Journal of
Information Technology,
Cybersecurity, and Artificial Intelligence

(RESEARCH)

FNDMFF: Enhancing Fake News Detection via Multi-Source Feature Fusion Framework

Dr. Hani Iwidat
Assistant. Professor; Al-Istiqlal University,

Journal of Information Technology, Cybersecurity, and Artificial Intelligence, 2026, 3(2), 39–58

Article DOI: https://doi.org/10.70715/jitcai.2026.v3.i1.055

Abstract

The rapid spread of fake news on social media platforms has emerged as a major societal challenge, weakening public trust, intensifying polarization, and threatening democratic processes worldwide. Current approaches to identifying fake news frequently overlook the comprehensive utilization of layered characteristics or fail to capture advanced relational patterns during information dissemination. In response, we introduce FNDMFF, a framework designed to improve fake news identification by integrating structural, temporal, and textual features through a specialized gating mechanism for multi-source integration. The process begins by extracting dissemination structures via an improved hypergraph neural network, followed by analyzing temporal patterns with a variable-scale timing component, and concludes by extracting textual details using a multi-head attention approach. In particular, FNDMFF includes a gating unit for feature fusion that adaptively modifies the importance of various feature aspects, allowing for the seamless integration of diverse data types. Experiments on the Politifact and Twitter16 datasets reveal that FNDMFF surpasses established models such as UPFD, HGNN, and the state-of-the-art RTRUST method. Our system delivers a 3.63% boost in accuracy and a 3.40% gain in F1-score for Politifact, alongside 0.54% and 0.55% enhancements for Twitter16. This highlights the value of fusing multiple data sources for detecting misinformation.

Keywords: Fake news detection, Hypergraph neural network, Multi-source feature fusion, Propagation graph, Attention mechanism.

1. INTRODUCTION

News creation and sharing have undergone significant transformations due to the swift growth of mobile online access and the widespread adoption of social platforms. While social media allows for remarkably fast information dissemination, the rise of false information has become a major societal challenge. Fake news is defined as inaccurate content intentionally fabricated by creators with deceptive intent to deceive audiences [1]. Recent advances in generative AI models [2] have significantly lowered the barrier to producing synthetic misinformation, posing more severe challenges to social stability. According to recent systematic reviews, misinformation and AI-driven fake news have emerged as the preeminent short-term threats to global stability, significantly eroding democratic deliberation and institutional trust. Furthermore, empirical studies on deepfake technology demonstrate that fabricated content has intensified social polarization and public mistrust [3].

Fake news detection primarily considers structure, timing, and content. The characteristics of these three sources are as follows: structural features include the dissemination path, depth, and breadth of fake news; typical dissemination network structures include rapid spread through a few "opinion leader" nodes and the formation of highly concentrated dissemination groups. Temporal features mainly depict the dynamic changes in the spread of fake news, including short-term explosive dissemination exhibiting abnormally high dissemination speed and interaction volume. Content features focus more on the textual information of the news itself, including exaggerated headlines and descriptions. The multidimensional complexity of these characteristics poses a significant challenge to fake news detection.

Current mainstream fake news detection methods primarily utilize content and dissemination relationship information. For instance, models apply BERT encoders to better extract semantic representations from articles, which boosts overall detection accuracy. Other approaches involve deep learning tools such as CNNs for local patterns or RNNs for broader sequences in text, each aiding in improving classification outcomes. Additionally, hypergraph neural networks can model higher-order dependencies in multivariate relationships, further improving performance by accurately capturing the complex connections between news, users, and social interactions. However, existing methods have two major limitations: (1) most rely on only one type of feature, and (2) they underexploit the structural and temporal patterns in the news dissemination process.

The research problem addressed in this study is the inability of current fake news detection methods to fully integrate structural, temporal, and content features, which results in limited detection performance because most approaches rely on only one feature type and fail to exploit dissemination patterns effectively. To solve this problem, the objectives of this work are: (1) to extract and model high-order structural relationships, dynamic temporal patterns, and rich semantic content using specialized modules; (2) to design an adaptive multi-source feature fusion mechanism that dynamically balances the importance of each feature type; and (3) to evaluate the resulting framework on benchmark datasets to achieve state-of-the-art accuracy and robustness. The main contributions of this work are threefold. First, we introduce a hierarchical multi-level attention framework together with an adaptive fusion module that integrates structural, temporal, and content features. Second, we develop three specialized extraction modules: an enhanced hypergraph neural network for structural modeling, a multi-scale temporal module for propagation dynamics, and a multi-head self-attention mechanism for semantic extraction. Third, extensive experiments on the Politifact and Twitter16 datasets demonstrate that FNDMFF achieves state-of-the-art performance on two public datasets, Politifact and Twitter16, verifying the effectiveness and robustness of the method.

The remainder of this paper is organized as follows. Section 2 reviews related work on fake news detection. Section 3 presents the proposed FNDMFF framework in detail. Section 4 evaluates its performance on two public datasets. Section 5 concludes the paper and discusses directions for future research.

2. Literature Review

The literature on automated fake news detection has evolved along three major axes: content-based analysis, graph-structural modelling, and temporal propagation dynamics. The following subsections review representative works in each area, identifying their respective contributions and limitations.

2.1. Fake News Detection Based on News Content

Existing fake news detection research is primarily based on machine learning and deep learning methods, using textual content to distinguish between real and fake news. This is because linguistic characteristics (such as lexicon, syntax, and grammatical structure) can directly reflect the credibility of news. Among them, machine learning-based fake news detection research usually requires complex feature engineering, which is labor-intensive and often fails to capture the full breadth of available features, thus reducing overall detection performance. On the other hand, deep learning effectively alleviates the shortcomings of machine learning methods. Alghamdi et al. [4] used pre-trained models such as BERT to fully capture the semantic information and long-distance dependencies of news, effectively improving detection performance. Raza et al. [5] designed a framework based on the Transformer architecture that captures long-distance dependencies and complex contextual relationships, incorporating metadata (auxiliary information in news content and social background) to improve detection performance. Ruchansky et al. [6] further improved detection performance through the multimodal fusion of news text, user reactions, and promoter-user features. . Shu et al. [7] introduced a collaborative attention network [8] that jointly models news content and user comments, improving both performance and explainability.

2.2. Fake News Detection Based on Graph Neural Networks

Deep learning on graph-structured data is a promising research field [9], Graph Neural Networks (GNNs) [10] can effectively improve the quality of node representation through neighborhood aggregation and information transmission. Graph Convolutional Networks (GCNs) are neighborhood aggregation methods based on averaging, which can quickly mine topological information and complex features [11]. Graph Attention Network (GAT) can dynamically learn the weight distribution of neighbors during aggregation. Li et al. [12] designed a Multi-Relational Graph Attention Networks (MRGAT), which optimized the network structure representation by integrating the self-attention mechanism into the network with different node weights. Wu et al. [13] introduced the Phrase Dependency Relational Graph Attention Network (PD-RGAT) constructs a relation graph based on phrase dependency graphs. By aggregating directed dependency edges and phrase information, and using GAT with multiple attention heads to model multiple relation edges and nodes, it can capture deep information of text from multiple dimensions and different levels, thereby significantly improving the performance of text classification.

DOI: https://doi.org/10.70715/jitcai.2026.v3.i2.055

Fake news typically propagates through distinctive structural patterns, such as rapid diffusion via a small number of opinion-leader nodes or the formation of dense dissemination clusters. Traditional deep learning methods usually **employ** the textual features of news, ignoring dissemination structural features. GNN methods can more effectively capture key structural information by directly leveraging graph structure data including dissemination paths, node relationships, and network topology. Han et al. [14] designed a dissemination-based fake news detection method using GNN to distinguish between different dissemination patterns of real and fake news on social media. Other studies based on GNN modeling of news propagation trees, such as BiGCN [15], GCNFN [16], and UPFD [17] effectively combine news semantics with graph structure (user interactions, timing, and entity relationships), providing a more comprehensive representation. Ren et al. [18] used a Hierarchical Graph Attention Network (HGAT) in a Heterogeneous Information Network (HIN) to learn and classify news node representations, simultaneously accounting for the content of different node types and the influence of varying connection patterns.

In the real world, the relationships between different objects are much more complex than pairwise relationships, which prevents traditional graph neural network methods from effectively modeling the relationships between objects. The introduction of hypergraph neural networks allows for the modeling of more complex high-order relationships between objects [19, 20], thus making them widely used in recommender systems [21], image annotation [22] and document classification [23]. By modeling complex forwarding cascades and high-order relationships, hypergraph neural networks effectively capture both propagation structure and semantic relationships.

To address the aforementioned issues, this paper extracts news features across three dimensions: structure, temporal sequence, and content. A hierarchical attention mechanism is employed to achieve dynamic interaction between features from different sources. Regarding structural features, an enhanced hypergraph neural network approach is used to effectively extract high-order relationship features of shared news fragments among users. For temporal features, multi-scale temporal modeling is employed to better capture the unique temporal characteristics of fake news dissemination. In terms of content feature extraction, a multi-head self-attention mechanism is utilized to extract key semantic information from different semantic perspectives, thereby further improving the extraction efficiency.

2.3. Fake News Detection Based on Temporal Propagation Modeling

The temporal dimension of news dissemination provides important discriminative signals for fake news detection, as fake news typically exhibits abnormal propagation patterns such as explosive short-term repost surges followed by rapid decay. Ma et al. [24] proposed tree-structured Recursive Neural Network (RvNN) models using bottom-up and top-down propagation trees to capture the sequential patterns of rumor spread on social media, demonstrating that temporal propagation structure carries strong detection signals. Song et al. [25] further introduced the Temporally Evolving Graph Neural Network (TGNF), which models the news propagation graph as a continuous-time dynamic diffusion network by fusing structural, semantic, and temporal features through graph attention, thereby improving detection performance over static graph-based methods. However, these methods rely on single-scale temporal attention and lack adaptive multi-source feature fusion, limiting their ability to comprehensively model complex fake news propagation. Recent advancements, such as dynamic temporal networks [26], have begun to address these multi-scale temporal complexities by tracking real-time dissemination patterns.

A summary of the previous studies discussed in Sections 2.1, 2.2, and 2.3 is presented in Table 1.

Table 1. Summary of representative studies in Fake News Detection providing a concise overview of each work's objective, methodology, and principal findings to contextualize the contributions of the present study.

Study (Ref)	Title / Key Method	Objective	Methodology	Key Findings / Limitations
[4]	Towards COVID-19 fake news detection (BERT transformer)	Detect fake news during pandemics	Pre-trained BERT and CT-BERT models	Captures semantic information and long-distance dependencies effectively / Limitation: content-only
[5]	Fake news detection based on news content and social contexts (Transformer)	Incorporate content + social metadata	Transformer architecture with metadata	Improves detection by modeling complex contextual relationships / Limitation: mainly content-focused

[6]	CSI: A hybrid deep model	Multimodal fusion (text + user reactions)	Hybrid deep model (Capture-Score-Integrate)	Significantly improves performance through multimodal integration / Limitation: single-feature reliance
[7]	dEFEND: Explainable fake news detection (co-attention)	Provide explainable detection	Collaborative attention network (content + comments)	Enhances both accuracy and explainability / Limitation: content + comments only
[14]	Graph neural networks with continual learning	Model dissemination patterns on social media	GNN with continual learning	Effectively distinguishes real vs. fake propagation patterns / Limitation: no high-order relations
[15]	Bi-directional GCN (Bi-GCN)	Capture propagation + dispersion in rumors	Bi-directional Graph Convolutional Networks	Captures bidirectional propagation effectively / Limitation: pairwise relations only
[16]	Fake news detection using geometric deep learning (GCNFN)	Apply geometric deep learning to social media	Geometric GNN (GCNFN)	Strong performance on social media datasets / Limitation: limited to pairwise graphs
[17]	User preference-aware fake news detection (UPFD)	Incorporate user preferences	GNN with user preference modeling	Provides comprehensive representation combining semantics and graph structure / Limitation: no temporal modeling
[18]	Fake news detection on heterogeneous networks (HGAT)	Detection in heterogeneous information networks	Hierarchical Graph Attention Network (HGAT)	Learns rich node representations from heterogeneous connections / Limitation: no adaptive fusion
[12]	Multi-relational Graph Attention Networks (MRGAT)	Optimize network structure representation for relational data	Multi-relational GAT with self-attention mechanism	Integrates self-attention for different node weights; improves relational modeling / Limitation: originally designed for knowledge graphs, not fake news propagation
[13]	Phrase Dependency Relational Graph Attention Network (PD-RGAT)	Aspect-based sentiment analysis using relational graphs	GAT with multiple attention heads on phrase dependency graphs	Captures deep text information from multiple dimensions and levels / Limitation: focused on sentiment; adapted here for structural modeling
[24]	Rumor Detection with Tree-structured RvNN	Capture sequential propagation patterns	Tree-structured Recursive Neural Networks	Temporal structure carries strong discriminative signals / Limitation: single-scale temporal
[27]	Temporally evolving graph neural network (TGNF)	Model dynamic diffusion over time	Temporally Evolving GNN with graph attention	Outperforms static methods by fusing structural, semantic, and temporal features / Limitation: lacks multi-source fusion
[26]	Tracing truth: dynamic temporal networks (DTN)	Multi-modal fake news detection with dynamic temporal	Dynamic temporal networks	Addresses multi-scale temporal complexities in real-time propagation / Limitation: weak on high-order structural patterns
[28]	Nothing stands alone: Relational fake news detection with hypergraph neural networks	Relational fake news detection using higher-order relations	Hypergraph Neural Networks (HGNN)	Effectively models high-order relationships between news items and users; foundational for enhanced hypergraph approaches / Limitation: requires careful hyperedge construction

DOI: https://doi.org/10.70715/jitcai.2026.v3.i2.055

3. METHODOLOGY

3.1. Hypergraph and Propagation Tree

The identification of fake news requires not only the extraction of content features but also the robust modeling of propagation paths. In order to capture these dynamics, this paper constructs a hypergraph coding framework based on the propagation tree structure of news nodes. The coding structure of its propagation tree and hypergraph is shown in Figure 1. This section compares the definitions of propagation tree (a) and hypergraph (b) [28] and puts forward specific problem statements based on these structures. First, the definition of propagation tree is as follows:

Given a total of N news items, denoted as $X = \{X_i\}_{i=1}^{N}$, and user engagement associated with each news item, $U_i = \{U_{t_j}\}_{j=1}^{|U|}$, where U represents the timestamp sequence of user engagement (e.g., posting and forwarding news). The propagation tree is defined as $P_j = (\{x_i\}, U_i, A)$, where A is the adjacency matrix inferred over time.

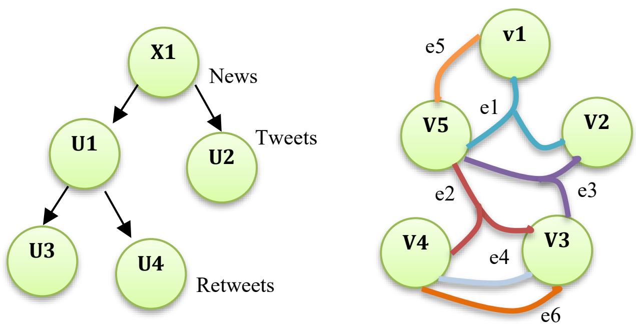


Figure 1. Propagation tree and hypergraph coding structure diagram. [28]

Propagation trees are used to represent individual news snippets and user engagement. This paper introduces a hypergraph to model the relationships between multiple news snippets. The definition of a hypergraph is as follows:

A hypergraph H can be defined as an ordered pair $H = (V, E)$, where V is the set of vertices (or nodes) and E is the set of hyperedge, where each hyperedge $e \in E$ is a non-empty element of the vertex set V. A hypergraph can be represented by an incidence matrix $H \in R^{NxM}$, defined as: [28]

$$H_{i,j} = \begin{cases} 1 & \text{if } v_i \in e_j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Fake news exhibits multidimensional complexity, yet most existing detection methods rely on only one category of features and thus struggle to capture its full complexity. To address the multidimensional complexity of fake news, we formally define the fake news detection task as follows:

Given a dataset $D = d_1, d_2, \dots, d_n$, where n represents the number of news items. $d_i = [d_i^1, d_i^2, \dots, d_i^m]$ represents the d news item in the dataset, where represents the j^{th} d_i^j feature value of news d, and m represents the feature hierarchy of news item d_i, which is greater than or equal to 2.

The multi-source feature fusion-enhanced fake news detection task learns a model $F: d_i \in D \to Y$, classifying news items into corresponding labels $Y \in \{0,1\}$, where Y=0 indicates the news is real and $Y = 1$ indicates the news is fake.

3.2. FNDMFF Overall Framework

This paper proposes the FNDMFF method consisting of three core components: propagation tree encoding, multi-source feature extraction, and news classification. The overall architecture is shown in Figure 2. First, a propagation tree is encoded, and GNNs are used to extract node features, which are then input into the multi-source feature extraction component. To comprehensively capture the multi-source features of news, the multi-source feature extraction component utilizes a multi-level encoding network, addressing structural, temporal, and content-related aspects.

Feature extraction is performed on each source independently before fusion, and a specially designed multi-source feature fusion gating unit is introduced to dynamically adjust the weights of different feature dimensions, achieving

DOI: https://doi.org/10.70715/jitcai.2026.v3.i2.055

efficient fusion of multi-source heterogeneous features. Specifically, structural features encode higher-order relationships between news items using an enhanced hypergraph neural network, revealing the complex propagation structure between news segments; temporal features extract propagation patterns at different time scales through temporal position encoding and a multi-scale time window; and content features capture key semantic information through a multi-head self-attention mechanism. Finally, the fused higher-order feature representation is input into a news classification component, which outputs detection probabilities through a fully connected neural network and a sigmoid activation function, classifying news authenticity into "true" and "false" types.

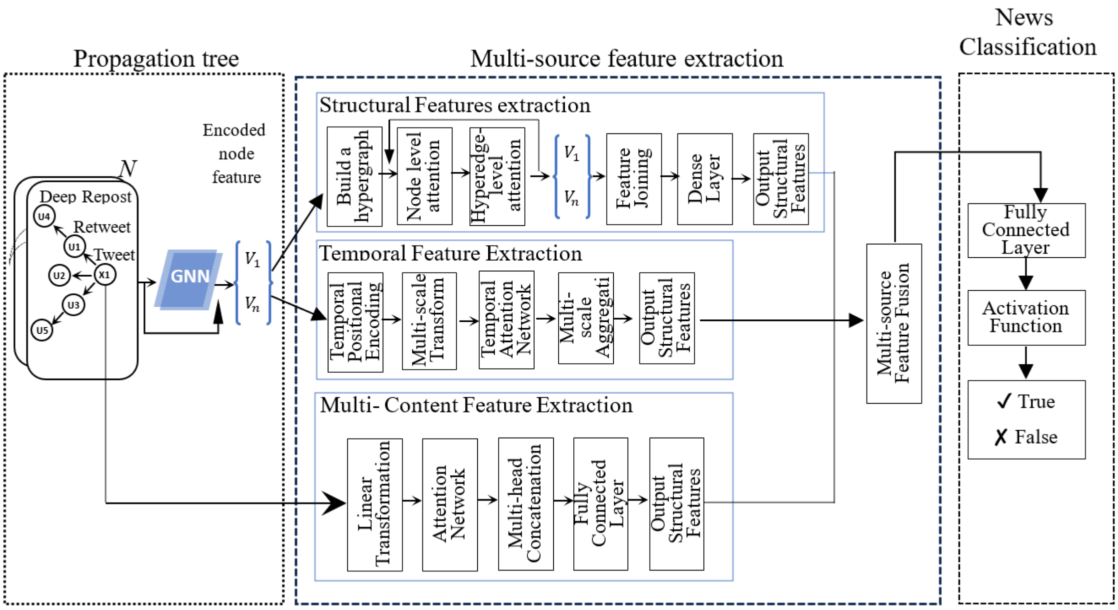


Figure 2. Overall framework diagram of FNDMFF

3.3. Encoding Propagation Tree

In terms of propagation tree encoding, this paper extracts the structural features of news propagation through GNN and generates a vectorized representation of the propagation pattern of news in social networks. First, given a set of propagation trees $P = \{P_i\}_{i=1}^{N}$, GNN is used to encode each propagation tree at the level of news propagation structure and temporal structure. In order to obtain the propagation pattern of news, this paper extracts the root representation ∈ RF of the propagation tree P_i, where F is the size of the input feature matrix. Next, the original news feature $\bar{x_i} \in R^F$ is connected by skip connections to generate a news representation with enhanced propagation pattern. Then, it is mapped to $v_i^0 \in R^d$ as the initial node representation of the hypergraph, where d is the size dimension of the hidden layer, and the formula is [28].

$$\overline{x_i} = \text{ROOT}(\text{GNN}(P_i)) \tag{2}$$

$$v_i^0 = f(\sigma(x_i \oplus \overline{x_i})) \tag{3}$$

Where is σ nonlinear activation function such as $ReLU$, f is the mapping of the fully connected layer, and GNN represents a multi-layer graph neural network encoder. The main structure of the GNN module adopts graph sampling and aggregation. At the news content level, it directly extracts the original news features to extract content structure features.

3.4. Multi-source feature extraction

3.4.1. Structural Feature Extraction

The structural feature extraction module is shown in Figure 3. The core idea of this module is to capture high-order structural features in the news dissemination process based on the hypergraph attention mechanism [28]. The module mainly consists of two parts: a hypergraph for constructing news relationships and a structure-aware hypergraph attention mechanism.

1. Constructing a hypergraph of news relationships: This paper starts from three core dimensions of fake news dissemination: social collaboration, temporal suddenness, and semantic consistency. It designs three types of hyperedge to construct a detection hypergraph to comprehensively model the high-order relationships in the news dissemination process.

User ID hyperedge (U^{ID}). The spread of fake news is often accompanied by the collaborative behavior of abnormal user groups (such as intensive forwarding by robot accounts and cross-community spread) [36]. If multiple news items are forwarded by the same user or user group, they are connected through the user ID superedge to capture potential collaborative patterns in the spread chain. For example, when the same group of users forwards multiple news items in a short period of time, it may indicate that they are using a fake information spread strategy (such as coordinated inauthentic behavior).

Timestamp hyperedge (U^{Time}). The number of reposts of fake news often surges rapidly and then decays rapidly in a very short period of time [29], showing obvious sudden characteristics. Inspired by this observation, we connect news nodes published within the same time window (e.g., one hour) via timestamp hyperedges. This captures abnormal propagation rates and provides a strong signal for detecting bursty fake-news patterns. For example, the propagation of real news usually shows a natural decay trend over time, while fake news may experience an abnormal surge in reposts within a specific time window (such as the proportion of reposts exceeding 80% within 10 minutes).

Shared Entity Hyperedge (U^{Entity}) . Fake news often tampers with or fabricates key entities (such as organization names and locations) to create misleading content [30]. Inspired by this, connecting multiple news items that share at least one named entity through shared entity hyperedge can effectively identify content-level semantic anomalies, such as when a fictitious organization name appears repeatedly across unrelated news articles, potentially indicating systemic fraud. To construct the proposed hypergraph, a combination of association matrices based on three construction methods ((U^{ID}), (U^{Time}) *and* (U^{Entity})) is used, resulting in $U^{\mathrm{ID}} \oplus U^{\mathrm{Time}} \oplus U^{\mathrm{Entity}}$, where $\oplus$ represents a concatenation operation. This design enhances the model's ability to characterize fake news propagation patterns by covering multi-dimensional high-order relationships, thereby improving detection accuracy and robustness.

2. Structure-Aware Hypergraph Attention: At the structural feature level, an attention-based hypergraph message passing strategy is proposed to enhance the expressive power of structural information. For a hypergraph $H = (V, E)$, where V is the set of nodes and E is the set of hyperedge, the hyperedge attention coefficient is defined.

$$\alpha_{ij} = \frac{\exp(\mathrm{LeakyReLU}(a^T[Wh_i \parallel Wh_j]))}{\sum_{k\in\mathcal{N}(i)}\exp(\mathrm{LeakyReLU}(a^T[Wh_i \parallel Wh_k]))} \tag{4}$$

Where h_i is the feature representation of the node, $N(i)$ is the set of neighboring nodes connected to the node, and W is a trainable weight matrix. After multiple propagation layers, the node representation is continuously updated, and its specific calculation method is as follows:

$$h_i^{(l+1)} = \sigma\left(\sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(l)} W^{(l)} h_j^{(l)}\right) \tag{5}$$

Where ${h_i}^{l+1}$ represents the node features of the first layer, $W^{(l)}$ represents the weight matrix of the first layer, represents the attention weights of the first layer, and is the activation function. After propagation through L layers, the final structural feature h is calculated as follows:

$$h_s = W_{\mathrm{fc}} h_i^{(L)} + b_{\mathrm{fc}} \tag{6}$$

Where W_{fc} andb_{fc} are the weights and biases of the fully connected layer. The hyperedge attention coefficient dynamically measures each hyperedge's contribution to node representation during message passing.

DOI: https://doi.org/10.70715/jitcai.2026.v3.i2.055

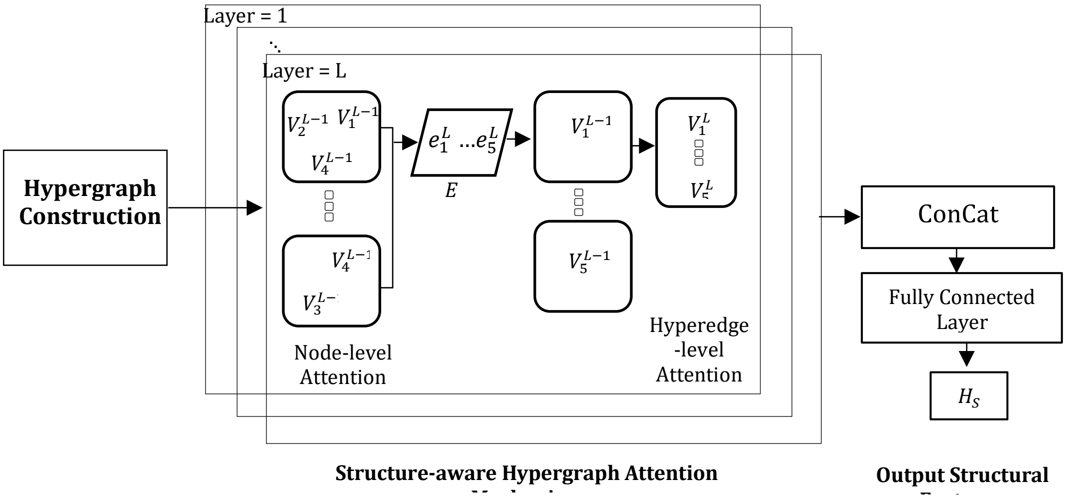


Figure 3. Structural feature extraction structure diagram. [28]

The hyperedge attention coefficients are dynamically learned from the interaction between node features and hyperedge connectivity. This mechanism is optimized end-to-end during training, allowing the model to capture higher-order structural patterns in news dissemination [38]. Specifically, the feature representation of a node and the relationship with its neighboring nodes jointly determine the attention intensity between the node and the hyperedge, which in turn affects the node representation in the information update process. Through this mechanism, the hyperedge attention coefficient not only improves the expressive ability of dissemination structural information, but also improves the performance, enabling the model to more accurately characterize complex dissemination patterns and ultimately improve the overall detection performance.

3.4.2. Temporal Feature Extraction

The temporal feature extraction module is shown in Figure 4. The core idea of this module is to capture the temporal features of news dissemination at a fine-grained level by combining the time decay attention mechanism with multi-scale temporal modeling [31]. The module mainly includes temporal position encoding, time decay attention mechanism and multi-scale time window hierarchical encoding.

1. Temporal position coding: In order to capture the temporal position information of nodes in the propagation sequence, this study has designed a temporal-aware position coding mechanism [31].

$$PE(t,i) = \begin{cases} \sin\left(\dfrac{t}{10000^{\frac{2i}{d_{\text{model}}}}}\right) & \text{if } i \text{ even} \\ \cos\left(\dfrac{t}{10000^{\frac{2i}{d_{\text{model}}}}}\right) & \text{if } i \text{ odd} \end{cases} \tag{7}$$

Where represents the node's timestamp, represents the dimension index of the location encoding, and represents the model's hidden dimension. By combining sine and cosine functions, we ensure that each timestamp has a unique representation across different dimensions and maintains a smooth, periodic change in the encoded values of adjacent time points. The final representation of the node is as follows:

$$h_i^t = h_i + PE(t,i) \tag{8}$$

This design ensures that the model can perceive the absolute temporal position of the nodes, thereby enhancing its ability to perceive temporal features and enabling it to capture temporal changes in the news dissemination process more accurately.

2. Time decay attention mechanism: Considering the timeliness of information dissemination, this paper proposes a time decay-based attention calculation method [32].

$$\alpha_{ij}^t = \frac{\exp(-\lambda \Delta t_{ij})\, f(h_i^t, h_j^t)}{\sum_{k \in \mathcal{N}(i)} \exp(-\lambda \Delta t_{ik})\, f(h_i^t, h_k^t)} \tag{9}$$

Where Δt_{ij} represents the time interval between nodes, Δt_{ik} represents the time interval i between a node k and its other neighboring nodes, λ is a learnable decay coefficient, and f is the attention scoring function.

$$f(h_i^t, h_j^t) = \text{LeakyReLU}(h_i^t \| h_j^t) \tag{10}$$

where the time decay term $\exp(-\lambda\Delta t_{ij})$ enables the model to adaptively adjust the influence weights between nodes at different time distances.

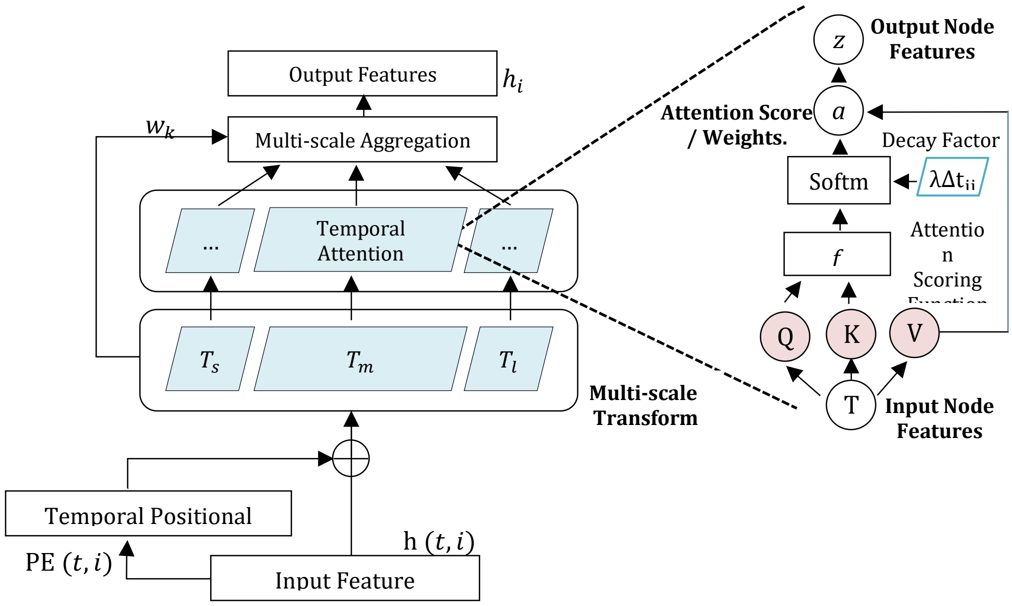


Figure 4. Temporal feature extraction structure diagram [31]

3. Hierarchical coding with multi-scale time windows: To comprehensively capture propagation patterns at different time scales, a hierarchical coding structure with multi-scale time windows was designed.

$$T = \{T_s, T_m, T_l\} \tag{11}$$

Where T_s, T_m, T_l represent the short-term, medium-term, and long-term time windows, respectively, corresponding to the three scales obtained by multiplying the time window of the GNN encoding output by 2, keeping it unchanged, and dividing it by 2. For each time scale, the corresponding propagation representation is calculated.

$$h^{(k)} = \text{AGG}_k\left(\{h_j^{(k)} \mid j \in \mathcal{N}_k(i)\}\right) \tag{12}$$

Where $N_k(i)$ represents the set of neighboring nodes for node i under the given time scale, and AGG is the aggregation function used to integrate the features of neighboring nodes. Through a residual structure, the features extracted by the attention network are adaptively weighted and fused to obtain the final temporal feature h, and the fusion process is shown in formula (13):

$$h_t = \sum_{k\in\{s,m,l\}} W_k\, h^{(k)} \tag{13}$$

The weight W_k is generated by the gating mechanism and is expressed as follows:

$$W_k = \frac{\exp(g_k)}{\sum_{k' \in s,m,l} \exp\left(L_{g_{k'}}\right)} \tag{14}$$

$$g_k = W_g h_k^{(k)} + b_g \tag{15}$$

Where $L_{g_{k'}}$ is the eigenvalue used for normalizing weights at time scale k, used to select the feature contribution at different time scales, and W_g is a trainable matrix.

3.4.3. Content Feature Extraction

At the content feature extraction level, in order to effectively capture the semantic information and contextual features of news text, a semantically enhanced self-attention module was designed, as shown in Figure 5. For a given initial feature matrix $X \in R^{nxd}$, of news nodes, where n represents the number of news nodes and d is the feature dimension of each node, the calculation process of a single attention head is first defined [32].

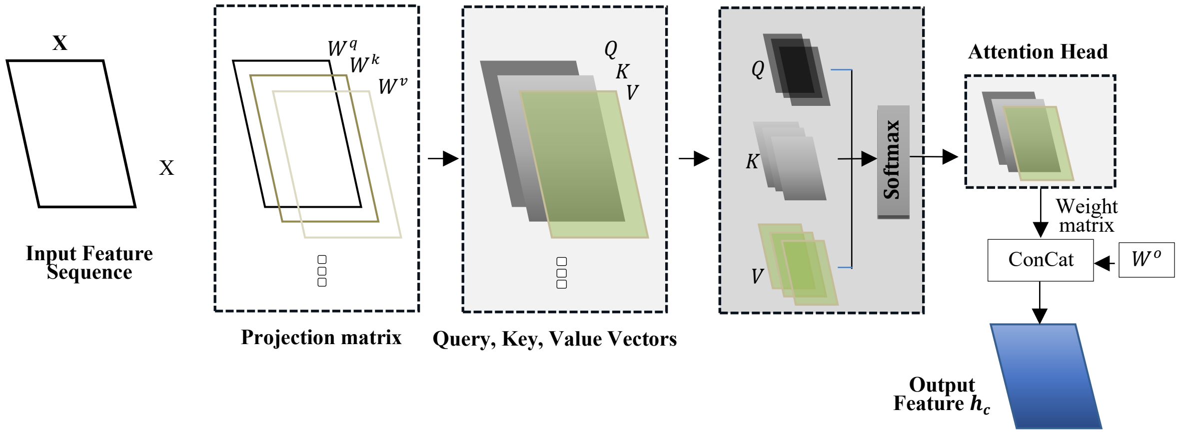


Figure 5. Structure diagram of news text feature extraction

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{16}$$

Where $Q = XW_i^q$, $K = XW_i^k$, $V = XW_i^v$, $W_i^v \in R^{d\,x\,d_k}$, $W_i^k \in R^{d\,x\,d_k}$, $W_i^q \in R^{dxd_k}$, $W_i^v \in R^{d\,x\,d_k}$, represents the learnable projection matrix of the i^{th} head, and d represents the scaling factor. The output of a single attention head can be represented as [33].

$$\text{head}_i = \text{Attention}(XW_i^q, XW_i^k, XW_i^v) \tag{17}$$

In this way, each attention head can learn different attention patterns, thereby enhancing the model's perception of various semantic information in news texts. In order to capture different semantic perspectives of news texts, this paper introduces a multi-head self-attention mechanism, which improves the model's ability to comprehensively express text features by concatenating the outputs of multiple attention heads and performing linear mapping[33].

$$h_c = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^o \tag{18}$$

Where $W^o \in R^{h\,x\,d_v}$,represents the learnable weight matrix, h represents the number of attention heads, each attention head is calculated independently according to Equation (17), and finally semantic features are integrated through Equation (18). This design enables the model to adaptively focus on key semantic information in news content, thereby improving the ability to capture text semantic features.

3.4.4. Multi-source feature fusion

To efficiently fuse extracted structural, temporal, and content features, this paper introduces the idea of a gating mechanism and designs an adaptive multi-source feature fusion mechanism. This mechanism can dynamically adjust the weight contribution of each feature dimension to achieve more accurate feature fusion. The specific structure of the adaptive multi-source feature fusion module is shown in Figure 6.

This module inputs multi-source features into an activation function and efficiently fuses them with the input multi-source features by calculating an adaptive gating vector g_k. The fusion process is shown in equations (19) and (20).

$$g_i = \sigma(W_g h_i + b_g) \tag{19}$$

$$h_i = \sum\nolimits_{k\in\{s,t,c\}} g_k \odot h_k \tag{20}$$

Where $h_s, h_t, and\ h_c$ represent structural, temporal, and content features, respectively, g_i is the adaptive gating vector, $\sum_{i=1}^{n} g_i = 1$, $\odot$ indicates point-wise multiplication, and h_r represents the output fused features.

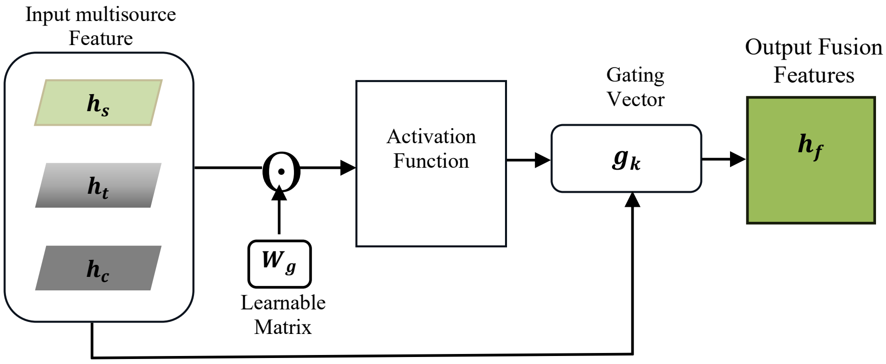


Figure 6. diagram of multi-source feature fusion

3.4.5. News Classification

For the news classification component, the fused features output by the multi-source feature extraction component serve as input. A fully connected layer with a trainable weight matrix $W_r \in R^{d \times 2}$ and bias term b, together with the *Sigmoid* activation function, is employed to classify each news article into one of two labels: "true" or "fake". Accordingly, negative log-likelihood loss is utilized to optimize the loss between the predicted label $\hat{y}_i$ and the true label y_i for each news item i , where y_t is the label set of the dataset. The detailed computation is presented in Equations (21) and (22) [28].

$$L = -\sum_{y_i \in Y_t} (y_i \ln(\hat{y}_i) + (1 - y_i)\ln(1 - \hat{y}_i)) \tag{21}$$

$$\hat{y}_i = \text{sigmoid}(f(W_r h_i^t + b)) \tag{22}$$

4. Experiments

4.1. Dataset and Evaluation Metrics

The UPFD dataset (which includes the Politifact dataset and the Twitter16 dataset) was constructed by Dou et al. [17] and is widely used for benchmarking.

The UPFD dataset contains two tree-structured graph sets designed to evaluate binary graph classification, graph anomaly detection, and fake news detection tasks. The dataset is stored as PyTorch-Geometric dataset objects and primarily consists of fake news propagation (retweeting) networks on the Twitter platform. Each graph in these networks is a hierarchical tree structure, where the root node represents the news item, and the leaf nodes represent Twitter users who retweeted it. If a user node retweeted the news item, there is an edge pointing to the news node. Statistics for the dataset are shown in Table 2.

As shown in Table 2, the Politifact dataset is considerably smaller than Twitter16, which has implications for overfitting discussed in Section 4.3. To ensure a fair and reproducible comparison, this study follows the evaluation protocol of the original UPFD benchmark [28] , which employs a 20%/10%/70% training/validation/test split. This unconventional ratio is intentional: the larger held-out test set is designed to provide a more rigorous assessment of generalisation performance across a wider range of unseen examples, which is particularly important given the class imbalance and limited size of the Politifact dataset.

4.2. Evaluation Indicators

For binary classification tasks in fake news detection, accuracy is usually used as the main evaluation metric. However, when training on an imbalanced dataset, the model may be biased towards the majority class, leading to a decrease in reliability. Therefore, in order to more comprehensively evaluate the model performance, in addition to accuracy, the F1 score was introduced as a supplementary evaluation metric in the experiment. The F1 score can comprehensively consider the model's precision and recall. The definitions of the above evaluation metrics are as follows [34].

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{23}$$

$$\text{F1-score} = \frac{2\,\text{Precision Recall}}{\text{Precision+Recall}} \tag{24}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{25}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{26}$$

Where True Positive (TP) is the number of positive samples classified as positive, True Negative (TN) is the number of negative samples classified as negative, False Positive (FP) is the number of negative samples classified as positive, and False Negative (FN) is the number of positive samples classified as negative.

4.3. Experimental setup

The hardware configuration for the experiments consisted of a high-performance workstation featuring an Intel® Core™ i9-13900K processor (5.40 GHz), supported by 64GB of DDR5 high-speed memory. To handle the computational demands of the hypergraph and temporal attention modules, an NVIDIA GeForce RTX 4090 GPU with 24GB of VRAM was utilized. The software environment was established on Windows 11 Professional, using Python 3.10.11 and PyTorch 2.2.0. Accelerated tensor computations were managed via CUDA 12.2 and cuDNN 8.9.

Table 2: Datasets Information

Dataset	# Graphs	# Fake News	# Real News	# Nodes	# Edges
Politifact	314	157	157	41,054	50,740
Twitter16	818	~400	500	120,000	130000

Note: "#" represents quantity.

4.4. Comparative Experiment

This paper compares the proposed FNDMFF method with existing representative fake news detection methods. The selected baseline methods include propagation tree-based methods (such as GNN-CL [14], Bi-GCN [15], GCNFN [16], RTRUST [35], UPFD [17] and its variants, TGNF [27], GTN [36]), heterogeneous graph-based methods (HGAT [18]), and hypergraph-based methods (HGNN [37]). Except for the Robust Trust evaluation architecture for fake news detection (RTRUST) method, the other baseline methods are all based on the reproduction results provided in reference [28] . The experimental results are shown in Table 3 and figure 7. This framework outperforms all baseline methods on both the Politifact and Twitter16 datasets, achieving the best detection performance. On the Politifact dataset, the FNDMFF method achieved an average accuracy of 93.75% and an F1 score of 93.27%; on the Twitter16 dataset, the average accuracy and F1 score reached 98.01% and 97.97%, respectively, further improving the detection performance compared to the current state-of-the-art baseline RTRUST.

Compared to the SOTA method RTRUST [35], FNDMFF outperforms all baselines on both datasets, achieving 93.75% accuracy and 93.27% F1 on Politifact, and 98.01% accuracy and 97.97% F1 on Twitter16.

The performance gains stem primarily from its ability to model high-order propagation relationships via an enhanced hypergraph neural network—something traditional propagation-tree methods (such as RTRUST) cannot capture. Unlike heterogeneous-graph and standard hypergraph approaches that rely on limited or single-type features, FNDMFF integrates structural, temporal, and content features through adaptive multi-source fusion, providing a more comprehensive representation of fake news dissemination patterns.

Table 3. Comparative Experimental Results (%)

Ref (Model)	Politifact (Accuracy)		Twitter16 (Accuracy)	
	Accuracy	F1-score	Accuracy	F1-score
[14]	65.79 ±8.96	65.02 ± 9.46	94.98 ±0.80	94.94 ±0.80
[15]	74.16 ±3.57	74.16 ±3.57	88.04 ±0.48	87.95 ±0.49
[17]	80.27 ±4.35	80.16 ±4.41	95.55 ±0.63	95.51 ±0.64
[38]	80.40 ±4.22	80.13 ±4.65	96.38 ±0.48	96.36 ±0.48
[27]	74.28 ±1.74	74.09 ±1.81	85.07 ±0.08	85.07 ±0.08
[36]	81.67 ±4.16	81.53 ±4.35	92.41 ±0.98	92.38 ±0.98

[18]	81.53 ±1.16	80.47 ±1.75	—	—
[16]	80.63 ±4.23	80.31 ±4.57	95.37 ±0.21	95.33 ±0.21
[37]	79.96 ±4.89	79.28 ±5.16	93.38 ±0.49	93.38 ±0.49
[35]	90.11	89.86	97.46	97.41
Our (Proposed)	93.75 ±1.56	93.27 ±1.68	98.01 ±0.08	97.97 ±0.08

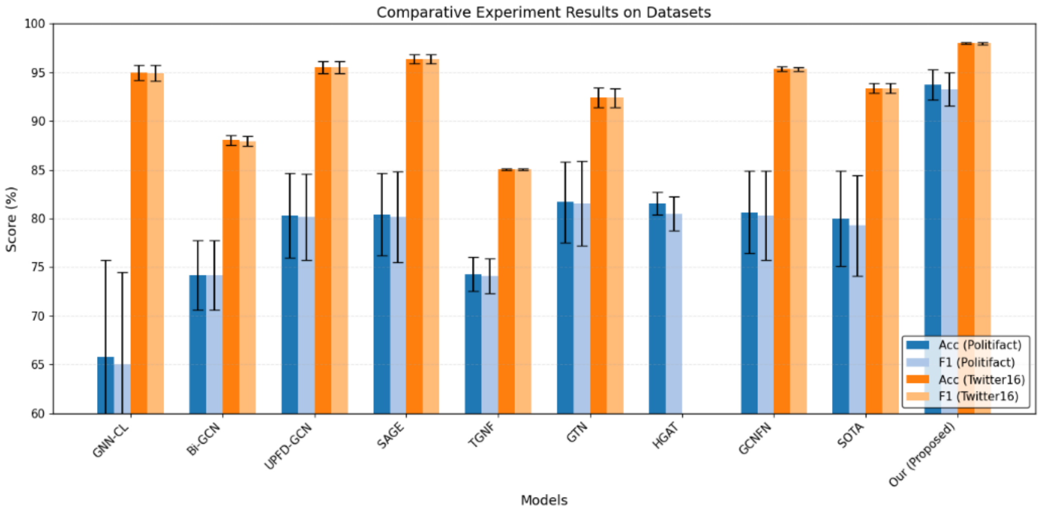


Figure 7. Performance comparison of FNDMFF against state-of-the-art baselines on datasets

The mechanism dynamically adjusts the weights of each feature source, achieving precise fusion of structural, temporal, and content features, thereby providing a more accurate semantic representation for fake news detection.

The size of the dataset and the number of feature types have a significant impact on the performance. Small datasets such as Politifact are more likely to cause overfitting of the detection model, while large datasets such as Twitter16 can effectively mitigate the risk of overfitting and improve the generalization ability of the detection model [50]. The FNDMFF method effectively reduces the risk of overfitting by using a multi-source feature fusion strategy to comprehensively consider the structural, temporal, and content features of news, and can achieve good detection performance even on small datasets. However, methods such as UPFD and HGNN only use the propagation structure features of news for fake news detection tasks, and lack comprehensive modeling of the multi-dimensional characteristics of news, which also leads to their poor performance on small datasets.

4.5. Ablation & Hyperparameter Analysis

To verify the effectiveness of the key components in the proposed method, various model variants were designed for ablation experiments, and the results are shown in Table 4. Here, S represents structural features, T represents temporal features, C represents content features, and F represents the multi-source feature fusion module.

The base model (removing S, T, C features and the F module) was tested in Politifact and Twitter16 The accuracy and F1 score were 84.62%/84.56% and 97.23%/97.22%, respectively. After adding structural features S to the base model, the accuracy and F1 score of the model on the Politifact and Twitter16 datasets improved to 92.19%/91.60% and 97.53%/97.53%, respectively.

After adding the temporal feature T to the base model, the accuracy and F1 score of the model on the Politifact and Twitter16 datasets improved to 89.94%/89.93% and 97.72%/97.71%, respectively. This is because the temporal feature enables the model to more accurately capture the dynamic changes in news dissemination, especially the rapid spread in the early stages of news dissemination and the decay effect in the later stages, which is particularly important for identifying abnormal dissemination patterns of fake news.

After adding content feature C to the base model, the accuracy and F1 score of the model on the Politifact and Twitter16 datasets improved to 89.42%/89.42% and 97.62%/97.61%, respectively. This is because the text content features of news can more intuitively reflect its authenticity and credibility, such as exaggerated headlines and descriptions that attempt to attract users' attention.

To verify the effectiveness of the multi-source feature fusion module (F), this paper designs a comparative model that does not use a gated fusion mechanism. This model directly concatenates three features: structure (S), time series (T), and content (C), and uses a fully connected layer for dimensionality reduction to obtain the final news representation. The comparative model achieves performance of 92.90%/92.86% and 97.89%/97.89% on the Politifact and Twitter16 datasets, respectively. Although the detection performance is improved compared to models using S, T, and C features alone, it is still lower than the complete model using a gated fusion mechanism (93.75%/93.27% and 98.01%/97.97%). This is because directly concatenating the three features and using a fully connected layer for dimensionality reduction cannot effectively capture the highly nonlinear relationships between different feature categories. The multi-source feature fusion module (F), by dynamically adjusting the weights of different feature categories, can more effectively capture the highly nonlinear relationships between structure, time series, and content feature categories, thereby improving the model's performance in detecting fake news.

To verify the impact of the core hyperparameter hidden size on the performance of the FNDMFF model, this paper conducted experiments on the Politifact and Twitter16 datasets, testing the accuracy on the validation set when the hidden size was set to 32, 64, 128, 256, and 512. The experimental results are shown in Figure 8. The model performed best when the hidden size was set to 128. A smaller hidden size may lead to underfitting, while an excessively large hidden size can easily cause overfitting. Therefore, this paper ultimately chose a hidden size of 128 as the optimal parameter configuration.

Table 4. Ablation test results

Feature type				**Politifact**		**Twitter16**	
S	T	C	F	(Acc)	(F1)	(Acc)	(F1)
✗	✓	✗	✓	84.62	84.56	97.23	97.22
✓	✗	✗	✗	92.19	91.60	97.53	97.53
✗	✓	✗	✗	89.94	89.93	97.72	97.71
✗	✓	✓	✗	89.42	89.42	97.62	97.61
✓	✓	✓	✗	92.90	92.86	97.89	97.89
✓	✓	✓	✓	**93.75**	**93.27**	**98.01**	**97.97**

4.6. Test set instance analysis

To verify the impact of the core hyperparameter training epochs on the performance of the FNDMFF model, this paper gradually increases the number of training epochs on the Politifact and Twitter16datasets, observing the changes in loss and accuracy on the training and validation sets. The experimental results are shown in Figure 8. On the Twitter16 dataset, the loss and accuracy gradually stabilize as the epoch increases to 100; however, on the Politifact dataset, due to the smaller data size, the model reaches a fitted state earlier, and training stops at epoch 60.

To illustrate the practical effectiveness of the proposed FNDMFF framework, we conduct a qualitative case study on representative instances from the test sets of Politifact and Twitter16. Table 5 presents four characteristic examples, categorized by detection difficulty (FNDMFF-sensitive vs. FNDMFF-insensitive). For each case, we report the tweet count (total retweets in the propagation tree), posting-time characteristics, and the original fake news headline. We then analyze why certain cases are easily detected while others remain challenging, highlighting the complementary roles of structural, temporal, and content features. To verify the impact of the core hyperparameter, namely the number of training epochs, on the performance of the FNDMFF model, this paper gradually increases the number of training epochs on datasets and observes the changes in loss and accuracy on both the training and validation sets. The experimental results are presented in Figure 9. On the Twitter16 dataset, the loss decreases and the accuracy increases steadily as the number of epochs grows, eventually stabilizing around epoch 100. In contrast, due to the smaller size of the Politifact dataset, the model reaches convergence earlier, and training stops at approximately epoch 60.

Analysis of FNDMFF-Sensitive Cases (easily detected). Cases 1 and 2 exemplify high-transmissibility fake news with distinctive multi-source signals. Both exhibit explosive structural patterns (dense clusters of retweets around a few opinion-leader accounts), abnormal temporal dynamics (sharp early peak followed by rapid decay), and sensational content markers (capitalized emotional language, exaggerated claims). The enhanced hypergraph module captures the high-order collaborative forwarding groups, the multi-scale temporal module flags the anomalous burst rate, and the multi-head self-attention mechanism highlights inflammatory keywords. Consequently, FNDMFF assigns high fake-news probabilities (>0.95) to these instances.

Analysis of FNDMFF-Insensitive Cases (more challenging). Cases 3 and 4 represent low-visibility fakes that spread subtly. Their propagation trees are sparse (few retweets, no concentrated clusters), temporal patterns are unremarkable (no explosive peaks), and headlines appear neutral or bureaucratic. In such cases, structural and temporal signals are weak; detection relies more heavily on fine-grained semantic anomalies extracted by the multi-head attention layer (e.g., subtle factual distortions or misleading framing). Although FNDMFF still correctly classifies these instances, the confidence scores are lower (0.68–0.82), indicating room for further refinement in low-propagation scenarios.

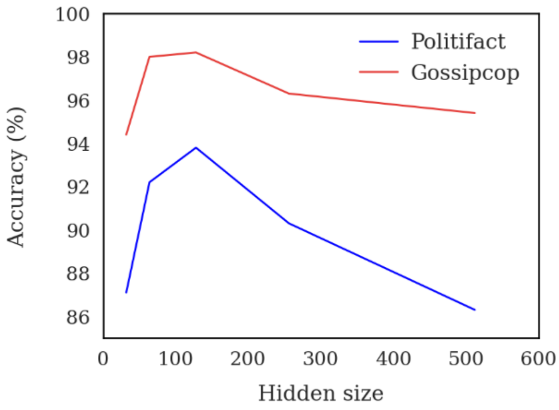
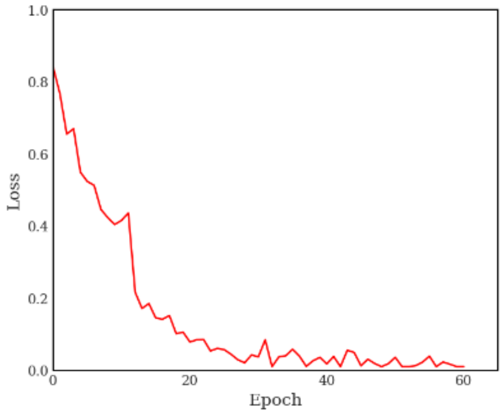
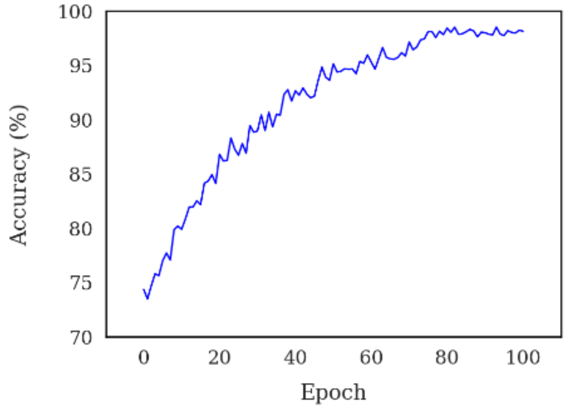


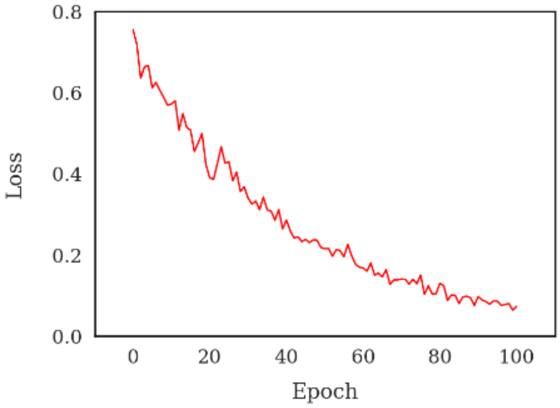
Figure 8. Effect of Hidden Size on Experimental Results



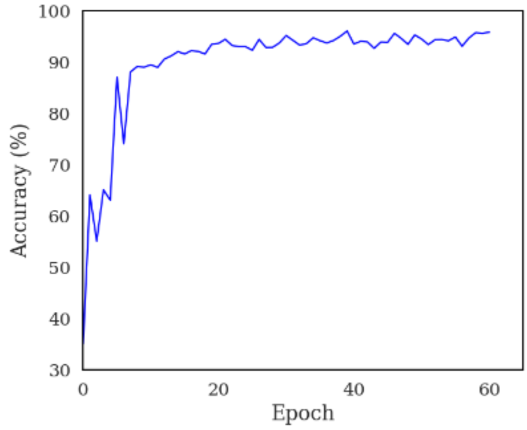
(a) Model Loss on the Politifact Dataset.



(b) Model Accuracy on the Twitter16 Dataset.



(c) Model Loss on the Twitter16 Dataset.



(d) Model Accuracy on the Politifact Dataset.

Figure 9. shows the training process of the algorithm on two datasets.

Table 5. Case Analysis of Fake News Detection

Category	Case	Tweet Count (Spread)	Posting Time Characteristics	Fake News
FNDMFF Sensitive	Case 1	12,450	Explosive surge within first 2 hours (80% of retweets)	"Pope Francis Shocks World by Endorsing Donald Trump for President"
FNDMFF Sensitive	Case 2	8,720	Rapid burst in first 30 minutes followed by quick decay	"Hillary Clinton Sold Weapons to ISIS, Leaked Emails Reveal"
FNDMFF insensitive	Case 3	18	Slow, sporadic diffusion over 5+ days	"Local Congressman Introduces Minor Amendment to State Tax Code"
FNDMFF insensitive	Case 4	9	Low-volume, steady propagation across weeks	"New Study Shows Slight Increase in Seasonal Flu Cases Nationwide"

4.7. Computational Complexity

4.7.1. Complexity Analysis of Structural Feature Extraction Module

1. *Time Complexity Analysis*: The time complexity of the structural feature extraction module mainly stems from the hypergraph construction, hypergraph attention mechanism, and hypergraph aggregation of the hypergraph module. The core of hypergraph construction lies in constructing the hypergraph adjacency matrix H, where H is an $N \times E$ sparse matrix, N is the number of nodes, and E is the number of edges of hypergraph. Calculating H requires traversing all hyperedge and filling the matrix, resulting in a time complexity of $O(N+E)$. The hypergraph attention mechanism involves calculating the attention coefficient of each hyperedge and performing sparse matrix multiplication, with a time complexity of $O(Nd+Ed)$, where d is the hidden layer dimension. During hypergraph aggregation, each GAT layer requires one hypergraph attention calculation, resulting in an overall time complexity of $O(L(Nd+Ed))$, where L is the number of GAT layers. Therefore, the overall time complexity of the structural feature extraction module is $O(L(Nd + Ed))$.

2. *Space Complexity Analysis*: The adjacency matrix H of the hypergraph has a dimension of N×E, and is usually stored sparsely, with a space complexity of approximately $O(N+E)$. In addition, the node feature matrix $X \in RNxd$ needs to be stored additionally, with a space complexity of $O(Nd)$. At the same time, the hypergraph attention layer needs to store the attention coefficients and related parameters. Therefore, the overall space complexity of the structural feature extraction module is $O(Nd+Ed)$.

4.7.2. Complexity Analysis of the Temporal Feature Extraction Module

1. *Time Complexity Analysis*: The time complexity of the temporal feature extraction module mainly comes from the linear changes in multi-scale temporal modeling, self-attention calculation, attention weighted aggregation, and multi-head attention merging. Among these, the complexity of the attention calculation module is the primary consideration. The time complexity of this part is $O(HNT^2d)$, where T is the length of the time window, H is the number of time attention heads, N is the number of nodes, and d is the dimension of the output of each time attention head.

2. *Space complexity analysis:* The space complexity of the temporal feature extraction module mainly comes from the storage of a large-scale temporal attention matrix, and its space complexity is $O(HNT^2)$.

4.7.3. Complexity Analysis of Content Feature Extraction Module

1. Time Complexity Analysis: The time complexity of the content feature extraction module mainly comes from the multi-head self-attention mechanism. In each layer, its time complexity is $O(N^2d)$, where N is the length of the input sequence and d is the dimension of the hidden layer. Since the multi-head self-attention mechanism uses parallel computation, the overall time complexity is $O(HN^2d/H) = O(N^2d)$, where H is the number of attention heads.
2. Space complexity analysis: The space complexity of the content feature extraction module is $O(h.n^2+nd+d^2)$, which mainly comes from the storage of the input sequence and the parameter overhead of the attention weight matrix.

DOI: https://doi.org/10.70715/jitcai.2026.v3.i2.055

4.7.4. Complexity Comparison Analysis

By analyzing the pseudocode of the existing SOTA method RTRUST. A systematic analysis reveals that the computational complexity is mainly distributed across graph convolution computation $O(L|E|D_h)$, user similarity computation $O(|U|^2D_h)$, and user embedding computation $O(|U|D_iD_h)$. Its overall complexity can be formally defined as $O(L|E|D_h + |U|^2Dh + |U|D_iD_h)$, where L is the number of graph convolution layers, $|E|$ is the number of social network edges, D_h is the hidden layer embedding dimension, $|U|$ is the total number of user nodes, and D is the dimension of the user input feature vector. In contrast, the FNDMFF method proposed in this paper, through a multi-source feature fusion mechanism, concentrates its main computational complexity in the attention module. Both methods maintain an overall complexity at the $O(n^2)$ level. The FNDMFF method achieves computational efficiency comparable to RTRUST while ensuring controllable computational complexity.

5. Conclusion

With the rapid evolution of the mobile Internet and Generative Artificial Intelligence (GenAI), the fabrication and dissemination of fake news have become increasingly prevalent and difficult to contain , frequently triggering widespread negative public sentiment. The multi-source feature fusion-enhanced fake news detection method proposed in this paper provides comprehensive multi-dimensional modeling by integrating structural, temporal, and content-based features.

This approach effectively improves the performance of news authenticity verification and holds the potential to mitigate the social costs associated with fake news, such as public misinformation, damage to corporate reputations, and policy misjudgments. Furthermore, the proposed methodology exhibits robust generalization capabilities. Its multi-source feature extraction and fusion framework can be transferred at a low cost to other domains, including social network fraud identification, cyber-attack detection, medical diagnosis, and public opinion analysis.

This paper proposes a novel multi-source feature fusion-enhanced fake news detection method that integrates three critical feature dimensions: structure, temporal dynamics, and content. The proposed approach effectively addresses the limitations of existing research, such as the underutilization of multi-level features and the difficulty of capturing high-order propagation relationships. Specifically, the structural features capture high-order propagation patterns; the temporal features track the dynamic evolution of news dissemination; and the content features extract the core semantic information within the news text.

Furthermore, the adaptive multi-source feature fusion mechanism captures highly non-linear relationships between different dimensions by dynamically adjusting the importance of each feature. Experimental results show that FNDMFF achieves 93.75% accuracy and 93.27% F1-score on Politifact, and 98.01% accuracy and 97.97% F1-score on Twitter16, outperforming the state-of-the-art RTRUST baseline by 3.63% and 3.40% on Politifact, and 0.54% and 0.55% on Twitter16, respectively. Extensive experiments on two public datasets, PolitiFact and Twitter16, demonstrate that FNDMFF significantly outperforms existing state-of-the-art baselines. Future research will focus on few-shot or zero-shot learning, incorporating prompt engineering and Large Language Models (LLMs) to further enhance generalization performance.

6. FUNDING INFORMATION

The authors state no funding is involved.

7. CONFLICT OF INTEREST STATE

The authors state no conflict of interest.

8. ETHICAL APPROVAL

This paper does not involve people or animals; no investigation has involved human subjects. Therefore, the authors did not seek approval from any institutional review board.

9. DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, [S], upon reasonable request.

10. REFERENCES

[1] F. Olan, U. Jayawickrama, and E. O. Arakpogun, "Fake news on social media: The impact on society," *Information Systems Frontiers,* vol. 26, no. 2, pp. 443-458, 2024, doi: 10.1007/s10796-022-10242-z.

[2] H. Sun, J. Tang, and S. Dang, "Privacy and distribution preserving generative adversarial networks with sample balancing," *Expert Systems with Applications,* vol. 258, p. 125181, 2024, doi: 10.1016/j.eswa.2024.125181.

[3] E. Broda and J. Strömbäck, "Misinformation, disinformation, and fake news: lessons from an interdisciplinary, systematic literature review," *Annals of the International Communication Association,* vol. 48, 03/06 2024, doi: 10.1080/23808985.2024.2323736.

[4] J. Alghamdi, Y. Lin, and S. Luo, "Towards COVID-19 fake news detection using transformer-based models," *Knowledge-Based Systems,* vol. 274, p. 110642, 2023, doi: 10.1016/j.knosys.2023.110642.

[5] S. Raza and C. Ding, "Fake news detection based on news content and social contexts: A transformer-based approach," *International Journal of Data Science and Analytics,* vol. 13, no. 4, pp. 335-362, 2022, doi: 10.1007/s41060-021-00302-z.

[6] N. Ruchansky, S. Seo, and Y. Liu, "CSI: A hybrid deep model for fake news detection," in *2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 797-806, doi: 10.1145/3132847.3132877.

[7] K. Shu, L. Cui, and S. Wang, "dEFEND: Explainable fake news detection," in *The 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 395-405, doi: 10.1145/3292500.3330935.

[8] G. Hu, G. Lu, and Y. Zhao, "Bidirectional hierarchical attention networks based on document-level context for emotion cause extraction," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, 2021, pp. 558-568, doi: 10.18653/v1/2021.findings-emnlp.51.

[9] J. Vatter, R. Mayer, and H. A. Jacobsen, "The evolution of distributed systems for graph neural networks and their origin in graph processing and deep learning: A survey," *ACM Computing Surveys,* vol. 56, no. 1, p. 6, 2023, doi: 10.1145/3597428.

[10] Z. Wu, S. Pan, and F. Chen, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 32, no. 1, pp. 4-24, 2021, doi: 10.1109/TNNLS.2020.2978386.

[11] U. A. Bhatti, H. Tang, and G. Wu, "Deep learning with graph convolutional networks: An overview and latest applications in computational intelligence," *International Journal of Intelligent Systems,* vol. 2023, no. 1, p. 8342104, 2023, doi: 10.1155/2023/8342104.

[12] Z. Li, Y. Zhao, and Y. Zhang, "Multi-relational graph attention networks for knowledge graph completion," *Knowledge-Based Systems,* vol. 251, p. 109262, 2022, doi: 10.1016/j.knosys.2022.109262.

[13] H. Wu, Z. Zhang, and S. Shi, "Phrase dependency relational graph attention network for aspect-based sentiment analysis," *Knowledge-Based Systems,* vol. 236, p. 107736, 2022, doi: 10.1016/j.knosys.2021.107736.

[14] 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.

[15] T. Bian, X. Xiao, and T. Xu, "Rumor detection on social media with bi-directional graph convolutional networks," in *The 34th AAAI Conference on Artificial Intelligence*, 2020, pp. 549-556, doi: 10.1609/aaai.v34i01.5393.

[16] F. Monti, F. Frasca, and D. Eynard, "Fake news detection on social media using geometric deep learning," *arXiv preprint arXiv:1902.06673,* 2019.

[17] Y. Dou, K. Shu, and C. Xia, "User preference-aware fake news detection," in *The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 2051-2055, doi: 10.1145/3404835.3462990.

[18] Y. Ren and J. Zhang, "Fake news detection on news-oriented heterogeneous information networks through hierarchical graph attention," in *2021 International Joint Conference on Neural Networks (IJCNN)*, 2021, pp. 1-8, doi: 10.1109/IJCNN52387.2021.9534362.

[19] S. Bai, F. Zhang, and P. H. S. Torr, "Hypergraph convolution and hypergraph attention," *Pattern Recognition,* vol. 110, p. 107637, 2021, doi: 10.1016/j.patcog.2020.107637.

[20] G. Lee, F. Bu, T. Eliassi-Rad, and K. Shin, "A Survey on Hypergraph Mining: Patterns, Tools, and Generators," *ACM Comput. Surv.,* vol. 57, no. 8, mar, articleno = 203 , numpages = 36 2025, doi: 10.1145/3719002.

[21] D. Sakong, V. H. Vu, and T. T. Huynh, "Higher-order knowledge-enhanced recommendation with heterogeneous hypergraph multi-attention," *Information Sciences,* vol. 680, p. 121165, 2024, doi: 10.1016/j.ins.2024.121165.

[22] M. Wang, W. Liu, and X. Yuan, "Parallel hypergraph convolutional neural networks for image annotation," in *2022 41st Chinese Control Conference (CCC)*, 2022, pp. 6582-6587, doi: 10.23919/CCC55666.2022.9901938.

[23] Y. Jin, W. Yin, and H. Wang, "Capturing word positions does help: A multi-element hypergraph gated attention network for document classification," *Expert Systems with Applications,* vol. 251, p. 124002, 2024, doi: 10.1016/j.eswa.2024.124002.

[24] J. Ma, W. Gao, and K.-F. Wong, "Rumor Detection on Twitter with Tree-structured Recursive Neural Networks," Melbourne, Australia, July 2018: Association for Computational Linguistics, in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1980-1989, doi: 10.18653/v1/P18-1184. [Online]. Available: https://aclanthology.org/P18-1184/ https://doi.org/10.18653/v1/P18-1184

[25] S. Yun, M. Jeong, R. Kim, J. Kang, and H. Kim, *Graph Transformer Networks*. 2019.

[26] J. Hu, J. Zhang, and Z. Li, "Tracing truth: dynamic temporal networks for multi-modal fake news detection," (in eng), *PeerJ Comput Sci,* vol. 11, p. e2998, 2025, doi: 10.7717/peerj-cs.2998.

[27] C. Song, K. Shu, and B. Wu, "Temporally evolving graph neural network for fake news detection," *Information Processing & Management,* vol. 58, no. 6, p. 102712, 2021, doi: 10.1016/j.ipm.2021.102712.

[28] U. Jeong, K. Ding, and L. Cheng, "Nothing stands alone: Relational fake news detection with hypergraph neural networks," in *2022 IEEE International Conference on Big Data (Big Data)*, 2022, pp. 596-605, doi: 10.1109/BigData55660.2022.10020234.

[29] C. Zhang, A. Gupta, and X. Qin, "A computational approach for real-time detection of fake news," *Expert Systems with Applications,* vol. 221, p. 119656, 2023, doi: 10.1016/j.eswa.2023.119656.

[30] M. R. Kondamudi, S. R. Sahoo, and L. Chouhan, "A comprehensive survey of fake news in social networks: Attributes, features, and detection approaches," *Journal of King Saud University-Computer and Information Sciences,* vol. 35, no. 6, p. 101571, 2023, doi: 10.1016/j.jksuci.2023.101571.

[31] Y. Zhang, L. Ma, and S. Pal, "Multi-resolution time-series transformer for long-term forecasting," in *The 27th International Conference on Artificial Intelligence and Statistics*, 2024, pp. 4222-4230.

[32] M. Hassanin, S. Anwar, and I. Radwan, "Visual attention methods in deep learning: An in-depth survey," *Information Fusion,* vol. 108, p. 102417, 2024, doi: 10.1016/j.inffus.2024.102417.

[33] K. T. Chitty-Venkata, S. Mittal, and M. Emani, "A survey of techniques for optimizing transformer inference," *Journal of Systems Architecture,* vol. 144, p. 102990, 2023, doi: 10.1016/j.sysarc.2023.102990.

[34] H. Yang, J. Zhang, and L. Zhang, "MRAN: Multimodal relationship-aware attention network for fake news detection," *Computer Standards & Interfaces,* vol. 89, p. 103822, 2024, doi: 10.1016/j.csi.2023.103822.

[35] N. Jiang, Z. Tu, and K. Pei, "RTrust: Toward robust trust evaluation framework for fake news detection in online social networks," *World Wide Web,* vol. 27, no. 6, p. 76, 2024, doi: 10.1007/s11280-024-01317-9.

[36] H. Matsumoto, S. Yoshida, and M. Muneyasu, "Propagation-based fake news detection using graph neural networks with transformer," in *2021 IEEE 10th Global Conference on Consumer Electronics (GCCE)*, 2021, pp. 19-20, doi: 10.1109/GCCE53005.2021.9621803.

[37] Y. Feng, H. You, Z. Zhang, R. Ji, and Y. Gao, "Hypergraph neural networks," presented at the Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence , articleno = 437 , numpages = 8, 2019. [Online]. Available: https://doi.org/10.1609/aaai.v33i01.33013558.

[38] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *The 31st International Conference on Neural Information Processing Systems*, 2017, pp. 1025-1035.