An Adaptive and Unsupervised Graph-Based Approach for Fake News Detection

Manish Vikrama[1][0009-0009-7019-4350] and Rashida Hasan[2][0000-0002-6231-8116]

[1] California State University Northridge, Los Angeles CA 91325, USA
manish-kumar.vikrama.731@my.csun.edu
[2] California State University Northridge, Los Angeles CA 91325, USA
rashida.hasan@csun.edu

Abstract. Detecting fake news has become a significant challenge in today's digital era, where false information spreads rapidly across various platforms. Current methods for fake news detection often overlook the social context of how news spreads and primarily rely on labeled datasets. As a result, they often struggle to effectively capture the nuanced relationships between news topics, entities, and content. This paper aims to address these challenges by proposing an unsupervised and adaptive framework that utilizes a variant of the Graph Sage Autoencoder to process heterogeneous graph data for improved fake news detection. Our approach captures complex relationships among articles, entities, and topics using adaptive data augmentation techniques like node feature masking and edge dropping, enhancing robustness and generalization. It utilizes SAGEConv layers within a contrastive learning framework to produce informative embeddings for node classification, incorporating adaptive feature masking and edge-dropping to selectively balance and refine the graph structure. Experimental results highlight the superiority of our approach compared to state-of-the-art algorithms, demonstrating improved detection accuracy and robustness while ensuring high scalability for fake news detection in complex information networks.

Keywords: Fake news detection · Heterogeneous subgraph · Graph sage autoencoder · Unsupervised learning · Natural language processing.

1 Introduction

In recent years, the propagation of fake news has posed serious challenges to the credibility of online sources of information. With social media's power to amplify content without rigorous fact-checking mechanisms, fake news can quickly reach millions of users, influencing public opinion and behavior. Therefore, it has become a major source for the spread of fake news[1]. With the rapid spread of misinformation on digital platforms and its detrimental effects on society, it is crucial to mitigate its impact on public discourse and protect individuals from misinformation that can lead to misguided decisions and societal divisions.

Currently, fake news detection methods primarily rely on feature rules, machine learning, and deep learning techniques. To offer a comprehensive understanding of the context for our work, we review two key areas: content-based and graph-based approaches for fake news detection. Content-based methods analyze writing style using NLP techniques to classify fake news based on linguistic, syntactic, and stylistic features. For example, Y. Kim [2] introduced a convolutional neural network (CNN) model to identify specific linguistic features in news content. Similarly, Horne et al. [3] analyzed variations in word choice and writing style between fake and real news, demonstrating how distinct language patterns can aid in detecting fake news. Recently, researchers have started integrating social context into fake news detection by analyzing relationships such as forwarding, comments, and user preferences, with Graph Convolutional Networks and Graph-based Label Propagation Networks emerging as popular approaches[4],[5]. These content-focused approaches often fall short in detecting subtle fake news that mimics the language and style of real news, resulting in suboptimal performance. This limitation stems from an insufficient exploration of the relationships between news articles and associated entities, such as topics and subjects. Additionally, these supervised methods depend heavily on large labeled datasets, which are time-consuming, labor-intensive, and require domain expertise to create[6].

To overcome these limitations of content-focused approaches, graph-based methods have recently gained attention. Wang et al. proposed a unified framework using a graph convolutional network to integrate textual data, knowledge concepts, and visual information for fake news detection[7]. Yao et al. [8] used a weighted graph for word-word relations and a GCN for news classification, while Hu et al. [9] employed a heterogeneous graph attention network, both relying on article parse trees but ignoring inter-article relations. Other studies explore relationships between news and users[10] or external knowledge sources[11], but their dependence on supplementary data sources poses considerable challenges. Some methods utilize Graph Auto-Encoders (GAE) to encode graphs into lower-dimensional spaces and reconstruct them efficiently, outperforming traditional models that rely on hand-crafted features[12].

Although previous methods have shown potential, they often struggle to effectively integrate inter-article relations and heterogeneous information. Another challenge with most methods is their reliance on labeled data, which is resource-intensive to acquire. Most of the methods often struggle to adapt to the evolving ways in which misinformation is crafted and shared, as they rely heavily on isolated text features without accounting for the broader relational context. However, most cannot seize the detailed relations between articles, entities, and topics [13], [14]. To tackle these challenges, we develop an adaptive and unsupervised framework for fake news detection. We construct a heterogeneous subgraph to model the relationships among articles, entities, and topics. The graph structure enables the analysis of subtle interactions within different components of a news piece, enhancing the detection of misleading information. We introduce a variant of the Graph Autoencoder (GAE) to learn embeddings that capture the semantic relationships in the graph. We leverage BERT-based embeddings to generate strong text representations while integrating adaptive data augmentation techniques with node feature masking and edge-dropping. Furthermore, we incorporate a contrastive learning loss to refine embedding quality by differentiating between positive and negative fake news articles. The contributions of this paper are as follows:

- Development of a Heterogeneous Subgraph Model: We construct a heterogeneous subgraph to represent the relationships among articles, entities, and topics, enabling a more comprehensive analysis of fake news detection.
- Graph Autoencoder (GAE) Variant: We introduce a variant of the Graph Autoencoder (GAE) to learn embeddings that effectively capture semantic relationships in the graph structure, enhancing the ability to detect misleading information. This variant is designed to handle unsupervised data, allowing for the detection of fake news without relying on labeled datasets.
- Adaptive Data Augmentation Technique: We utilize node feature masking and edge-dropping as adaptive augmentation techniques to capture the dynamic relationship in the graph.
- Experimental Evaluation: We conduct extensive experiments on two real-world datasets to demonstrate the effectiveness of our algorithm compared to the state-of-the-art.

The paper is organized as follows: Section2presents a detailed description of the proposed approach; Section 3discusses the experimental results and their implications, and Section 4 summarizes the key findings and suggests potential future directions.

2 Proposed Methodology

We construct a heterogeneous subgraph to capture the relationship and features among news articles, entities, and topics. The graph is encoded using SAGEConv layers and decoded using link prediction. Finally, we combine graph reconstruction and contrastive loss functions to classify news articles as real or fake based on learned representations.

2.1 Problem Statement

Let $G = \{V, L\}$ be a heterogeneous graph, where V represents nodes (news articles, entities, and topics), and L represents edges capturing their relationships. Here

$$V = \{\{a_i\}_{i=1}^{|A|}, \{e_i\}_{i=1}^{|E|}, \{t_i\}_{i=1}^{|T|}\} \tag{1}$$

is the set of nodes, where A, E, and T represent sets of articles, entities, and topics, respectively. Nodes a_i, e_i, $t_i$correspond to a specific news article, entity, and topic within each set. The edge set L includes links between nodes with types like articles-articles, articles-entity, and articles-topic, capturing both direct connections and shared characteristics. In this work, we aim to develop a graph autoencoder that learns latent features from unlabeled news to predict authenticity. Each news instance A_i in the dataset D=$\{$A1,A2,... An$\}$ is modeled as a graph based on its propagation. We define an unsupervised function f: D$\rightarrow$Y, where $Y \in \{F, R\}$ (Fake or Real News) to classify news.

2.2 Heterogeneous Graph Construction

Our heterogeneous graph includes three types of nodes and relationships :(i)article nodes, (ii)entity nodes, and (iii)topic nodes. Each article is represented as a node n_i. For a set of x articles, we define $Y = \{n_1, n_2, \ldots, n_x\}$, where $Y \subset V$. An edge (article-article) is established between two articles if they share some entities exceeding the average threshold or if they focus on the same topic. Entity nodes $e_i \in E$ represent named entities, such as people, organizations, or locations mentioned in the news articles. A news-entity edge is created to capture the relationship whenever an entity is referenced in a news article. We used Latent Dirichlet Allocation (LDA), a probabilistic topic modeling technique, to extract potential topics from a collection of news articles. LDA identifies k latent topics, with the optimal value of k determined empirically by evaluating topic coherence and perplexity scores. Each identified topic is represented as a topic node $t_i \in T$. An article-topic edge is created if a topic ranks among the top λ_t most relevant topics for a specific article, based on the LDA output. The pseudo-code for the heterogeneous graph is provided in Algorithm 1.

Algorithm 1 Construct Heterogeneous Graph

Input: Dataset with articles, entities, and topics.
Output: Heterogeneous graph G.

```
Initialize G.
for each row i in dataset do
    article_id ← "article_" + i_index
    Add article_id to G with type = "article" and text = i[statement].
    for each entity ∈ i[entities] do
        entity_id ← "entity_" + entity
        if entity_id ∉ G then
            Add entity_id to G with type = "entity";
        end if
        Add edge (article_id, entity_id) to G with type = "article-entity"
    end for
    for each topic ∈ i[topics] do
        topic_id ← "topic_" + topic
        if topic_id ∉ G then Add topic_id to G with type = "topic".
        end if
        Add edge (article_id, topic_id) with type = "article-topic".
    end for
end for
return G
```

2.3 Graph Autoencoder

Our Graph Autoencoder consists of two main modules: (i) The Encoder and (ii) The Decoder. The model constructs a heterogeneous graph from the dataset,

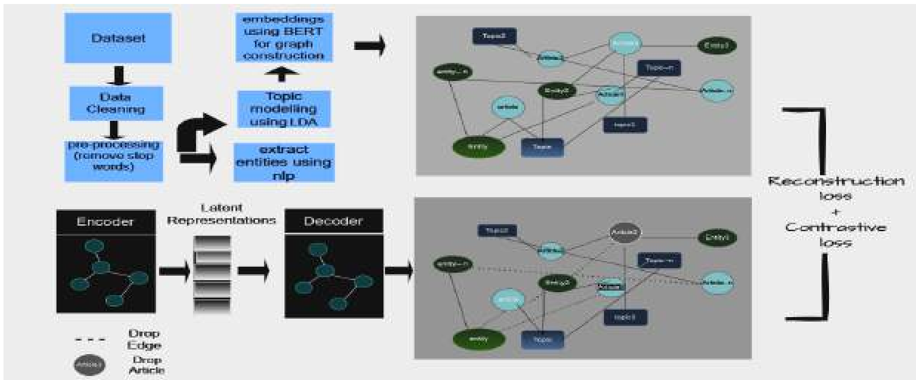


Fig. 1: General framework of the proposed approach

where nodes represent articles, topics, and entities, and are connected by edges. The graph undergoes augmentation through node feature masking and edge dropping, generating two perturbed versions. These augmented graphs are encoded into latent node representations, which the Decoder then uses to reconstruct the graph by predicting edge likelihoods. Training is guided by a composite loss function, combining reconstruction loss (for learning graph structure) and contrastive loss (for maintaining consistent representations across augmented views. Fig1 demonstrated the general framework of our proposed approach.

Graph Sage Encoder In GraphSAGE, node features are aggregated from neighboring nodes using a weight matrix W and an aggregation function. The hidden layer representation for each node is computed as follows:

$$h_{N(i)}^{(l+1)} = \text{aggregate}\left(\left\{h_j^{(l)}, \forall j \in N(i)\right\}\right) \tag{2}$$

$$h_i^{(l+1)} = \sigma\left(W \cdot \text{concat}\left(h_i^{(l)}, h_{N(i)}^{(l+1)}\right)\right) \tag{3}$$

$$h_i^{(l+1)} = \text{norm}\left(h_i^{(l)}\right) \tag{4}$$

where aggregate$(\cdot)$ combines features from neighbors, updating node embeddings. The aggregation function depends on the previous layer's representations and includes a nonlinearity $\sigma(\cdot)$.

Our encoder integrates HeteroConv layers with three relational dependencies: (i)article-entity and entity-article: to capture shared features between articles and entities, enhancing entity-driven context in news classification, (ii) article-topic and topic-article: Connect articles to topics, learning semantic themes, and (iii) Self-loop: Retain article-centric attributes for embeddings.

Decoder(link prediction) The decoder estimates the likelihood of a link using a neural network with two fully connected layers. The first layer applies a linear transformation to the concatenated node embeddings, followed by a ReLU activation. The second layer outputs the likelihood of the link. The forward function of the decoder takes a dictionary of node embeddings and an edge index dictionary. It concatenates the embeddings of source and destination nodes for each edge type (e.g., article-entity, article-topic) and passes them through the link predictor to generate positive edge logits. For hard negative sampling, non-existent edges are similarly formed by concatenating node embeddings and processed to produce negative edge logits. The output of the decoder, consisting of positive and negative edge logits, represents the predicted edges between pairs of nodes, supporting link prediction tasks on heterogeneous graphs.

2.4 Adaptive Augmentation

In our work, we employed two adaptive data augmentation techniques: node feature masking and edge dropping. We applied node feature masking to each node type in the graph using different masking rates, allowing us to vary the amount of information hidden for each node type. This introduces diversity into the training data, and we replaced any masked values with zeros for computational stability. For the adaptive edge dropping, We selectively drop edges from the graph based on a specified drop rate. This alters the graph's connectivity, helping the model generalize better by simulating missing or noisy connections. The pseudocode for the adaptive augmentation is depicted in Algorithm2

Algorithm 2 Adaptive Augmentation

Input: Graph G, initial rates for masking (R_{mask}) and dropping (R_{drop}), rate limits $[R_{\text{mask,min}}, R_{\text{mask,max}}]$, $[R_{\text{drop,min}}, R_{\text{drop,max}}]$, and adaptation factor α.
Output: Augmented graph G'.

for each epoch **do**
  Node Masking:
  for each node type n with features **do**
    Apply mask $R_{\text{mask}}[n]$ to X_n: $X'_n = X_n \cdot (1 - \text{mask})$.
  end for
  Edge Dropping:
  for each edge type e **do**
    Drop edges with rate R_{drop}.
  end for
  Adjust Rates:
  for each n **do**
    Update $R_{\text{mask}}[n]$ and R_{drop} within limits using α.
  end for
end for
return G'

2.5 Loss Function

We combine the Reconstruction Loss and Contrastive Loss to ensure accurate graph structure reconstruction while improving node representations. This combined loss helps the model effectively differentiate between real and fake news by capturing both structural and semantic features.

Reconstruction Loss: The reconstruction loss uses binary cross-entropy (BCE) to ensure accurate edge predictions. It penalizes misclassifications of positive and negative edges:

$$\begin{aligned}&\mathcal{L}_{\text{reconstruction}}\\&\quad=\sum_{\text{types}}\Big(\text{BCE}(\sigma(\text{pos\_edge\_logits}),1)+\text{BCE}(\sigma(\text{neg\_edge\_logits}),0)\Big)\end{aligned} \tag{5}$$

Here, the pos_edge_logits and neg_edge_logits refer to the predicted likelihood of actual and non-existent edges, respectively. The loss drives positive edges closer to 1 and negative edges to 0, helping the model identify real versus fake connections.

Contrastive Loss: The contrastive loss encourages node embeddings of positive pairs (connected by edges) to be closer than negative pairs by a specified margin. This helps to structure the latent space, making similar nodes cluster and pushing dissimilar nodes apart. The positive pairs are defined as actual edges, where the source node z_{src} and destination node z_{dst} should be close in the embedding space. The distance between them is given by:

$$\text{pos\_dist} = \|z_{\text{src}} - z_{\text{dst}}\| \tag{6}$$

For each positive pair, we select hard negative nodes—those close to the source but not connected by edges. These are chosen as the top k closest nodes in the embedding space, enhancing learning with challenging negative examples. The margin loss ensures that positive pairs are closer than negative pairs by a margin.

$$\mathcal{L}_{\text{margin}} = \max(0, \text{pos\_dist} - \text{neg\_dist} + m) \tag{7}$$

where neg_dist is the distance between negative pairs, calculated for each hard negative node.

Final Loss: The overall contrastive loss combines the binary cross-entropy loss on logits and the margin-based term. This encourages the model to produce embeddings that allow positive pairs to be distinguishable from negative pairs while minimizing reconstruction errors in the graph structure. Therefore, our total loss function is:

$$\mathcal{L} = \mathcal{L}_{\text{reconstruction}} + \mathcal{L}_{\text{contrastive}} \tag{8}$$

3 Experiments and Results

All of our experiments were conducted in Google Colab, utilizing the Tesla T4 GPU for accelerated computation, with Python 3.10. To ensure the reproducibility of our research, all our data, code, and datasets are available at https://anonymous.4open.science/r/Fake-news-detection-EA8E/.

3.1 Dataset

In our experiment, we evaluate the performance of our proposed model using two publicly available datasets: LIAR[15] and PolitiFact[16], both of which are widely used for fake news detection. LIAR is one of the largest fake news datasets, containing over 12,800 short news statements. The PolitiFact dataset contains news articles and statements related to U.S. politics, primarily sourced from the PolitiFact fact-checking website. The dataset contains 21,152 statements that are fact-checked by experts.

3.2 Experimental Setup

We split the data into 60% training, 20% validation, and 20% testing. We applied data augmentation with node feature mask rates of 30% for articles, 40% for entities, and 50% for topics, along with a 20% edge drop rate. We used an architecture with 768 input channels (node features), 256 hidden channels, and 64 output channels. The Adam optimizer was used with a learning rate of 0.001 and weight decay of 1×10^{-5}, while the ReduceLROnPlateau scheduler adjusted the learning rate based on validation performance. Adaptive augmentation parameters controlled feature masking and edge dropping with an adaptation factor of 0.01. We trained the model for up to 100 epochs with early stopping after 20 epochs without improvement.

We compare our proposed algorithm against several unsupervised fake news detection benchmarks including Majority Voting, Truth Finder[17], LTM[18], CRH[19], UFD[20], UFNDA[21], GTUT[22], $(UMD)^2$[23], and GAMC[24]. Majority Voting outputs the most frequently verified opinion for each news item, while TruthFinder iteratively assesses credibility based on source reliability. Methods like LTM and CRH use graphical models and credibility frameworks, whereas UFD, UFNDA, and $(UMD)^2$ leverage advanced deep learning and graph-based approaches to uncover hidden feature relationships and enhance news authenticity assessment.

3.3 Performance Analysis on LIAR Dataset

For the LIAR dataset, we compare our algorithm against five state-of-the-art unsupervised methods including Majority Voting, Truth Finder, LTM, CRH, and UFD. We evaluate our algorithm in two configurations: with adaptive augmentation and without augmentation. Table1 demonstrates the performance of our

Table 1: Performance analysis of the proposed approach vs. state-of-the-art methods on the LIAR Dataset

Methods	**Accuracy**	**Precision**	**Recall**	**F1**
Majority Voting	0.586	0.624	0.628	0.626
Truth Finder	0.634	0.650	0.679	0.664
LTM	0.641	0.654	0.691	0.672
CRH	0.639	0.653	0.687	0.669
UFD	0.759	0.766	0.783	0.774
Proposed Approach (with augmentation)	**0.810**	**0.796**	**0.838**	**0.818**
Proposed Approach (without augmentation)	**0.834**	**0.825**	**0.842**	**0.831**

approach (with and without adaptive augmentation) in comparison to state-of-the-art unsupervised methods on the LIAR dataset.

The proposed approach with adaptive augmentation outperforms state-of-the-art methods on the LIAR dataset, achieving the highest scores in accuracy (0.810), precision (0.796), recall (0.838), and F1 score (0.818). Compared to UFD, the second-best performer, the proposed method shows significant improvements, particularly in recall and F1, demonstrating its superior ability to detect fake news. Even without augmentation, our approach still outperforms all state-of-the-art methods. With an accuracy of 0.834, it surpasses UFD, the second-best method, by 7.5%. However, despite a slight decrease in accuracy (0.834 to 0.81) compared to without augmentation, the adaptive augmentation in our approach introduces benefits in terms of model robustness and feature representation. This approach leverages both the content and structural aspects of the data, improving the overall quality of node representations through targeted feature masking and edge dropping, thereby capturing nuanced patterns in news verification. Thus, while the non-augmented version achieves marginally higher accuracy, the augmented GraphSAGE model still provides a substantial edge in adaptability and generalization, making it a valuable approach in scenarios where resilience to data variation is key.

3.4 Performance Analysis on PoltiFact Dataset

We compare our algorithm against six state-of-the-art unsupervised methods, including Truth Finder, UFNDA, UFD, GTUT, UMD^2, and GAMC, on the PolitiFact dataset. Table2 depicts the performance of our approach (with and without adaptive augmentation) in comparison to state-of-the-art unsupervised methods on the PoltiFact dataset. From the results, it is evident that our approach outperforms traditional unsupervised methods across all four metrics. Specifically, our algorithm with augmentation shows a notable accuracy improvement of 3.6% over $(\text{UMD})^2$ and 2.4% over GAMC. Additionally, the without augmentation further enhances accuracy, achieving a 5.1% increase compared to $(\text{UMD})^2$ and a 3.0% improvement over GAMC.

Table 2: Performance analysis of the proposed approach vs. state-of-the-art methods on the PoltiFact Dataset.

Methods	Accuracy	Precision	Recall	F1
Truth Finder	0.581	0.572	0.576	0.573
UFNDA	0.685	0.667	0.659	0.670
UFD	0.697	0.652	0.641	0.647
GTUT	0.776	0.782	0.758	0.767
$(\text{UMD})^2$	0.802	0.795	0.748	0.761
GAMC	0.828	0.825	**0.817**	0.823
Proposed Approach (with augmentation)	**0.836**	**0.842**	0.811	**0.826**
Proposed Approach (without augmentation)	**0.853**	**0.851**	0.841	**0.841**

In summary, these results suggest that our graph-based approach, with or without augmentation, provides a more robust alternative to traditional unsupervised methods. While the model without augmentation achieves the highest scores, the adaptive augmentation process brings additional flexibility, particularly valuable in dynamic real-world fake news detection scenarios where the data characteristics are continually evolving. Thus, our algorithm not only achieves strong performance across both configurations but also contributes a versatile solution suited for varied data conditions.

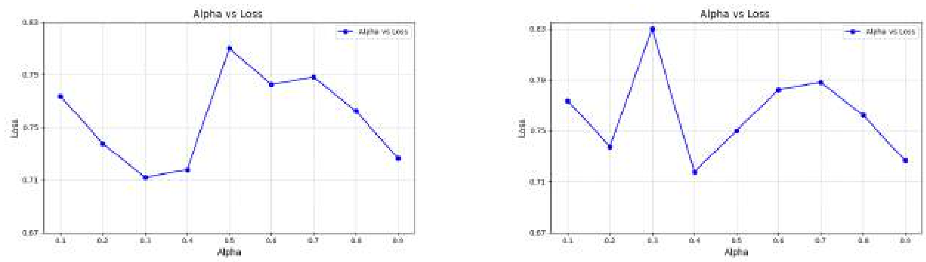
3.5 Parameter Evaluation

We conduct experiments by varying both the mask and drop rates within their effective range of 0.1 to 0.9 to examine their impact on model performance. The optimal balance between reconstruction and contrastive losses is achieved with $\alpha = 0.3$ on the Politifact dataset and $\alpha = 0.5$ on the Liar dataset, where the best results are obtained. For adaptive augmentation, we set the initial mask rates to 0.3 for Articles, 0.4 for Entities, and 0.5 for Topics. The initial drop rate is set to 0.2, and the rate boundaries for both masking and dropping are defined as [0.1, 0.5]. The adaptation factor is set to 0.01 to ensure gradual adjustments during training.

4 Conclusion

Our proposed graph-based approach effectively utilizes a Graph Autoencoder (GAE) to learn rich, meaningful embeddings from heterogeneous graph data, achieving strong performance in fake news detection. The integration of a contrastive predictor, along with reconstruction loss enhances the model's ability to generalize and ensures stable training. Negative sampling in the decoder further contributes to dataset balance and model robustness.

While our method demonstrates promising results, there are still avenues for future work, particularly in addressing challenges posed by AI-generated



(a) Analysis of α on LIAR dataset (b) Analysis of α on PoltiFact dataset

Fig. 2: Parameter Evaluation on Liar and PoltiFact Dataset

fake news and enabling real-time content verification through APIs. Exploring the use of Graph Attention Networks (GAT) or transformer models with supervised labels could enhance relationship detection. Additionally, applying transfer learning with pre-trained embeddings may help the model adapt to the evolving landscape of online content, offering opportunities to improve performance and applicability in real-world scenarios.

References

1. Wang, Z., Hou, D., Gao, C., Li, X., Li, X.: Lightweight source localization for large-scale social networks. In: Proceedings of the ACM Web Conference 2023, pp. 286–294. ACM, (2023)
2. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
3. Horne, B., Adali, S.: This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In: Proceedings of the International AAAI Conference on Web and Social Media, vol. 11, no. 1, pp. 759–766. AAAI, (2017)
4. Bian, J., Xiao, X., Xu, T., Zhao, P., Huang, W., Rong, Y., Huang, J.: Rumor detection on social media with bi-directional graph convolutional networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 549–556 (2020)
5. Yuan, C., Ma, Q., Zhou, W., Han, J., Hu, S.: Jointly embedding the local and global relations of heterogeneous graph for rumor detection. In: 2019 IEEE International Conference on Data Mining (ICDM), pp. 796–805. IEEE, November (2019).
6. Mehta, D., et al.: Exploring the efficacy of natural language processing and supervised learning in the classification of fake news articles. Advances of Robotic Technology, vol. 2, no. 1 (2024)
7. Wang, Y., Qian, S., Hu, J., Fang, Q., Xu, C.: Fake news detection via knowledge-driven multimodal graph convolutional networks. In: Proceedings of the 2020 International Conference on Multimedia Retrieval, pp. 540–547. (2020)
8. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 7370–7377. AAAI, (2019).

9. Hu, L., Yang, T., Shi, C., Ji, H., Li, X.: Heterogeneous graph attention networks for semi-supervised short text classification. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 4821–4830. (2019)
10. Su, X., Yang, J., Wu, J., Zhang, Y.: 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, pp. 51–59. ACM, (2023)
11. Hu, L., Yang, T., Zhang, L., Zhong, W., Tang, D., Shi, C., Duan, N., Zhou, M.: Compare to the knowledge: Graph neural fake news detection with external knowledge. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 754–763. (2021)
12. Kipf, T.N., Welling, M.: Variational graph auto-encoders. arXiv preprint arXiv:1611.07308 (2016)
13. Dou, Y., Shu, K., Xia, C., Yu, P.S., Sun, L.: User preference-aware fake news detection. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2051–2055. ACM, (2021)
14. Su, X., Yang, J., Wu, J., Zhang, Y.: 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 (WSDM), pp. 51–59. ACM, (2023)
15. Wang, W.Y.: "Liar, liar pants on fire": A new benchmark dataset for fake news detection. arXiv preprint arXiv:1705.00648 (2017)
16. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H.: Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. Big Data, 8(3), 171–188 (2020)
17. Yin, X., Han, J., Yu, P.S.: Truth discovery with multiple conflicting information providers on the web. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1048–1052. ACM, (2007)
18. Zhao, B., Rubinstein, B.I.P., Gemmell, J., Han, J.: A Bayesian approach to discovering truth from conflicting sources for data integration. arXiv preprint arXiv:1203.0058 (2012)
19. Li, Q., Li, Y., Gao, J., Zhao, B., Fan, W., Han, J.: Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, pp. 1187–1198. ACM, (2014)
20. Yang, S., Shu, K., Wang, S., Gu, R., Wu, F., Liu, H.: Unsupervised fake news detection on social media: A generative approach. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, no. 01, pp. 5644–5651. (2019)
21. Gangireddy, S.C.R., P, D., Long, C., Chakraborty, T.: Unsupervised fake news detection: A graph-based approach. In: Proceedings of the 31st ACM Conference on Hypertext and Social Media, pp. 75–83. ACM, (2020)
22. Li, D., Guo, H., Wang, Z., Zheng, Z.: Unsupervised fake news detection based on autoencoder. IEEE Access, 9, 29356–29365 (2021)
23. Silva, A., Luo, L., Karunasekera, S., Leckie, C.: Unsupervised domain-agnostic fake news detection using multi-modal weak signals. IEEE Transactions on Knowledge and Data Engineering (2024)
24. Yin, S., Zhu, P., Wu, L., Gao, C., Wang, Z.: GAMC: An unsupervised method for fake news detection using graph autoencoder with masking. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, no. 1, pp. 347–355. (2024)