classification," *Pattern Recognition*, vol. 103, p. 107298, 2020.

[38] D. Chicco, "Support vector machines in bioinformatics: a survey," *Politecnico di Milano, Dipartimento di Elettronica e Informazione*, 2012.

[39] M. Zekić-Sušac, N. Šarlija, A. Has, and A. Bilandžić, "Predicting company growth using logistic regression and neural networks," *Croatian operational research review*, vol. 7, no. 2, pp. 229–248, 2016.

[40] Z. Hu, Y. Dong, K. Wang, K.-W. Chang, and Y. Sun, "Gpt-gnn: Generative pre-training of graph neural networks," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1857–1867, 2020.

[41] X. T. Dinh and H. Van Pham, "Social network analysis based on combining probabilistic models with graph deep learning," in *Communication and Intelligent Systems: Proceedings of ICCIS 2020*, pp. 975–986, Springer, 2021.

[42] C. Gao, X. Wang, X. He, and Y. Li, "Graph neural networks for recommender system," in *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pp. 1623–1625, 2022.

[43] H. Cai, H. Zhang, D. Zhao, J. Wu, and L. Wang, "Fp-gnn: a versatile deep learning architecture for enhanced molecular property prediction," *Briefings in bioinformatics*, vol. 23, no. 6, p. bbac408, 2022.

[44] C.-T. Li, "Explainable detection of fake news and cyberbullying on social media," in *Companion Proceedings of the Web Conference 2020*, pp. 398–398, 2020.

[45] H. Wang, Y. Dou, C. Chen, L. Sun, P. S. Yu, and K. Shu, "Attacking fake news detectors via manipulating news social engagement," in *Proceedings of the ACM Web Conference 2023*, pp. 3978–3986, 2023.

[46] P. Hiremath, S. S. Kalagi, and Mohana, "Analysis of fake news detection using graph neural network (gnn) and deep learning," in *2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS)*, pp. 1805–1811, 2023.

[47] S. Chandra, P. Mishra, H. Yannakoudakis, M. Nimishakavi, M. Saeidi, and E. Shutova, "Graph-based modeling of online communities for fake news detection," *arXiv preprint arXiv:2008.06274*, 2020.

[48] S. Li, W. Li, C. Cook, C. Zhu, and Y. Gao, "Independently recurrent neural network (indrnn): Building a longer and deeper rnn," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5457–5466, 2018.

[49] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, 2020.

[50] J. Chen, H. Jing, Y. Chang, and Q. Liu, "Gated recurrent unit based recurrent neural network

for remaining useful life prediction of nonlinear deterioration process," *Reliability Engineering & System Safety*, vol. 185, pp. 372–382, 2019.

[51] W. Yu, I. Y. Kim, and C. Mechefske, "Analysis of different rnn autoencoder variants for time series classification and machine prognostics," *Mechanical Systems and Signal Processing*, vol. 149, p. 107322, 2021.

[52] K. M. Tarwani and S. Edem, "Survey on recurrent neural network in natural language processing," *Int. J. Eng. Trends Technol*, vol. 48, no. 6, pp. 301–304, 2017.

[53] J. Li, R. Zhao, H. Hu, and Y. Gong, "Improving rnn transducer modeling for end-to-end speech recognition," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 114–121, IEEE, 2019.

[54] R. Misra, "News category dataset," *arXiv preprint arXiv:2209.11429*, 2022.

[55] C. Room, "Confusion matrix," *Mach. Learn*, vol. 6, p. 27, 2019.