A Weakly Supervised MIL Approach to Fake News Detection via Propagation Tree Analysis

Shariq Bashir
College of Computer and Information Sciences,
Imam Mohammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia

***Abstract*—This paper presents a weakly supervised Multiple Instance Learning (MIL) framework for fake news detection in social media, leveraging propagation tree analysis to model the spread of misinformation across online networks. Unlike traditional text-based or graph-based methods, our approach captures fine-grained post-level stances (support, denial, question, comment) and aggregates them to infer news veracity using a novel hierarchical attention mechanism. The framework incorporates social network dynamics of information diffusion, offering deeper insights into how user interactions amplify or suppress misinformation. We evaluate our model on benchmark datasets, including PolitiFact and GossipCop from FakeNewsNet, comprising over 23,000 news articles and hundreds of thousands of user engagements, as well as on the SemEval-8 dataset for binary classification of true vs. fake news. Our method achieves up to 94.3% accuracy and 91.7% F1-score, outperforming state-of-the-art machine learning and deep learning baselines. Ablation studies further validate the contribution of stance aggregation and attention-based propagation modeling. These results highlight the effectiveness of integrating stance detection, propagation structures, and weakly supervised learning for scalable and interpretable fake news verification in online environments.**

Keywords*—*Identifying fake news; social network analysis; post stance detection; deep learning; information retrieval; multiple instance learning

I. Introduction

Misinformation, commonly known as fake news, has been characterized as low-quality news in various studies [1], [2], [3], [4]. Fake news, by definition, refers to a news article that is verifiably and intentionally false, with the potential to mislead readers [2]. Fake news verification or detection, a critical endeavor in today's information age, is dedicated to assessing the accuracy of news articles on various subject matters [5]. Traditional fake news detection methods rely heavily on supervised learning using handcrafted features [6], which can be labor-intensive and lack adaptability. To address these limitations, researchers have adopted neural approaches such as CNNs and RNNs [7], [8], as well as transformer-based models [9].

In addition to neural methods, kernel learning algorithms have been introduced as a means of comparing propagation trees [10]. Propagation tree, which represents the dissemination of information across social networks, plays a crucial role in understanding the dynamics of fake news propagation. Kernel learning allows for the capture of complex propagation patterns, providing a more refined perspective on how misinformation spreads and infiltrates various online platforms. This approach takes into account not only the content of the news but also the intricate network structure of replying posts through which it circulates [9].

A. Main Contribution

Prior studies have shown that posts expressing skepticism, like doubts or questions, play a crucial role in assessing the credibility of information [11], [12], [13]. These critical user responses act as informal fact-checking and help slow the spread of misinformation by encouraging more thoughtful engagement. The proposed approach centers on the structure of news dissemination, known as the propagation structure [5]. This structure maps how users share and reply to posts, forming a tree-like network of interactions. As illustrated in Fig. 1 and 2, each post can be traced back to a source article, and replies create chains that reflect evolving opinions [13]. Importantly, users often reply to the most recent post rather than the original news, which shapes how discussions unfold. For instance, denial posts about fake news typically receive supportive replies that reinforce the denial, while denial of real news tends to spark counter-replies, creating a dynamic debate. These interaction patterns within the propagation trees offer valuable insights for distinguishing between true and false news. Fake news generally triggers more denying responses, resulting in denser and more complex reply chains. This feedback loop demonstrates how user reactions can amplify or suppress information spread, making it essential to understand these dynamics for effective misinformation detection.

Building on this framework, we propose a novel, weakly supervised method to detect fake news by leveraging Multiple Instance Learning (MIL) [14]. Unlike traditional MIL methods that require labels for individual sentences, our approach uses only document-level veracity annotations, which is more practical for large-scale data. A key challenge is linking these overall news labels to the finer details found in individual sentences and posts within the propagation tree. To address this, we uniquely utilize the propagation trees themselves as the foundational structures for learning stance and veracity patterns from the social media post interactions [15]. To manage the complexity of classifying multiple news and stance types, we decompose the problem into several binary classification tasks. We designate real news and denial stances as "positive" classes at their respective levels, with all other classes treated as "negative." This formulation allows us to train separate MIL models for each veracity-stance pair, effectively capturing the diverse nature of misinformation and user responses. Finally, we introduce a novel hierarchical attention mechanism to integrate the binary model outputs into a coherent multi-class prediction. This mechanism weighs and combines stance predictions across all MIL models to infer the overall truthfulness of news articles. Additionally, we incorporate a weighted strategy that prioritizes critical sentences during news-level classification, further enhancing prediction accuracy. Altogether, this is the first application of propagation trees within a MIL framework for fake news verification, offering a scalable and insightful approach to understanding and combating misinformation.

The remainder of this paper is organized as follows. Section II reviews related work on fake news detection, including machine learning, deep learning, and social network analysis approaches. Section III defines the problem statement and introduces the formal notations used in our study. Section IV presents the proposed weakly supervised Multiple Instance Learning framework with propagation tree modeling and stance aggregation. Section V describes the construction of the news-post propagation tree and its role in capturing information diffusion. Section VI reports the experimental setup, datasets, evaluation metrics, and results, followed by ablation and component studies. Section VII provides a detailed discussion and interpretation of the findings. Finally, Section VIII concludes the paper and outlines potential directions for future research.

II. Related Work

Fake news detection (FND) research spans three main domains: Machine Learning (ML), Deep Learning (DL), and Social Network Analysis (SNA). ML and DL methods have shown strong performance in identifying fake news through content and user behavior features, while SNA leverages network structures to analyze how misinformation spreads [16]. However, approaches explicitly incorporating social-context models remain limited, highlighting a gap our work addresses. ML techniques typically rely on supervised classifiers like Support Vector Machines (SVM) and Decision Trees (DT). SVMs, using syntactic and lexical features, have achieved F-measures up to 0.87 [17], [18], while DTs and Random Forests (RF) effectively combine content and contextual features, yielding accuracies around 0.86 [19], [20]. Semi-supervised graph-based ML approaches also show promise in detecting fake user accounts spreading misinformation on Twitter, achieving over 90% accuracy [21]. DL methods overcome the manual feature engineering bottleneck of ML by learning hierarchical representations directly from data. Models such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been successfully applied to FND tasks, with hybrid architectures reaching accuracies above 0.8 on benchmark datasets [22], [23], [24]. Geometric deep learning and propagation-based frameworks further capture the relational structure of information diffusion [25]. Comprehensive evaluations demonstrate that ML and DL methods both contribute effectively to fake news detection [26], [27]. SNA approaches analyze the diffusion and interaction patterns within social networks. Large-scale studies leveraging millions of tweets have employed algorithms like Collective Influence to identify super-spreaders of misinformation [28], [29]. Network-based pattern-driven methods [30] and Graph Neural Networks (GNNs) [31] utilize topological and behavioral features to enhance detection accuracy. Knowledge graph techniques enable fact-checking through entity relationships [32], while diffusion network analyses reveal that fake news spreads faster and deeper than truth, amplified by echo chambers and bots [33].

Our approach differs from these prior works by explicitly modeling the social-contextual dynamics of fake news propagation through a novel propagation tree framework that captures user reactions—such as support, doubt, and denial—and their influence on misinformation spread [15]. Unlike conventional ML/DL models focused primarily on content or user-level features, our method leverages weakly supervised Multiple Instance Learning on propagation trees, enabling efficient learning from article-level labels while incorporating the complex interaction patterns inherent in social media discussions. Moreover, we introduce a hierarchical attention mechanism to integrate multi-level stance predictions, improving interpretability and detection accuracy. This social-context-centric approach fills the identified gap in existing research, providing a more holistic understanding and detection of fake news in online networks.

III. Problem Statement

Our strategy for identifying fake news relies on a well-organized dataset denoted as G. Each instance within this dataset, represented by g_i, is a triplet comprising $g_i = \{d_i, C_{d_i}, L_{d_i}\}$ contains the following items:

- News Item (d_i): The central object of investigation, represented as a news article, post, or similar textual content suspected of being fake.
- Replying Posts Set on d_i (C_{d_i}): A chronologically ordered sequence of users' posts on d_i. This sequence captures the surrounding discussion and engagement, providing contextual information for analysis.
- Veracity Label (L_{d_i}): A binary label indicating the truthfulness of the news. $L_{d_i} = 0$ signifies verified truth, while $L_{d_i} = 1$ denotes confirmed falsity.

While C_{d_i} contains all the posts on d_i in temporal order, a crucial aspect lies in acknowledging the explicit connections, such as reply-to and share relationships, that exist between them [34]. We represent each news d_i through a news-post propagation tree (newsPostTree) [15]. The newsPostTree depicts the potential spread of the news from its origin to its wider audience. Edges in this tree follow the direction of potential dissemination. The proposed approach investigates the veracity of fake news on the basis of stances of replying posts on the news. Having stated this, the proposed approach identifies the veracity of fake news on the basis of the following two tasks.

1) Stance detection: Given a post c_i discussing the veracity of a news item d_i, predict the post's overall stance L_{c_i} on the news's accuracy. Possible stances include: *Support:* The post expresses agreement with the news's truthfulness. *Deny:* The post disagrees with the news's accuracy. *Question:* The post raises doubts or seeks clarification about the news's validity. *Comment:* The post focuses on aspects unrelated to the news's veracity.

2) Fake news verification: For each news article d_i, this task aims to determine its overall veracity as either real or fake. We hypothesize that the news article's veracity can be modeled as a weighted collective assumption based on the stances of its associated posts, similar to the Multiple-Instance Learning (MIL) framework [14]. In our approach, the news article's label is predicted based on the most likely stance category among the stances expressed in the associated posts, considering their distribution within a newsPostTree.

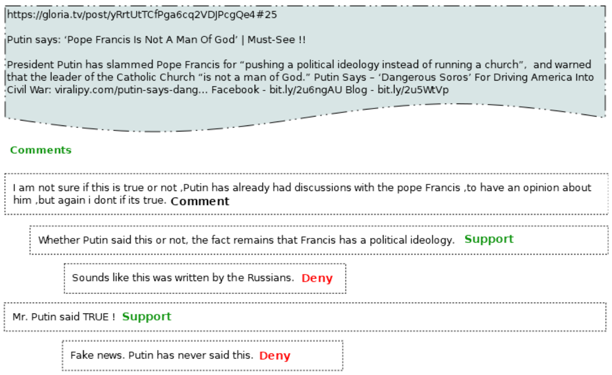


Fig. 1. An example fake news and replying posts on the news.

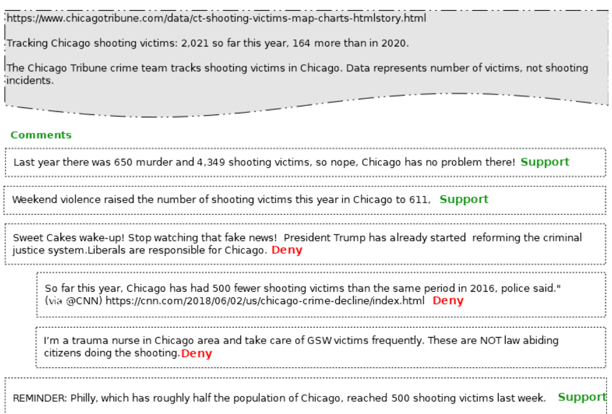


Fig. 2. An example real news and replying post on the news.

IV. Proposed Approach

Traditionally, fake news verification has often been tackled as a single, multi-class classification problem [6]. However, this approach overlooks the intricate interplay between the fine-grained nuances of post stances (e.g. support, deny, question, comment) and the multifaceted nature of news article veracity (e.g. real or fake). To unlock the full potential of this rich information, we propose a novel extension to the Multiple-Instance Learning (MIL) framework. We recognize that a single, multi-class approach might miss the subtle interactions between stance and veracity. Therefore, we decompose the problem into a multitude of binary classification tasks, each focusing on a specific veracity-stance pairing. Let S be the number of stance classes of posts, and let E be the number of veracity classes for news. Given E and S, there exist $H = E * S$ potential veracity-stance target class pairs, each requiring the training of an individual binary classifier for weak supervision in the task of detecting false information. This granularization allows each model to hone its expertise on a precise combination, capturing the subtle details that might be obscured in a global approach.

While individual binary models provide valuable insights, we need to unite their findings to paint a complete picture. This is where the strength of the MIL framework shines. We strategically aggregate the predictions of the individual models, leveraging their collective knowledge to arrive at a robust verdict on the overall news article's veracity.

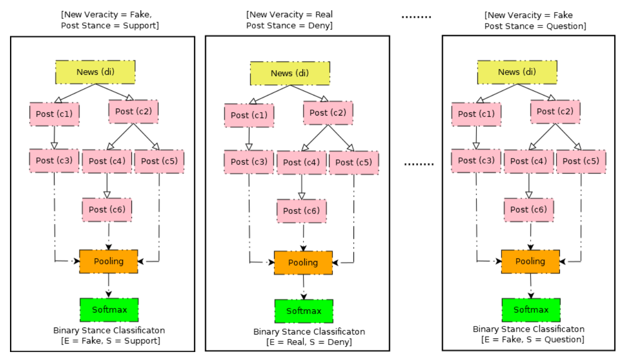


Fig. 3. An architecture of post encoding and binary stance classification of posts.

A. Encoding Replying Posts

To analyze the content of each post and its stance towards news veracity, we leverage powerful neural networks. Each post is first transformed into a sequence of terms $T_{c_i} = \{t_1, t_2, t_3, ..., t_{[c_i]}\}$, a linguistic fingerprint captured by pre-trained term embeddings. These embeddings translate each term t_i into an n-dimensional vector $t_i \in V^n$, encapsulating its meaning and relationships to other terms. Next, we utilize a specialized neural network called a Gated Recurrent Unit (GRU) [35] to understand the flow of information within the post. The GRU processes the term sequence one by one, effectively capturing the context and sentiment woven through the terms. This results in a fixed-size hidden vector that summarizes the essence of the entire post. Finally, to assess the overall stance towards the news article's veracity, we employ two separate GRU-based encoders [36]. These encoders act like dedicated detectives, each scrutinizing the post from a different angle. Their analyses are then condensed into two vectors, capturing the key aspects of the stance of post L_{c_i} and the associated news d_i. In simpler terms, given a number of terms ($T_{c_i} = \{t_1, t_2, t_3, ..., t_{[c_i]}\}$) in the post c_i, the approach breaks down the post into its building blocks (terms) and analyzes how they flow together in a standard GRU transition equation $GRU()$, extracting key aspects of its message (encoders) to ultimately understand its stance towards the news's trustworthiness.

$$V_{d_i} = V_{|d_i|} \tag{1}$$

$$V_{|d_i|} = GRU\left(t_{|d_i|}, V_{|d_i|-1}, \Delta_{d_i}\right) \tag{2}$$

$$V_{c_i} = V_{|c_i|} \tag{3}$$

$$V_{|c_i|} = GRU\left(t_{|c_i|}, V_{|c_i|-1}, \Delta_{c_i}\right) \tag{4}$$

$V_{|d_i|-1}$ and $V_{|c_i|-1}$ serve as hidden units capturing information about the preceding terms in news item d_i and associated post c_i. Δ_{d_i} and Δ_{c_i} encapsulate the entirety of the news and post encoder parameters.

V. News Post Propagation Tree (newsPostTree)

Given a news item d_i, to identify whether d_i is fake news, the proposed approach utilizes the hierarchical tree structure of the replying posts $C_{d_i} = \{c_1, c_2, c_3, ..., c_{|C_{d_i}|}\}$ on d_i, capturing intricate propagation patterns essential for classification (see Fig. 3). Imagine a complex web of online posts, where some spread truth and others spin yarns of misinformation. One such news-post pattern is the "*support-comment-support*" sequence. Imagine some replying posts on a news support the news. Others chime in with questions or comments, and if the evidence in a supporting post holds up, we might see further support emerge. This cyclical pattern tends to be more prevalent in true news, where credible replying posts attract validation and build momentum [15]. On the other hand, fake news often struggle to generate sustained support. They might trigger initial curiosity or even controversy, but without a solid foundation, they tend to fizzle out, lacking the magnetism to attract lasting support.

This model acts like a skilled investigator, navigating this web through the proposed *newsPostTree* structure. Unlike models who focus on individual posts, this model investigates the entire path information takes, starting at the source (root node) and moving down each branch. Each post (node of the tree) can be considered a piece of evidence. This model gathers clues (called "*features*") from each post, like suspicious language or biased references, and strengthens them based on the path they took. If a post agrees with its parent's denial of the original news, the evidence against both strengthens as the model travels down the branch. This "*evidence boost*" happens for each non-leaf node, building a stronger case against potential misinformation. Now, what makes this model special? Unlike models who start at the leaf nodes and piece things together (bottom-up), this model starts at the news (root node) and builds the information tree level by level. Each post's "*evidence file*" combines its own clues with those from its parent, creating a richer picture with each step down the tree. It's like a team of sub-models, each adding their findings to the case news, building a more comprehensive picture with every level [15].

Consider a scenario where the hidden information within a non-leaf node can be seamlessly transmitted to all its child nodes simultaneously, ensuring no loss occurs in the process. In such a context, the determination of the hidden state V_{c_i} for a given node c_i involves a computation that combines the existing hidden state $V_{\text{Parent}(c_i)}$ of its parent node Parent(c_i) with the input vector specific to the node itself. This integration process allows the formulation of transition equations for node c_i, presenting them in a manner consistent with the principles of a standard Gated Recurrent Unit (GRU).

$$\tilde{V}_{c_i} = V_{c_i}\Gamma \tag{5}$$

$$u_{c_i} = \phi\left(Z_u\tilde{V}_{c_i} + \Upsilon_u V_{Parent(c_i)}\right) \tag{6}$$

$$x_{c_i} = \phi\left(Z_{c_i}\tilde{V}_{c_i} + \Upsilon_x V_{Parent(c_i)}\right) \tag{7}$$

$$\widehat{V}_{c_i} = \tanh\left(Z_V\tilde{V}_{c_i} + \Upsilon_V\left(V_{Parent(c_i)} \odot u_{c_i}\right)\right) \tag{8}$$

$$V_{c_i} = (1 - x_{c_i}) \odot h_{Parent(c_i)} + x_{c_i} \odot \widehat{V}_{c_i} \tag{9}$$

In this context, c_i represents a post, and Γ represents the parameter matrix responsible for transforming this input post. The transformed representation of c_i is denoted as $\tilde{V}c_i$, and Z and Υ signify the weight connections within the GRU (Gated Recurrent Unit). Vc_i refers to the hidden state of c_i, and $\widehat{V}c_i$ represents the candidate activation of the hidden state of the current node. Similar to the standard GRU, $\odot$ indicates element-wise multiplication. The reset gate uc_i determines how to combine the current input $\tilde{V}c_i$ with the memory of children, while the update gate xc_i defines the extent to which memory from the children is incorporated into the current node.

The approach recursively explores the tree, gathering information and "*learning*" about the data point. This information gets condensed into a hidden vector for each leaf. However, the problem is that trees can have different numbers of leaves! This means we can't directly stick those hidden vectors into a neural network with a fixed number of inputs. To solve this, we use a special layer called "*max pooling*" (see Fig. 3). This layer scans all the leaf vectors and picks the highest value for each dimension. This way, we capture the most important features that were highlighted during the walk down the tree, regardless of how many leaves there are. Finally, it considers the best-of-the-best information and feeds it into another layer called "*softmax.*" This layer analyzes the pooled features and makes a prediction about the data point's category. In essence, it takes all the insights gathered from the top-down journey through the tree and uses them to make a confident call. This approach allows us to analyze variable-length trees and effectively leverage the learned features for accurate prediction, even when the journey down the tree can take different paths.

$$L_{d_i} = \text{Softmax}\left(\boldsymbol{W}V_{C_{d_i}} + \boldsymbol{\beta}\right) \tag{10}$$

The parameter $V_{C_{d_i}}$ represents the pooling vector over all leaf nodes, while the parameters W (weight) and β (bias) are instrumental in shaping the output layer.

Our initial investigation explored the feasibility of bottom-up propagation as an alternative information flow mechanism within the model. However, empirical evaluations revealed a substantial decrease in model performance. Analysis identified the overreliance on the root node's representation as the primary culprit for this information loss. This limitation stems from the inherent nature of bottom-up propagation, where information from lower levels aggregates towards the apex, potentially discarding refined features accumulated on diverse propagation paths.

A. Identifying Stance of Posts

Our goal for stance classification is to automatically identify the stance expressed in a social media post, whether it be an opinion or attitude. We can visualize this process as information diffusing through a tree-like structure, with each node representing a component of the post, such as words, phrases, or sentences. Now, think of information flowing down this tree. We assume this diffusion happens perfectly and simultaneously, ensuring all children receive the stance features of their parent. This allows us to build a representation of the overall stance within each "*binary classifier*" $H = E \times S$, which are essentially decision-making units at each leaf node. To achieve this, we used the newsPostTree representation learning approach [37]. This approach works like a translator, converting a node's raw information (including its own content and the stance received from its parent) into a "*context vector.*" This vector captures the gist of the stance expressed at that specific point. Imagine a node c_i, by combining the information from its parent node $Parent(c_i)$ and with its own content, c_i gets translated into a new context vector named $\widehat{V}_{c_i}$. This process repeats recursively as we travel down the tree, with each node's context vector building on the accumulated stance information from its ancestors. In essence, this newsPostTree approach allows us to understand the overall stance of a post by considering not just its individual parts but also the broader context and relationships built up through the hierarchical structure. This information-rich representation empowers the binary classifiers at each leaf node to make more accurate and refined decisions about the specific stance expressed in that part of the text.

$$\widehat{V_{c_i}^h} = \widehat{RNN}\left(V_{c_i}^h, V_{Parent(c_i)}^h, \alpha^h\right) \tag{11}$$

In the context of our model, $\widehat{\mathrm{RNN}}()$ signifies the RNN transition function based on top-down processing [37] of the newsPostTree, with α^h encompassing all associated parameters. Subsequently, a fully-connected softmax layer is employed to forecast the stance probability of c_i concerning the news vector $V_{d_i}^h$ in relation to classifier h:

$$p_{c_i}^h = \mathrm{softmax}\left(\delta_1^h \widehat{V_{c_i}^h} + \delta_2^h V_{d_i}^h + \omega^h\right) \tag{12}$$

In this formulation, δ_1^h, δ_2^h, and ω^h denote the weights and bias of the prediction layer. Subsequent to this, the stance probability for each individual post $p_{c_i}^h$ within the newsPostTree can be calculated using a methodology akin to that described in the above equation.

B. Post-Stance Aggregation

In our quest to assess fake news veracity, not all pieces of evidence hold equal weight. Consequently, it's crucial to selectively amplify the contributions of those pieces of evidence – the binary classifiers within the tree – that demonstrate the strongest correlation between the expressed stances and the actual news veracity. To achieve this, we enlist the power of an "*attention mechanism*" [38]. It carefully scans all the binary classifiers within the tree, analyzing their individual capabilities in capturing the link between stance and news veracity. Those classifiers exhibiting a stronger correlation indicate greater reliability as evidence sources. Conversely, classifiers with weaker correlations fade into the background, their contributions minimized.

$$V_a = GRU\left(t_{|d_i|}, V_{|d_i|-1}, \Delta_a\right) \tag{13}$$

$$\Lambda_h = \frac{\exp\left(V_a \cdot V_{d_i}^{h^\top}\right)}{\sum_h \exp\left(V_a \cdot V_{d_i}^{h^\top}\right)} \tag{14}$$

Δ_a symbolizes the internal configuration of the GRU encoder. $V_{c_i}^h$ represents the news encoded by the h-th classifier. To understand the different stances, our model relies on a collection of specialized "*binary stance classifiers,*" each focusing on a particular stance type. We aggregate all binary stance classifiers targeting the same stance type into a single set $K(S)$, where $s \in S$ can be support, deny, comment, or question. So, while they're all investigating the same type of news, they each have a unique perspective based on their specific training data. The final probability for that stance type is then computed using:

$$\widehat{p_{c_i,S}} = \sum_{h \in K(S)} \Lambda_h \cdot p_{c_i}^h \tag{15}$$

To assess the overall stance landscape, let $K(S)$ represent a team of classifiers focusing on a specific stance type $S \in \{deny, support, comment, question\}$. Each classifier within $K(S)$ outputs a prediction ($p_{c_i}^h$) the probability that the post c_i expresses the target stance S through the stance classifier h. Consequently, $p_{c_i,S}$ denotes the probability that post c_i is categorized as stance S, taking into account the significance of the corresponding classifiers with S as the target class. Despite addressing the same stance, these classifiers vary because each concentrates on a distinct veracity label of the news. By accumulating the weighted predictions for the specific stance S, we obtain the final probability distribution $\widehat{p_{c_i}}$. This process repeats for all possible stance types in S, ultimately painting a comprehensive picture of the different opinions and their relative strengths within the post.

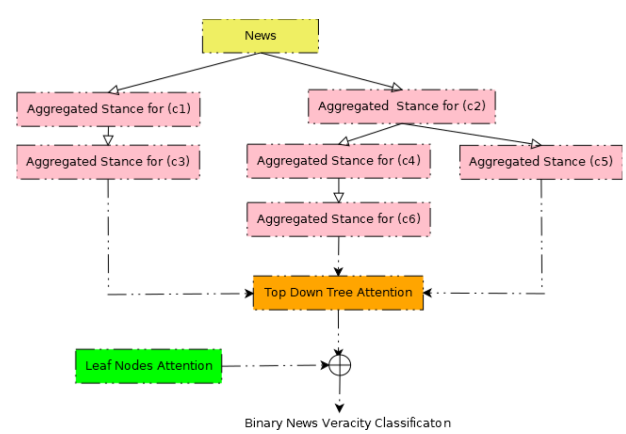


Fig. 4. An architecture of identifying fake news.

C. Identifying Fake News

In the context of verifying news authenticity, simply taking the overall sentiment of a discussion about the news might not be enough. We need to delve deeper, analyzing the individual stances expressed at different replying posts in the conversation and prioritizing those carrying greater evidential weight. To achieve this, we propose a "*newsPostTree Attention Mechanism*" that selectively aggregates stance information throughout the discussion tree (see Fig. 4). Think of the discussion as a branching tree, where each node represents a stance towards the news. Our model focuses on attending to specific nodes within each branch – the ones offering strong evidence for or against the news's veracity. We call these "*evidential stance nodes.*" Formally, let's call a path β_l through the tree from its root d_i (news) to a leaf node c_i. The set of all nodes along that path is denoted by C_{β_l}. To determine the overall aggregated stance for the path β_l, our model accomplishes this in two steps:

1) Selective attention: For each path, the model focuses on the evidential stance nodes. It assigns attention scores to each node, essentially highlighting the ones carrying the most significant evidence for or against the news's veracity. This selective attention mechanism ensures that the aggregated stance doesn't get diluted by less informative opinions.

2) Aggregation: Once the attention scores are calculated, the model combines the information from the evidential stance nodes for the path. This aggregation, using the attention scores as weights, produces a single "*aggregated stance*" for the path. This stance reflects the dominant sentiment, filtered through the lens of the most valuable pieces of evidence identified along the path.

$$\Theta^h_{c_i} = \frac{\exp\left(\widehat{V^h_{c_i}} \cdot V^{h^\top}_{d_i}\right)}{\sum_{i \in C_{\beta_l}} \exp\left(\tilde{V}^h_{d_i} \cdot V^{h^\top}_{d_i}\right)} \tag{16}$$

$$p^h_{\beta_l} = \sum_{n \in C_{\beta_l}} \Theta^h_{c_n} \cdot p^h_{c_n} \tag{17}$$

The symbol $\Theta^h_{c_i}$ denotes the attention coefficient assigned to each node along the path β_l, while $p^h_{\beta_l}$ represents the aggregated probability of stances from leaf nodes along the propagation path β_l. The symbiotic relationship between $\Theta^h_{c_i}$ and $p^h_{\beta_l}$ is pivotal, as it quantifies attention given to individual nodes and determines the collective stance probability of leaf nodes along the specific propagation path.

To capture the overall sentiment surrounding the news, we need to further aggregate information across different paths. Here's where the second stage of our Tree Attention Mechanism starts its work, focusing on informative path selection. Imagine each possible discussion about the news (a path through the tree) as a potential piece of evidence. We want to pick the paths that offer the most valuable insights into the news's veracity. To do this, we leverage the hidden vectors V^k_l embedded in the leaf nodes of each path (β_l). These vectors, generated in the first stage, encapsulate the accumulated stance information at the end of each branch.

Formally, let $\widehat{C_{d_i}}$ represent the set of all leaf nodes within the entire tree, with its root being the news itself. We now employ the Tree Attention Mechanism again, but this time on the set $\widehat{C_{d_i}}$. This second-stage attention assigns scores to each individual path based on the informativeness of its leaf node's hidden vector. Paths with stronger evidence, reflected in their leaf node representation, receive higher scores and are ultimately weighted more heavily in the final aggregation. This weighted aggregation, considering the attention scores for each path, culminates in an "*aggregated path stance.*" This stance represents the collective opinion about the news, filtered through the lens of the most informative paths identified by the second-stage attention mechanism. By prioritizing paths conveying robust evidence, we ensure that our final verdict on the news's veracity is not swayed by less insightful discussions or irrelevant tangents.

$$\Theta^h_{\beta_l} = \frac{\exp\left(\widehat{V^h_{\beta_l}} \cdot h^{h^\top}_{d_i}\right)}{\sum_{\beta_l \in \widehat{C_{d_i}}} \exp\left(\widehat{V^h_{\beta_l}} \cdot h^{h^\top}_{d_i}\right)} \tag{18}$$

$$\widehat{p}^h_{d_i} = \sum_{\beta_l \in \widehat{C_{d_i}}} \Theta^h_{\beta_l} \cdot p^h_{\beta_l} \tag{19}$$

The $\Theta^h_{\beta_l}$ signifies the attention coefficient for each individual leaf node, while $\widehat{p^h_{d_i}}$ encapsulates the probability associated with fake news veracity, derived from aggregating stances along various paths.

D. Aggregating Fake News Classifiers

Instead of analyzing each fake-news veracity classifier individually, our model employs a clever strategy to leverage their collective wisdom. Similar to stance classifiers aggregation, we gather all classifiers targeting the same "*fake news class label*" into a single set. Within each set, the members, which are individual binary veracity classifiers, provide their own assessments of the news's truthfulness. To combine these diverse insights, we utilize a "*weighted sum*" approach.

We used an indicator set $K(e_o)$ to group binary classifiers targeting the same veracity class e_{fake}, e_{real} in E. Each classifier within $K(e_o)$ provides a binary veracity prediction (fake or real) for the news item d_i, representing the probability $p^h_{d_i}$ of it belonging to veracity class e_o through the classifier h. Notably, although they share the same veracity target, classifiers in $K(e_o)$ differ in their assigned "*target stance classes.*" These stance classes represent specific sentiment or opinion types they analyze within the news. This differentiation leads to diverse stance probability predictions across classifiers, even for the same veracity class. By aggregating these stance-informed predictions, we obtain a richer representation of the news item's veracity. The final probability distribution over all veracity classes, denoted by $\widehat{p_{d_i}} = [\widehat{p_{d_i,fake}}, \widehat{p_{d_i,real}}]$, captures the combined insights from each classifier's unique perspective.

E. Training and Optimizing the Learning Model

To train our model using multiple binary classifiers, we convert the fine-grained veracity and stance annotations into binary labels. Each classifier targets a specific veracity-stance pair, where the veracity label of a news article d_i is mapped to either the target class or *others* (non-matching cases). Similarly, each post's stance is modeled as a probability distribution over the target stance class. This label transformation is applied uniformly across all classifiers, enabling them to distinguish between a specific veracity-stance combination and all other instances. While stance predictions are utilized during training, the ground truth for fake news verification focuses solely on the veracity of the news article, as post-level stance annotations are often sparse and potentially unreliable.

1) Training binary MIL-based classifiers: Our model hinges on the negative log likelihood (NLL) as the key metric for measuring prediction accuracy.

$$M = -\sum_{h=1}^{H}\sum_{d_i=1}^{D} L_{d_i} * \log \hat{p}_{d_i}^{h} + (1 - L_{d_i}) * \log\left(1 - \hat{p}_{d_i}^{h}\right) \quad (20)$$

For each news item d_i, let L_{d_i} represent the ground truth. $p_{d_i}^{h}$, on the other hand, signifies the predicted probability generated by classifier h for the news item d_i. This equation takes into account all D news items and H binary classifiers in the model, making it a powerful tool for evaluating how well predicted probabilities align with actual outcomes across the entire dataset.

2) Training aggregation model: To guide our aggregation model towards optimal performance, we leverage a powerful training technique called the "*negative log-likelihood loss function.*"

$$\widehat{M} = -\sum_{f=1}^{F}\sum_{e=1}^{E} L_{f,e} * \log \hat{p}_{f,e} + (1 - L_{f,e}) * \log\left(1 - \hat{p}_{f,e}\right) \quad (21)$$

The model produces a "*predicted probability*" ($p_{f,e}$) of the f news belonging to a specific veracity class $e \in E$. In this context, $L_{f,e}$ represents a binary value that signifies whether the veracity class of the f-th news, based on groundtruth, is e. Finally, E signifies the total number of possible veracity classes.

The negative log-likelihood loss function then steps in with its rigorous assessment. It calculates a penalty based on the discrepancy between the model's prediction and the actual truth. If the model confidently predicts the wrong class, the penalty is severe, urging it to adjust its reasoning. Conversely, a confident and accurate prediction earns minimal punishment. As the model processes numerous news items, it undergoes a continuous learning process guided by the penalties. It refines its ability to interpret and weigh the individual stance information, gradually honing its skill in predicting the overall veracity of the news with increasing accuracy. By minimizing the negative log-likelihood over countless news samples, the model effectively learns to distinguish fake news news from real news within the chosen framework of veracity classes.

To fine-tuning We used "*back-propagation*" [39], which meticulously analyzes the model's output and works backwards to optimize the model. To guide this optimization process, we leverage the "*Adam*" optimizer [40]. which maneuvering the learning rate (the pace of adjustments) to ensure the model progresses efficiently without getting stuck in overfitting or underfitting traps. We set the initial learning rate at 0.001, and it can dynamically adapt based on the model's performance. To infuse our model with rich linguistic understanding, we pre-train its word embedding layer using pre-trained "*GloVe Wikipedia 6B word embeddings*" [41]. These pre-trained vectors offer an initial advantage, enabling the model to swiftly understand the intricacies of language within the news it analyzes. The training process persists until two crucial conditions are satisfied:

- Convergence of the loss value: We monitor the "*loss function,*" a metric that quantifies the model's errors. When the loss stabilizes and stops decreasing significantly, it indicates that the model has likely reached its optimal performance.
- Maximum epoch number reached: In case the loss doesn't converge perfectly, we also set a pre-defined maximum number of training iterations ("*epochs*") as a safety net. This ensures the training process doesn't run indefinitely and allows us to evaluate the model's performance at various stages.

At the end the model applies the "*weighted aggregation model,*" which combines the insights from all individual classifiers to deliver a unified verdict on the news item's overall veracity.

VI. Experiments

For investigating the impact of news propagation patterns on fake news detection, we used the FakeNewsNet datasets [42], which are integral to understanding the dynamics of misinformation in online spaces. These datasets encompass source news that has undergone fact-checking by reputable sources such as PolitiFact and GossipCop, along with the associated tweets and social engagement data of users. The PolitiFact dataset comprises a total of 1,056 news articles, including 432 identified as fake news and 624 classified as real news articles. Similarly, the GossipCop dataset encompasses a total of 22,140 news articles, with 5,323 categorized as fake news and 16,817 as real news articles. Regarding the social context, PolitiFact observed 95,553 users posting tweets about fake news and 249,887 users posting tweets about real news. In the case of GossipCop, fake news generated 265,155 user tweets, while real news garnered 80,137 user tweets. The user social data is comprehensive, involving the crawling of user attributes and user-following relations, providing insights into how individuals interact with and disseminate information. The labels within the dataset categorize news as either "*real news*" or "*fake news*". To ensure a focused analysis, retweets are filtered out from the datasets, allowing a more clear examination of the primary propagation patterns.

We trained and tested our model on the SemEval-8 dataset [43]. SemEval-8 offers a rich tapestry of data, with each news item meticulously labeled as either "*true,*" "*fake,*" or "*unverified-fake.*" We consider the samples only labeled as either "*true*" or "*fake*". This veracity labeling goes beyond a simple binary classification, offering valuable insights into the spectrum of information trustworthiness. Moreover, each news item is accompanied by user posts categorized by their expressed stance: "*support,*" "*deny,*" "*question,*" or "*comment.*" While prior research on SemEval-8 often focused on a holistic understanding of rumor detection and stance prediction, our investigation took a more granular approach. We focused exclusively on news items categorized as true or fake, effectively framing the task as a binary veracity classification. This decision allowed us to hone in on the core challenge of distinguishing factually accurate news from misinformation. Furthermore, instead of simply predicting the overall stance category for each post, we opted for a more fine-grained analysis. We trained eight individual binary stance classifiers, one for each possible stance-veracity combination.

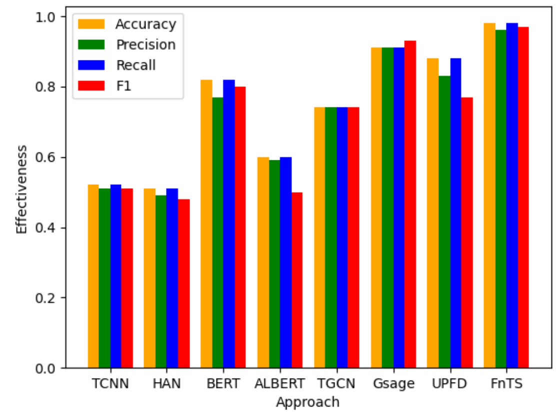
To optimize our model's performance, we held out 20 percent of the test data as a validation set. This allowed us to adjust hyperparameters precisely, ensuring the best possible performance. We further employed a rigorous set of evaluation metrics, including Accuracy, Precision, Recall, and F1 score (F1). These metrics, taken together, offer a comprehensive picture of how effectively our models distinguish real and fake news, while also capturing the fine details of stance detection. Furthermore, we built all our neural networks using the well-regarded PyTorch framework [44]. This ensured consistency and reliability throughout the computational aspects of the study.

A. Results

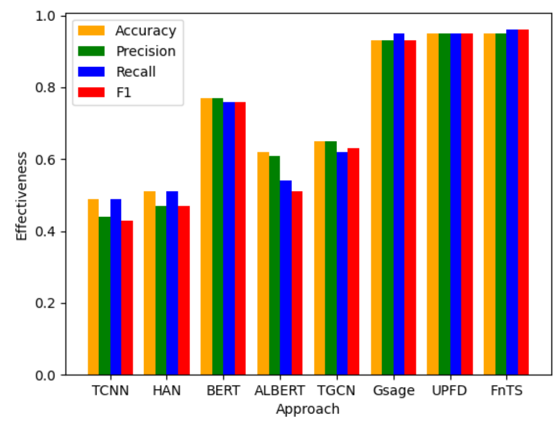
To evaluate the efficacy of any novel model demands a rigorous testing ground. For the proposed approach, we constructed a two-pronged evaluation strategy. First, we compare the performance of our proposed approach (**FnTS**) against several baseline approaches: TextCNN [45], HAN [46], BERT [47], ALBERT [48], TextGCN [49], GraphSage [50], and UPFD [51]. Second, we armed ourselves with a quartet of robust evaluation metrics: Accuracy, Precision, Recall, and F1-score (F1). These metrics collectively paint a clear picture of how effectively FnTS discerns real from fake news, capturing both its overall success rate and its ability to avoid both false positives and negatives.

According to Fig. 5, results on both datasets show that the proposed approach (FnTS) achieves the optimal score on all four metrics. This consistent performance across metrics and datasets underscores the strength of our approach, suggesting that considering the stance of posts truly makes a difference in the war against misinformation. Further investigation unveils the secret sauce behind FnTS's success: its top-down exploration of social networking posts. By delving deeper into the intricacies of social connections, the newsPostTree hierarchy exploration captures a richer tapestry of information compared to baseline methods. Specifically, the attention mechanism, a built-in spotlight highlighting vital post-stances, plays a crucial role in identifying and extracting distinctive user signals. These signals ultimately empower FnTS to make more informed and accurate fake news detection decisions.

Graph models, which utilize both textual and structural information, consistently surpass their counterparts that focus solely on text. Notably, GraphSage [50], powered by a customized news propagation graph, consistently demonstrates outstanding performance, outperforming all text-based methods. On the other hand, UPFD [51], another strong competitor, also uses user information as a tool. However, its reliance on historical data makes its performance vulnerable to fluctuations specific to the dataset, unlike the steadfast accuracy displayed by FnTS. To conclude, the comprehensive evaluation of FnTs paints a compelling picture of its effectiveness. Not only does it outperform or closely match baseline approaches, but it also sheds light on the power of incorporating user connections into fake news detection models. These findings pave the way for further advancements in leveraging social network dynamics to combat the ever-evolving landscape of misinformation.



(a) Politifact.

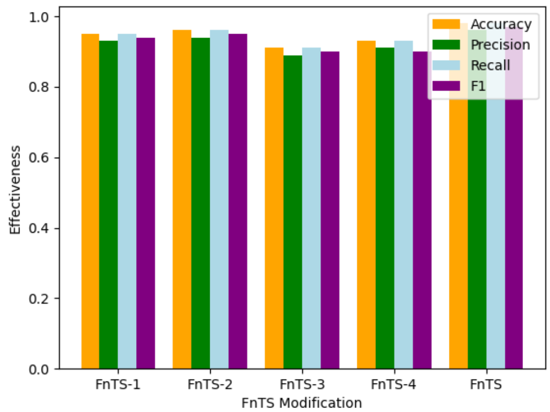


(b) Gossipcop.

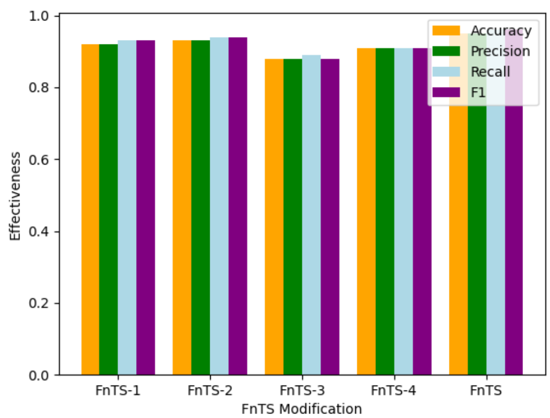
Fig. 5. Effectiveness of the proposed approach (FnTS) and baseline approaches on two benchmark datasets.

B. Component Study

To understand how each components of our proposed approach contributes to its success in fighting fake news, we conducted an experiment called an "*ablation study*" on FnTS with four modifications.



(a) Politifact.



(b) Gossipcop.

Fig. 6. Component studies of the proposed approach.

- FnTS-1: In this modification, we kept the tree-based attention mechanism but replaced the "*post encoder*" with non-structured post encoder.
- FnTS-2: We replaced the "*newsPostTree attention mechanism*" with a more common one (dot product) for stance aggregation.
- FnTS-3: In this modification we remove the recursive tree-based stance aggregation.
- FnTS-4: In this modification we remove the tree-based attention mechanism and replace the tree-based post encoder with a non-structural post encoder.

The results, shown in Fig. 6, indicate that FnTS-3 performs the worst, showing that the tree structure in both post encoding and stance aggregation is crucial. Without it, the model's ability to spot fake news drops significantly. FnTS-4 also stands out with the lowest performance scores, displaying substantial declines in accuracy, precision, recall, and F1 score for fake news verification. Changes to the tree-based attention mechanism (FnTS-2/FnTS-4) also hurt performance the most. This tells us that how the model focuses on its relevant parts while understanding their relationships through their tree structure is critical for its accuracy. In simpler terms, our experiment reveals that both the tree structure and the special way the model pays attention to different parts of the posts are essential for its success in detecting fake news. By understanding these key components, we can further improve our model and keep up with the ever-evolving world of misinformation.

TABLE I. ATTENTION SCORES OF BRANCHES OF NEWS-POST PROPAGATION TREE FOR CLASSIFYING FAKE NEWS

Branch	**Attention Score**
News− >Support− >Query− >Support	0.12
News− >Support− >Query− >Comment	0.07
News− >Support− >Support	**0.46**
News− >Deny− >Deny	**0.20**
News− >Comment− >Support− >Support	**0.21**
News− >Comment− >Comment	0.02
News− >Comment− >Support	0.04
News− >Query− >Comment	0.03
News− >Query− >Deny	0.02

TABLE II. ATTENTION SCORES OF BRANCHES OF NEWS-POST PROPAGATION TREE FOR CLASSIFYING REAL NEWS

Branch	**Attention Score**
News− >Comment− >Deny− >Support	0.12
News− >Comment	0.02
News− >Comment− >Deny	0.09
News− >Comment− >Deny− >Support	**0.15**
News− >Query− >Deny	0.09
News− >Deny− >Support	**0.43**
News− >Deny− >Support− >Comment	**0.23**

VII. DISCUSSION ON THE PROPOSED APPROACH

To peek into the inner workings of the proposed approach, we designed an experiment using the FakeNewsNet datasets [42]. We selected two trees from this dataset, one where the source news is "*Real*" and another where it's "*Fake*." Table I and Table II show how our model analyzes the comments and replies associated with each news, represented as branches and leaves in the "*trees*." The bold paths highlight the tree branches that are deemed most relevant for determining whether the news is actually true or false. By looking at the attention scores assigned to each post, we can uncover fascinating patterns:

1) Fake news: When the source news is labeled "*Fake,*" the model focuses heavily on comments that deny the news and then support it (branches with "*deny-support*" and "*deny-support-comment*"). This "*doubling back*" behavior suggests the model identifies these conflicting stances as suspicious clues.

2) Real news: On the other hand, for "*Real*" news, the model prioritizes comments that consistently support the news or deny the deny post (branches with "*support-support,*" "*deny-deny,*" and "*comment-support-support*"). This aligns with our intuition that consistent stances reinforce the legitimacy of the news.

3) Deny vs. Support: Interestingly, comments expressing denial play a more significant role when the news is classified as "*Fake*" (higher attention scores on denial branches). Conversely, support comments hold greater weight when the news is deemed "*Real.*"

4) Propagation paths: The branches trace the pathways through the trees that the model deems most influential in reaching its verdict. This highlights the importance of considering the structure of the discussion along with the individual comments themselves.

These observations confirm our belief that the "*tree propagation structure*" is vital for accurately recognizing fake news. By analyzing the relationships between comments and their stances, the proposed approach builds a nuanced understanding of online discourse, ultimately separating fact from fiction in the ever-evolving world of information.

VIII. Conclusion

This research presents a novel weakly supervised propagation model built upon the Multiple Instance Learning (MIL) framework. The proposed model is designed to tackle the dual challenge of simultaneously verifying the authenticity of fake news and identifying the stances expressed in associated posts. Crucially, our model achieves this feat using only bag-level annotations, specifically fake news veracity labels, demonstrating its ability to effectively learn from limited supervision. This unique approach empowers the model to collectively infer both the truthfulness of news and the previously unseen stance labels for individual posts within the associated propagation tree. The distinct tree-based stance aggregation mechanisms, deployed using newsPostTree configurations, are a key innovation of our work. This mechanism has yielded demonstrably superior performance for fake news identification tasks, surpassing the benchmarks established by state-of-the-art supervised and unsupervised models. The significance of our findings resides in the successful application of weakly supervised learning to significantly enhance the accuracy and efficiency of fake news verification in social media settings. The proposed framework opens several avenues for future work. One promising direction is extending the model to handle multi-modal data by incorporating images, videos, and metadata alongside textual and propagation features. Another important line of research is applying the framework to emerging real-time scenarios, such as early detection of misinformation during breaking news events. Additionally, improving interpretability through explainable AI techniques would make the system more transparent and valuable for journalists, fact-checkers, and policymakers.

Acknowledgment

This work was supported and funded by the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University (IMSIU).

References

[1] Z. Jin, J. Cao, Y.-G. Jiang, and Y. Zhang, "News credibility evaluation on microblog with a hierarchical propagation model," in *2014 IEEE International Conference on Data Mining*. IEEE, 2014, pp. 230–239.

[2] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of Economic Perspectives*, vol. 31, no. 2, pp. 211–36, May 2017. [Online]. Available: https://www.aeaweb.org/articles?id=10.1257/jep.31.2.211

[3] K. Shu, L. Cui, S. Wang, D. Lee, and H. Liu, "defend: Explainable fake news detection," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 395–405.

[4] J. Alghamdi, S. Luo, and Y. Lin, "A comprehensive survey on machine learning approaches for fake news detection," *Multimedia Tools and Applications*, vol. 83, no. 17, pp. 51 009–51 067, 2024.

[5] V.-H. Nguyen, K. Sugiyama, P. Nakov, and M.-Y. Kan, "Fang: Leveraging social context for fake news detection using graph representation," in *Proceedings of the 29th ACM international conference on information & knowledge management*, 2020, pp. 1165–1174.

[6] K. Pelrine, J. Danovitch, and R. Rabbany, "The surprising performance of simple baselines for misinformation detection," in *Proceedings of the Web Conference 2021*, 2021, pp. 3432–3441.

[7] R. K. Kaliyar, A. Goswami, P. Narang, and S. Sinha, "Fndnet–a deep convolutional neural network for fake news detection," *Cognitive Systems Research*, vol. 61, pp. 32–44, 2020.

[8] Y. Liu and Y.-F. Wu, "Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[9] X. Ge, S. Hao, Y. Li, B. Wei, and M. Zhang, "Hierarchical co-attention selection network for interpretable fake news detection," *Big Data and Cognitive Computing*, vol. 6, no. 3, p. 93, 2022.

[10] N. Rosenfeld, A. Szanto, and D. C. Parkes, "A kernel of truth: Determining rumor veracity on twitter by diffusion pattern alone," in *Proceedings of The Web Conference 2020*, 2020, pp. 1018–1028.

[11] B. Xie, X. Ma, J. Wu, J. Yang, and H. Fan, "Knowledge graph enhanced heterogeneous graph neural network for fake news detection," *IEEE Transactions on Consumer Electronics*, 2023.

[12] E. Min, Y. Rong, Y. Bian, T. Xu, P. Zhao, J. Huang, and S. Ananiadou, "Divide-and-conquer: Post-user interaction network for fake news detection on social media," in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 1148–1158.

[13] A. Jarrahi and L. Safari, "Evaluating the effectiveness of publishers' features in fake news detection on social media," *Multimedia Tools and Applications*, vol. 82, no. 2, pp. 2913–2939, 2023.

[14] J. Foulds and E. Frank, "A review of multi-instance learning assumptions," *The knowledge engineering review*, vol. 25, no. 1, pp. 1–25, 2010.

[15] R. Yang, J. Ma, H. Lin, and W. Gao, "A weakly supervised propagation model for rumor verification and stance detection with multiple instance learning," in *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, 2022, pp. 1761–1772.

[16] F. Olan, U. Jayawickrama, E. O. Arakpogun, J. Suklan, and S. Liu, "Fake news on social media: the impact on society," *Information Systems Frontiers*, vol. 26, no. 2, pp. 443–458, 2024.

[17] H. Zhang, Z. Fan, J. Zheng, and Q. Liu, "An improving deception detection method in computer-mediated communication," *Journal of Networks*, vol. 7, no. 11, p. 1811, 2012.

[18] F. Fkih, D. Rhouma, and H. Alghofaily, "A semantic approach for sarcasm identification for preventing fake news spreading on social networks," *International Journal of Information Technology*, pp. 1–19, 2024.

[19] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on twitter," in *Proceedings of the 20th international conference on World wide web*, 2011, pp. 675–684.

[20] E. J. Briscoe, D. S. Appling, and H. Hayes, "Cues to deception in social media communications," in *2014 47th Hawaii international conference on system sciences*. IEEE, 2014, pp. 1435–1443.

[21] M. Balaanand, N. Karthikeyan, S. Karthik, R. Varatharajan, G. Manogaran, and C. Sivaparthipan, "An enhanced graph-based semi-supervised learning algorithm to detect fake users on twitter," *The Journal of Supercomputing*, vol. 75, pp. 6085–6105, 2019.

[22] O. Ajao, D. Bhowmik, and S. Zargari, “Fake news identification on twitter with hybrid cnn and rnn models,” in *Proceedings of the 9th international conference on social media and society*, 2018, pp. 226–230.

[23] W. Y. Wang, “” liar, liar pants on fire”: A new benchmark dataset for fake news detection,” *arXiv preprint arXiv:1705.00648*, 2017.

[24] A. Zubiaga, M. Liakata, R. Procter, G. Wong Sak Hoi, and P. Tolmie, “Analysing how people orient to and spread rumours in social media by looking at conversational threads,” *PloS one*, vol. 11, no. 3, p. e0150989, 2016.

[25] S. M. TS and P. Sreeja, “Fake news detection on social media using adaptive optimization based deep learning approach,” *Multimedia Tools and Applications*, pp. 1–21, 2024.

[26] S. S. V. U, P. S. R, S. M, and S. R.B.V., “I-s2fnd: a novel interpretable self-ensembled semi-supervised model based on transformers for fake news detection,” *J. Intell. Inf. Syst.*, vol. 62, no. 2, p. 355–375, Oct. 2023. [Online]. Available: https://doi.org/10.1007/s10844-023-00821-0

[27] J. Y. Khan, M. T. I. Khondaker, S. Afroz, G. Uddin, and A. Iqbal, “A benchmark study of machine learning models for online fake news detection,” *Machine Learning with Applications*, vol. 4, p. 100032, 2021.

[28] K. Shu, H. R. Bernard, and H. Liu, “Studying fake news via network analysis: detection and mitigation,” *Emerging research challenges and opportunities in computational social network analysis and mining*, pp. 43–65, 2019.

[29] A. Bovet and H. A. Makse, “Influence of fake news in twitter during the 2016 us presidential election,” *Nature communications*, vol. 10, no. 1, p. 7, 2019.

[30] X. Zhou and R. Zafarani, “Network-based fake news detection: A pattern-driven approach,” *ACM SIGKDD explorations newsletter*, vol. 21, no. 2, pp. 48–60, 2019.

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

[32] G. L. Ciampaglia, P. Shiralkar, L. M. Rocha, J. Bollen, F. Menczer, and A. Flammini, “Computational fact checking from knowledge networks,” *PloS one*, vol. 10, no. 6, p. e0128193, 2015.

[33] N. Grinberg, K. Joseph, L. Friedland, B. Swire-Thompson, and D. Lazer, “Fake news on twitter during the 2016 us presidential election,” *Science*, vol. 363, no. 6425, pp. 374–378, 2019.

[34] X. Su, J. Yang, J. Wu, and Y. Zhang, “Mining user-aware multi-relations for fake news detection in large scale online social networks,” in *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, 2023, pp. 51–59.

[35] R. Dey and F. M. Salem, “Gate-variants of gated recurrent unit (gru) neural networks,” in *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*. IEEE, 2017, pp. 1597–1600.

[36] J. Zhang, J. Du, and L. Dai, “A gru-based encoder-decoder approach with attention for online handwritten mathematical expression recognition,” in *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, vol. 1. IEEE, 2017, pp. 902–907.

[37] J. Ma, W. Gao, and K.-F. Wong, “Rumor detection on twitter with tree-structured recursive neural networks.” Association for Computational Linguistics, 2018.

[38] A. de Santana Correia and E. L. Colombini, “Attention, please! a survey of neural attention models in deep learning,” *Artificial Intelligence Review*, vol. 55, no. 8, pp. 6037–6124, 2022.

[39] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, “Natural language processing (almost) from scratch,” *Journal of machine learning research*, vol. 12, no. ARTICLE, pp. 2493–2537, 2011.

[40] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[41] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[42] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, “Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media,” *Big data*, vol. 8, no. 3, pp. 171–188, 2020.

[43] L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga, “Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours,” in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, 2017.

[44] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” *Advances in neural information processing systems*, vol. 32, 2019.

[45] Y. Kim, “Convolutional neural networks for sentence classification,” *arXiv preprint arXiv:1408.5882*, 2014.

[46] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, “Hierarchical attention networks for document classification,” in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2016, pp. 1480–1489.

[47] J. D. M.-W. C. Kenton and L. K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.

[48] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, “Albert: A lite bert for self-supervised learning of language representations,” in *International Conference on Learning Representations*, 2019.

[49] L. Yao, C. Mao, and Y. Luo, “Graph convolutional networks for text classification,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 7370–7377.

[50] W. Hamilton, Z. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” *Advances in neural information processing systems*, vol. 30, 2017.

[51] X. Su, J. Yang, J. Wu, and Y. Zhang, “Mining user-aware multi-relations for fake news detection in large scale online social networks,” in *Proceedings of the sixteenth ACM international conference on web search and data mining*, 2023, pp. 51–59.