

Article

A Fake News Detection Model Based on Capsule Networks and Collaborative Attention

Junkuo Cao ¹, Shiyu Zhuo ¹, Jintao Su ^{2,*} and Guolian Chen ^{1,*}¹ College of Information Science Technology, Hainan Normal University, Haikou 571158, China; cjk@hainnu.edu.cn (J.C.); qq1014221408@gmail.com (S.Z.)² The Information Network and Data Center, Hainan Normal University, Haikou 571158, China

* Correspondence: 20212083900012@hainnu.edu.cn (J.S.); cgl@hainnu.edu.cn (G.C.)

Abstract

Fake news may obscure the truth or mislead readers through subtle manipulations of textual content, such as modifying a few keywords or adjusting syntactic structures. Such local-level alterations are often difficult for detection models to capture, which undermines their overall performance. To address the limitations in processing fine-grained textual details, we propose a novel fake news detection framework—BCCU, which integrates a pre-trained language model, capsule networks, and a co-attention mechanism. Specifically, BCCU employs BERT to extract global semantic representations from news text, leverages Capsule Network to identify subtle local patterns, and synergistically fuses these two feature streams via a Co-Attention mechanism. Additionally, it incorporates User attributes as auxiliary features to further enhance detection accuracy. We evaluate the BCCU framework on three benchmark datasets—Twitter15, Twitter16, and Weibo—achieving accuracies of 0.864, 0.851, and 0.945, respectively, outperforming existing baseline models. The results demonstrate that by effectively combining global and local textual features and integrating user profile information, BCCU can robustly detect fake news even when relying solely on the unimodal text modality.

Keywords: BERT; fake news; capsule network; co-attention mechanism

Academic Editor: Luis Javier Garcia Villalba

Received: 13 October 2025

Revised: 5 November 2025

Accepted: 12 November 2025

Published: 17 November 2025

Citation: Cao, J.; Zhuo, S.; Su, J.; Chen, G. A Fake News Detection Model Based on Capsule Networks and Collaborative Attention. *Appl. Sci.* **2025**, *15*, 12190. <https://doi.org/10.3390/app152212190>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

According to the 55th Statistical Report on China's Internet Development issued by the China Internet Network Information Center (CNNIC), online news has undeniably become a key channel for news dissemination and publication [1,2]. However, the virtual and interactive nature of cyberspace has led to significant variability in the quality of online news. Some content distorts facts or misleads public opinion, thereby negatively impacting social order, public cognition, and social stability [3]. Consequently, automated detection techniques have become a critical research direction for purifying cyberspace and curbing the spread of fake news.

In recent years, research in fake news detection has largely focused on incorporating diverse external information, such as social network structures, propagation paths, or image features, to support model design and veracity assessment [4]. Nevertheless, text remains the primary carrier of news content and fundamentally shapes the audience's perception and emotional response. Fake news often employs subtle textual manipulations—such as keyword replacement, syntactic restructuring, or semantic blurring—to obscure

meaning and mislead models. These covert text-level modifications are difficult for conventional detection approaches to recognize, motivating the need for refined and deep-text analysis techniques.

In the context of online misinformation, user-centric metadata and behavioral signals play a crucial role in illuminating the social dynamics behind false information circulation. For example, a range of studies demonstrate that features such as follower counts, posting frequency, friend/follower ratios, and account age help distinguish credible from non-credible users, thereby offering an auxiliary dimension beyond the textual content of a news item. Moreover, behavioral patterns—such as rapid reposting, anomalous interaction bursts, and echo-chamber participation—have been shown to correlate with higher susceptibility to spreading fake news. By embedding such user attributes and activities into detection frameworks, one can compensate for the limitations of text-only models, which may struggle to detect subtly crafted misinformation. Building on these insights, we hypothesize that: The integration of user-attribute and behavior signals with textual analysis leads to a significantly improved detection of fake news compared to content-only models in social media contexts. The main contributions of this work are as follows:

1. We propose a novel detection framework that integrates textual analysis with user-centric metadata and user behavioral signals to enhance fake news detection in social media contexts.
2. We develop a multi-granular semantic reasoning mechanism based on hierarchical feature extraction that addresses subtle text-level manipulations (e.g., syntactic restructuring, semantic blurring) often overlooked by conventional models.
3. We design an architecture that fuses propagation structure, user engagement indicators, and content features through a co-attention mechanism, thereby bridging the gap between content-only and social-contextual detection paradigms.
4. We conduct comprehensive experiments on multiple real-world datasets, demonstrating the superior performance of our approach compared to content-only baselines and existing user-metadata or propagation-based methods.

The remainder of the paper is organized as follows: Section 2 reviews prior work and positions our study in the context of existing literature. Section 3 presents the motivation and detailed methodology of the proposed model. Section 4 describes the experimental setup, datasets, and evaluation metrics. Section 5 reports the results, conducts ablation and comparative analyses. Finally, Section 6 concludes the paper and discusses future research directions.

2. Related Work

Accurate detection of fake information in news texts typically involves feature extraction, text classification, sentiment analysis, and topic-model design to comprehensively uncover potential deceptive cues. In early research, machine learning played a dominant role in fake news detection. Wang et al. [5] innovatively applied classical models—including Support Vector Machines (SVM), Naive Bayes, and Decision Trees—and proposed an adaptive training strategy based on model confidence. Their approach dynamically adjusts sample weights according to classifier confidence levels, directing the model's focus to misclassified or low-confidence samples and improving classification performance. However, as this method still relies heavily on manually designed features and simple models, its applicability in large-scale and rapidly evolving social-media environments remains constrained. Similarly, Feng et al. [6] focused on improving the SVM model by introducing a Gaussian (RBF) kernel, optimizing sample space distribution, and enhancing intra-class cohesion; while this enhanced separability on handcrafted features, it nevertheless shared the same bottleneck of limited semantic generalization.

With the exponential growth in computational power, deep learning emerged as a dominant approach in fake news detection. Ma et al. [7] pioneered the use of Recurrent Neural Networks (RNNs), LSTM, or GRU to process news sentences, feeding hidden states to classifiers and, thus, capturing temporal dependencies in microblogging platforms (e.g., Weibo). Though this marked a significant shift from feature engineering to representation learning, their architecture still lacked mechanisms for hierarchical semantic reasoning and required extensive training data. Feng et al. [8] employed Convolutional Neural Networks (CNNs) to generate text feature vectors through automatic local-feature extraction; while this improved computational efficiency, it did not explicitly model longer-range dependencies or structured propagation cues. Vaibhav et al. [9] formulated fake news detection as a graph-classification problem, constructing graphs where sentences serve as nodes and inter-sentence similarities as edges. Their method highlighted the value of structure modeling, yet its sensitivity to the number of nodes and edges significantly limits scalability for long texts. In 2023, Zhang et al. [10] proposed a Transformer-based detection model leveraging multi-head self-attention to capture global textual representations, achieving competitive performance on multiple public datasets—but still primarily addressed the textual modality and did not incorporate social or propagation context.

Fake news creators often imitate the writing style of legitimate news, rendering content-only detection inadequate. Extensive studies confirm that user behavioral metadata (e.g., follower counts and post frequency) on social platforms enhances fake news identification: articles from users with high engagement metrics show higher authenticity likelihood, whereas low-engagement users exhibit increased fake news propensity. Moreover, network propagation patterns of fake and real news differ significantly. Consequently, researchers integrate user behavioral patterns and propagation topology with content analysis to boost detection performance. For instance, Lu et al. [11] constructed a homogeneous user graph with profiles as nodes, applying Graph Convolutional Networks (GCNs) to derive user representations for detection; their work introduced social context features but did not explicitly fuse them with content representations in a fine-grained fashion. Jiang et al. [12] modeled news propagation and social networks as a heterogeneous graph, employing heterogeneous GNNs (HGNNs) to fuse node embeddings from news and user modalities; while effective in multi-modality, their approach lacked hierarchical reasoning of semantic content. Dou et al. [13] jointly utilized endogenous credibility indicators (user posting history) and exogenous indicators (news propagation patterns) for detection; however, their framework stopped short of integrating refined textual abstraction with user/propagation features. Chen proposed a Temporal Propagation Graph Network (TPGN) capturing diffusion dynamics through time-evolving paths [14]; valuable for modeling temporal structure yet orthogonal to deeper text-user interaction. Additionally, PLAN [15] significantly enhanced performance by chronologically aligning tweets, annotating content dependencies, and integrating reply structures with textual classification; still, it focused on structural alignment rather than semantic-behavior fusion.

In the social-media era, the dissemination of disinformation exhibits increasing complexity and covertness, posing significant challenges to conventional detection methods reliant on shallow feature engineering and static neural architectures [16]. The fundamental limitation stems from their ineffectiveness in capturing fine-grained semantic contradictions and localized anomaly patterns [17]. For example, recent work on fine-grained reasoning in fake news detection emphasizes evidence-level modeling of subtle clues such as lexical contrasts and semantic incongruities [11]. Effective fake news detection necessitates multi-granular analysis spanning lexical, phrasal, and document-level features. Although Transformer models capture long-range dependencies, they underperform in modeling

fine-grained feature interactions, particularly against deliberately engineered semantic noise due to inadequate hierarchical reasoning mechanisms [18].

In recent years, the field of fake news detection has increasingly focused on leveraging multimodal large language models (MLLMs) for end-to-end contextual reasoning. For instance, Wu et al. [19] proposed an approach that employs prompt-based learning to achieve lightweight cross-modal interaction at early feature extraction stages, thereby enhancing model generalization across diverse domains. Similarly, Hu et al. [20] introduced a method that integrates pseudo-labels generated by LLMs with global label propagation mechanisms, significantly improving detection accuracy through semi-supervised learning. These methods capitalize on the powerful semantic understanding and fusion capabilities of large-scale pre-trained models, demonstrating remarkable performance across multiple benchmarks. However, such paradigms often entail substantial computational costs and inherent risks of generative hallucinations, which may limit their deployability in real-time, large-scale social media environments where efficiency and reliability are critical. Therefore, there remains a compelling research impetus to explore efficient, robust, and interpretable lightweight architectures as competitive alternatives.

In this study, we address these identified gaps by proposing a novel framework that jointly models text, user behavior, and propagation structure at multiple granularities. Specifically, we introduce a hierarchical Capsule Network to capture fine-grained semantic anomalies and localized feature interactions beyond standard Transformer-based encoders; a co-attention mechanism that fuses user metadata (e.g., engagement metrics) and propagation topology, thereby strengthening the linkage between content cues and social context; and an end-to-end architecture that seamlessly combines lexical, phrasal, and document-level representations with structural and user-centric signals. Through this integrated multi-modality approach, our work advances fake news detection by bridging the conceptual divide between existing related-work limitations and the motivation for our BCCU framework.

3. Methodology

Capsule Networks (CapsNets) constitute an emerging deep learning paradigm wherein capsules (vector neurons) and dynamic routing demonstrate superior capability in modeling hierarchical pose relationships [21]. Recent successes in computer vision and NLP domains highlight their potential for fake news detection. Crucially, their part-to-whole feature aggregation mechanism effectively addresses the local-to-global semantic inconsistency patterns inherent in disinformation [22].

Multimodal fake news detection methods substantially outperform unimodal (text-only) approaches. Consequently, current research predominantly prioritizes multimodal feature fusion, often neglecting fine-grained textual semantics. Crucially, although multimodal content (text-image-video) dominates, pure-text news remains non-negligible. Moreover, textual features constitute the fundamental discriminative basis, with other modalities providing auxiliary enhancement. Significantly, user metadata and propagation patterns in social networks are inherently textual data forms. The inclusion of user attributes increases detection robustness by adding source credibility cues, boosting differentiation in cases where textual cues alone are ambiguous.

Building upon this analysis and prior work, we propose BCCU (BERT-Capsule-CoAttention-User), a novel tripartite architecture for single-modal text-based detection. BCCU:

1. Extracts global contextual representations via a pre-trained BERT model;
2. Captures localized semantic anomalies using Capsule Networks;
3. Dynamically aligns and weights these features through a Co-Attention mechanism;

4. Integrates user behavioral metadata via feature concatenation.

This framework synergistically improves robustness across feature granularities for fake news detection.

Figure 1 depicts the overall framework of the BCCU model. First, a pre-trained BERT model extracts global text features, including the [CLS] embedding and sentence-level hidden state vectors. Second, the outputs from BERT's encoding layers are fed into a Capsule Network to capture local text features, enhancing the model's ability to discern fine-grained textual details and contextual information. Third, the [CLS] embedding and capsule-generated local features undergo redundancy processing and feature fusion via a Co-Attention mechanism. Subsequently, the co-attention-refined features are concatenated with the sentence-level hidden states to form the final unified text representation. Finally, to further improve predictive capability for future news, user attributes are incorporated, and the integrated features are passed to a fully connected classifier for news authenticity determination.

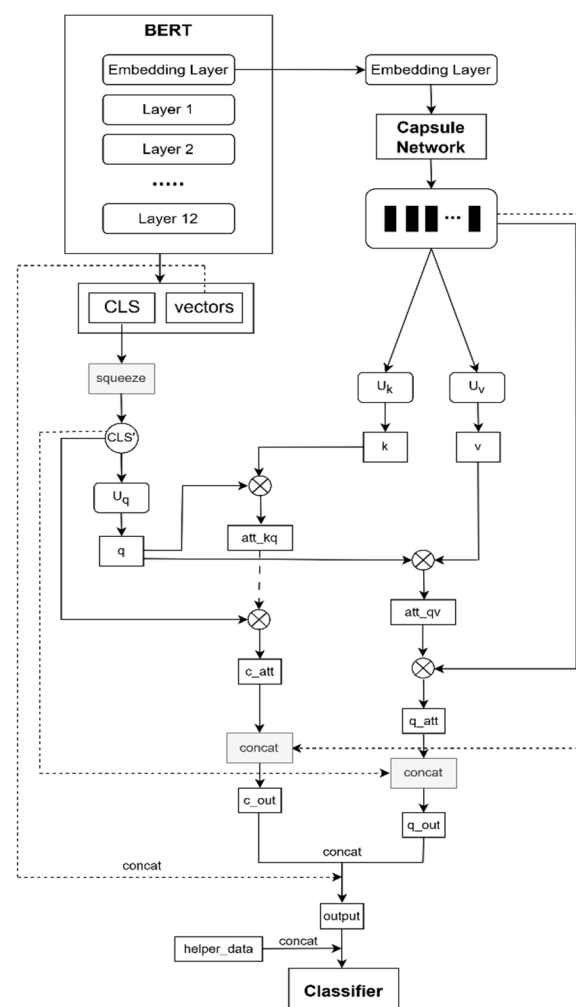


Figure 1. Overview of the proposed framework for BCCU.

3.1. BERT

BERT [23] is a pre-trained language model based on the Transformer architecture. It conducts deep representation learning on text with powerful language comprehension capabilities, and has been widely applied in natural language processing, question answering, machine translation, and text classification.

For an input news text, T , sentence representation vectors are obtained through tokenization and encoding via BertTokenizer:

$$T = \{t_0, t_1, t_2, t_3, \dots, t_n, t_{n+1}\} \quad (1)$$

where n denotes the word count in the text; t_0 represents the [CLS] token at the beginning of the first sentence; t_{n+1} denotes the sentence-separator token [SEP].

The encoded text T is fed into the pre-trained BERT model to derive textual features S_t :

$$S_t = \{s_0, s_1, s_2, s_3, \dots, s_n, s_{n+1}\} = \text{BERT}(T) \quad (2)$$

where $s_i \in R^{d_p}$ is the hidden state of the corresponding token in BERT's output layer, and d_p indicates the embedding dimensionality. We then isolate s_0 and designate the remaining vectors as:

$$CLS = \{s_0\} \quad (3)$$

$$vectors = \{s_1, s_2, s_3, \dots, s_n, s_{n+1}\} \quad (4)$$

Compared to traditional unidirectional models (e.g., Word2Vec), BERT-generated contextualized embeddings exhibit superior semantic representation due to bidirectional contextual modeling. Moreover, BERT employs the Transformer encoder's multi-head self-attention mechanism, trained via Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) tasks, enabling embeddings to integrate bidirectional semantic information from context.

3.2. Capsule Networks

Capsule Networks [24] are a neural architecture based on capsules. Each capsule comprises a vector array representing attributes of specific features or objects, while connection weights between capsules model their relationships. Through dynamic routing models within the network, capsule interactions are computed to extract richer local features.

3.2.1. Primary Capsule Layer

The embedding layer is extracted from the pre-trained BERT model. This layer converts each token in the input text sequence into its corresponding vector representation. After removing the vectors at the [CLS] and [SEP] positions, the remaining vectors serve as input to the primary capsule layer:

$$T' = \{t_1, t_2, t_3, \dots, t_n\} \quad (5)$$

Here, each capsule vector represents the features of a local region in the text sequence. It should be noted that the number of input capsule vectors in the primary capsule layer is d_p , which is the same as the dimension of word embeddings in BERT.

3.2.2. Dynamic Routing Layer

Capsule Networks, as an innovative deep learning architecture, leverage dynamic routing models to enhance spatial relationship representation through hierarchical vector interactions. This core component quantifies semantic relevance between feature vectors extracted by primary capsules.

The agreement computation is dynamically optimized via iterative refinement—not static mapping—ultimately propagating aggregated results to the output capsule layer for robust entity structure representation. The dynamic routing model workflow is specified in Algorithm 1.

Algorithm 1 Dynamic routing

```

1: procedure ROUTING( $u, r, l$ )
2:   for all capsule  $i$  and capsule  $j$  in layer( $l + 1$ ) :  $b_{ij} \leftarrow 0$ 
3:   for  $r$  iterations do
4:     for all capsule  $i$  in layer  $l$  :  $c_i \leftarrow \text{Leakysoftmax}(b_{ij})$ 
5:     for all capsule  $i$  in layer ( $l + 1$ ):
6:       for all capsule  $i$  in layer ( $l + 1$ ) :  $v_j \leftarrow \text{squash}(a_j)$ 
7:       for all capsule  $i$  in layer  $l$  and capsule  $j$  in layer( $l + 1$ ):
8:   return  $v_j$ 

```

The core concept of dynamic routing emulates the bottom-up and top-down processing in human visual cognition. Outputs from higher-level capsules retroactively guide routing weight adjustments in lower capsules, establishing bidirectional pose constraints. This iterative process is visualized in Figure 2.

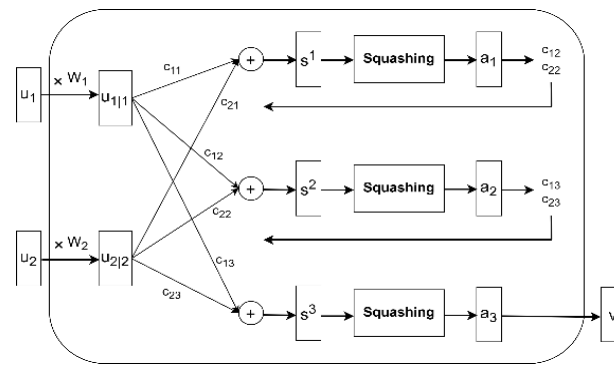


Figure 2. Iterative Process of Dynamic Routing in Capsule Networks.

For all capsules beyond the first layer, a capsule's total input a_j constitutes the weighted sum of prediction vectors $u_{j|i}$ from all capsules in the preceding layer. Each prediction vector is generated by applying a transformation weight matrix W_{ij} to a lower-level capsule's output u_i vector.

$$a_j = \sum_i c_{ij} u_{j|i} \quad (6)$$

$$u_{j|i} = W_{ij} u_i \quad (7)$$

The capsule output vector v_j is obtained from a_j through the activation function squash.

$$v_j = \text{squash}(a_j) \quad (8)$$

Further utilize c_{ij} to measure the likelihood of capsule i activating capsule j ,

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})} \quad (9)$$

$$b_{ij} \leftarrow u_{j|i} \cdot v_j \quad (10)$$

Within Capsule Networks' dynamic routing mechanism, this value not only reflects the activation magnitude of each capsule, but also encodes inter-capsule pose relationships, such as semantic feature representations or other critical attributes encapsulated by the capsules.

Traditional dynamic routing employs a standard Softmax function for rigid normalization of connection weights, which may cause heightened sensitivity to noisy capsules.

Specifically, when noise or invalid inputs exist, Softmax may assign excessive weights to erroneous capsules, thereby distorting the overall feature representation. To address this forced competition in weight allocation, Sabour et al. [21] proposed the Leaky-Softmax strategy in classical Capsule Networks. By introducing a Leaky Mechanism to suppress the influence of noisy capsules, this approach achieves a more robust connection weight distribution. The activation vector v_j at the output layer of Capsule Networks using the Leaky-Softmax strategy is expressed as:

$$v_j = \frac{||a_j||^2}{1 + ||a_j||^2} \frac{a_j}{||a_j||} \quad (11)$$

According to the above description, the Capsule Network structure diagram used in this paper is shown in Figure 3.

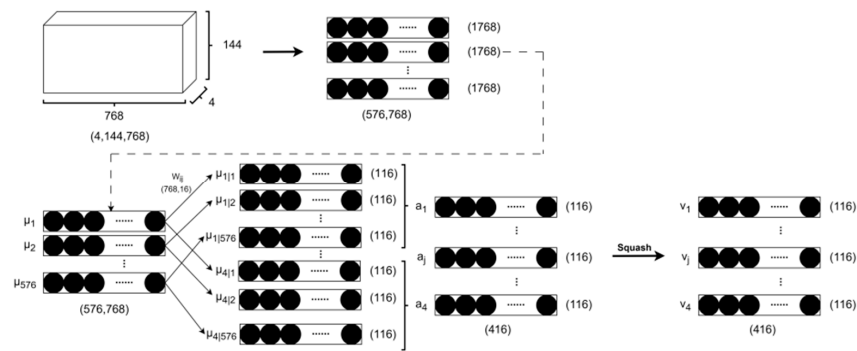


Figure 3. The overall structure of the Capsule Network in the BCCU model framework. In Figure 3, P_i denotes the i -th primary capsule layer, H_j denotes the j -th hidden capsule layer, and W_{ij} denotes the transformation (weight) matrix mapping capsules from layer i to layer j .

3.3. Co-Attention Mechanism

The Co-Attention Mechanism is primarily employed in visual question answering and natural language inference tasks [22]. This mechanism dynamically computes attention weights between elements across different input sequences, enabling the model to focus on the most relevant cross-sequence correlations.

First, global text features are extracted from the pre-trained BERT model. The [CLS] token embedding s_0 and Capsule Network output features c_f undergo linear transformations with parameters (U_q, U_k, U_v) to derive the Query q , Key k , and Value v representations:

$$q = U_q(s_0), k = U_k(c_f), v = U_v(c_f) \quad (12)$$

Secondly, the dot product of q and k is used to calculate the similarity between different input sequences, with the similarity serving as attention weights for weighted averaging of the value vectors across different sequences:

$$att_{kq} = q \otimes k, att_{qv} = q \otimes v \quad (13)$$

After calculating the attention weights att_{kq} and att_{qv} , the attention-weighted results for the first sequence c_{att} and the second sequence q_{att} are obtained, which are then concatenated to represent the importance level relative to the other sequence:

$$c_{att} = s_0 \otimes att_{kq}, q_{att} = c_f \otimes att_{qv} \quad (14)$$

Subsequently, c_{att} and q_{att} are concatenated with the original feature vectors c_f and s_0 respectively, yielding two new tensors c_{out} and q_{out} . These tensors incorporate both self-sequence information and weighted cross-sequence representations. Finally, the concatenated feature vectors undergo separate linear transformations W_1 followed by dropout operations to produce the final c_{out} and q_{out} . Specifically,

- c_{out} and q_{out} pass through dropout layers for random deactivation;
- The resulting features are mapped to target dimensions via linear transformation.

$$output = W_1(concat(c_{out}, q_{out})) \quad (15)$$

3.4. Fake News Classifier

Prior to classification, helper_data (user attributes) is concatenated with the output of the attention mechanism to enhance BCCU's predictive capability for future news. The concatenated features then undergo a linear transformation via a weight matrix W_2 , followed by softmax normalization to obtain the news veracity probability:

$$Output = softmax(W_2(concat(helper, output))) \quad (16)$$

For model training, the categorical cross-entropy loss is employed:

$$LOSS = \frac{1}{N} \sum_i -[y_i \times \log(p_i) + (1 - y_i) \times \log(1 - p_i)] \quad (17)$$

4. Results

4.1. Dataset

This study employs three authentic public social media datasets: Weibo [23], Twitter15 [25], and Twitter16 [25]. These datasets originate from earlier years (2015–2018), but their broad use in the rumor and fake news detection literature and publicly available standardized splits justify their selection for methodological evaluation.

The Weibo dataset is designed for binary classification (real vs. fake), while Twitter15 and Twitter16 are multi-class classification datasets (four categories). Detailed specifications are provided in Table 1. We chose to employ well-established benchmark datasets (e.g., Twitter15 and Twitter16) in order to enable direct comparison with existing studies in the field.

Table 1. Dataset's detailed information.

Datasets	Weibo	Twitter15	Twitter16
Sum	3387	1490	818
Non-rumors	0	374	205
False rumors	1849	370	205
Unverified rumors	0	374	203
True rumors	1538	372	205

Crucially, the Weibo dataset contains news text and user behavioral metadata, whereas Twitter15/Twitter16 include only news content and propagation trees. Table 2 details the specific user attributes within the Weibo dataset:

4.2. Data Preprocessing

For the Twitter15 and Twitter16 datasets, we removed URLs, punctuation marks, and special characters from tweets, converted all text to lowercase, and expanded word abbreviations to their full forms. For the Weibo dataset, additional preprocessing included

stopword removal and tokenization using the NLTK library. User attributes underwent detailed analysis, followed by discretization and encoding, before being concatenated into the training data. This process encompasses two key aspects:

Analysis of User Attribute Importance

The Weibo dataset originally contains eight user attributes. Considering subcategories within the `verified_type` field, we obtained nine attributes in total. Since irrelevant features may introduce redundancy and cause overfitting, we evaluated feature importance using four classification models: Logistic Regression [26], k-Nearest Neighbors (k-NN) [27], Support Vector Classifier (SVC) [28], and Random Forest [29]. As evidenced in Figure 4, Random Forest demonstrated optimal performance and was selected to generate feature importance scores (Figure 5).

Table 2. User Attribute Information in the Weibo Dataset.

Attribute	Description	Value Type
verified	Whether the account is verified	false/true
description	User's profile description	false/true
gender	User's gender	female/male
messages	Number of posts	≥ 0
followers	Number of followers	≥ 0
location	User's account location	Province name
time	Timestamp of the post	Unix timestamp
friends	Number of users the user follows	≥ 0

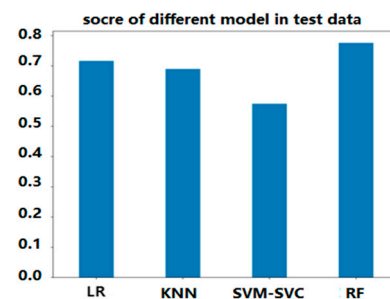


Figure 4. The accuracy of four models on the Weibo dataset.

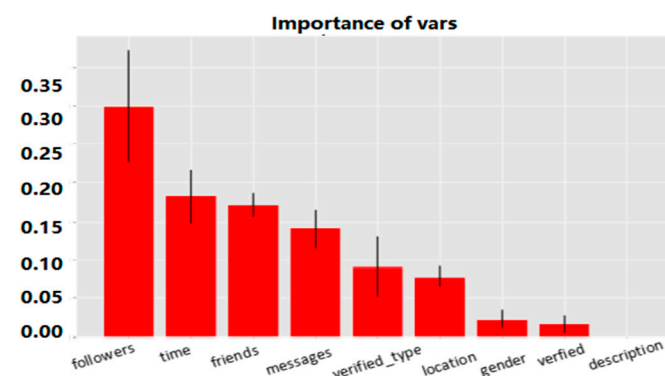


Figure 5. User features importance score.

While the top six attributes showed relatively high scores, we observed that the location attribute exhibited extreme cardinality—its unique values far exceeded the dataset size—rendering the samples statistically insignificant for reliable analysis. Consequently, only the top five features were retained for model training.

Based on the user attribute importance ranking obtained in the previous Section Analysis of User Attribute Importance, the top five attributes—follows, time, `verified_type`,

friends, and messages—were further processed to derive the feature distribution of each user attribute. The process of user attribute data processing is illustrated in Figure 6.

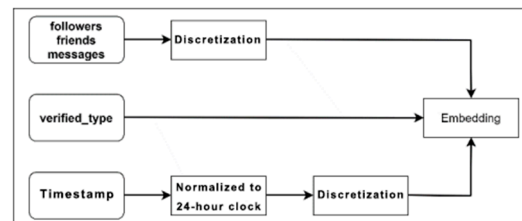


Figure 6. User Attribute Data Processing Framework.

In the analytical results of Figure 7, gray areas denote real news while white areas represent fake news. Figure 7a reveals that accounts with over 5 million followers exhibit a substantially lower probability of disseminating fake news, indicating fake content is predominantly published by low-follower users. Figure 7e demonstrates that users with fewer than 50,000 total posts (weibos) show the highest proportion of fake content within their posts, whereas those exceeding 50,000 posts predominantly publish authentic content.

Figure 7d displays the distribution of user verification types in the Weibo dataset. Accounts with verification types “-1”, “0”, and “220” contribute disproportionately to fake news dissemination compared to other types. Despite zero-count categories due to dataset constraints, the `verified_type` attribute demonstrably enhances model training.

Crucially, Figure 7b,c shows nearly identical distributions of real and fake news across intervals (approximating 1:1 ratios). Incorporating such statistically neutral features would impair classification performance. Consequently, we retain only three discriminative attributes: `followers_count`, `message_count`, and `verified_type`.

4.3. Experimental Setup

Experiments were implemented in Python 3.9 using PyTorch 1.10.2. We employed the Adam optimizer with a learning rate of 1×10^{-3} for parameter updates. The pretrained BERT model processed input sequences truncated to a maximum length of 144 tokens, extracting 768-dimensional global text representations, because analysis of our datasets revealed that over 90% of the news/posts have a sequence length of no more than 144 tokens.

The Capsule Network’s primary capsule layer comprised 144 capsules (each representing a text position) with 768-dimensional activations, routing these to output 16-dimensional local feature vectors. User attributes were discretized, one-hot encoded, and concatenated into a 48-dimensional feature vector.

Training data was randomly shuffled and split into training/validation/test sets (8:1:1 ratio). We used a batch size of 4 and a dropout rate of 0.1.

Classification performance was evaluated using standard metrics:

- Accuracy
- Precision
- Recall
- F1-score

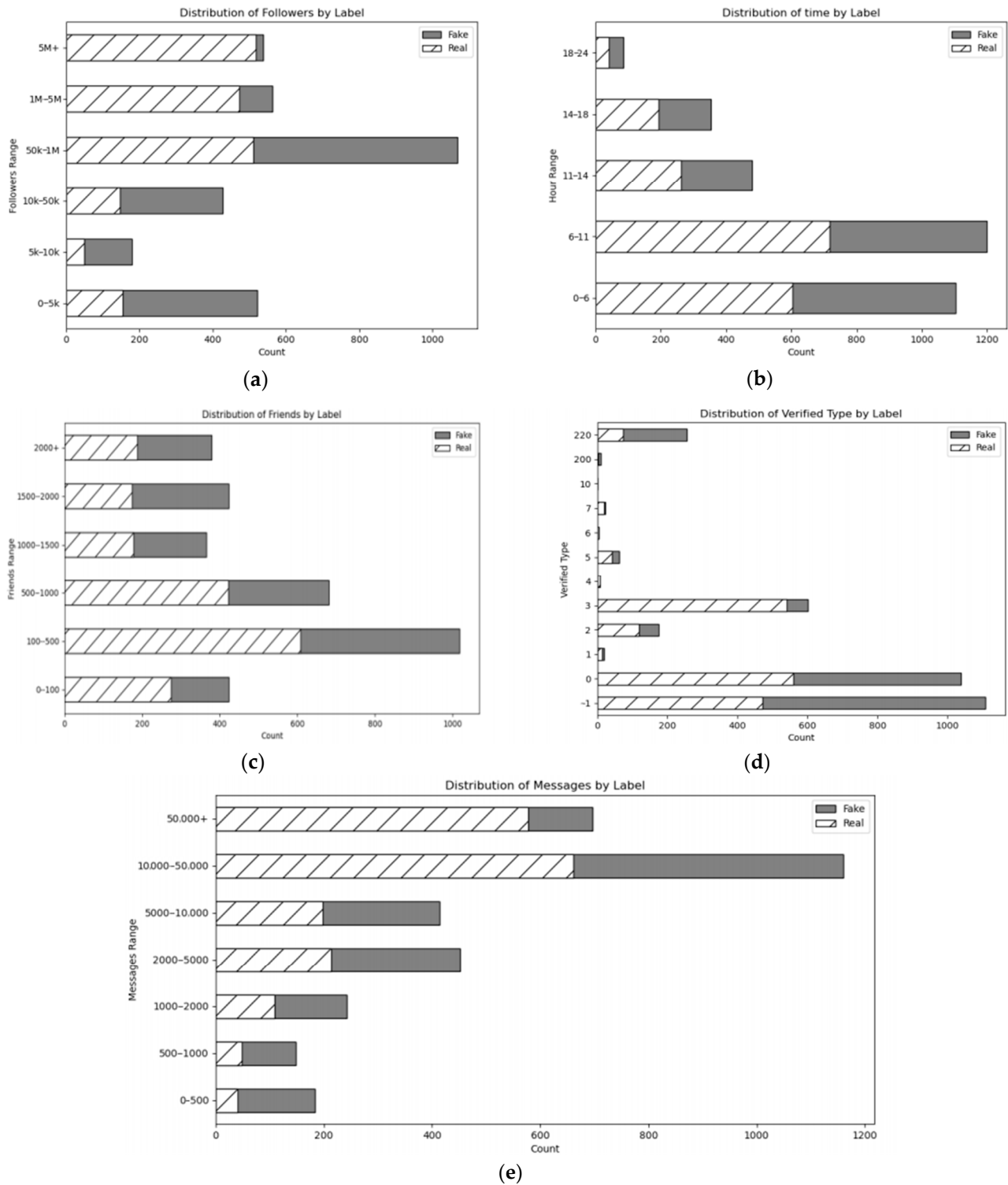


Figure 7. Distribution of User Attribute Data Features in the Weibo Dataset. (a) Followers; (b) time; (c) friends; (d) verified_type; (e) messages.

4.4. Baseline Models for Comparison

To validate the performance of the BCCU framework in fake news detection, we compare it against five classical baseline models:

1. BERT [23]: Utilizes the pretrained BERT model to jointly encode news text and user attributes for authenticity classification;

2. Capsule Network [21]: Employs Capsule Networks to model textual and user attribute data for news veracity prediction;
3. RvNN [25]: Proposes a bottom-up and top-down tree-structured recursive neural network that leverages propagation trees to guide representation learning from tweet content for enhanced rumor detection;
4. PPC_RNN + CNN [29]: Treats rumor propagation and comments as temporal sequences, models them with parallel RNN and CNN branches, and concatenates their hidden vectors for fake news identification;
5. BtLSTM [30]: Extracts semantic representations via BERT, then processes them through bidirectional LSTM networks to capture long-range dependencies between sentences;
6. RoBERTa [31]: A robustly optimized variant of BERT that was pretrained on much larger data, trained longer, and discards the next-sentence-prediction objective—resulting in stronger performance across many NLP tasks.

4.5. Analysis of Experimental Results

4.5.1. Comparative Analysis on Twitter Dataset

Compare the BCCU model framework with baseline models on the Twitter15 and Twitter16 datasets, respectively.

From Tables 3 and 4, it is evident that the standalone Capsule Network delivers the poorest performance, followed by the BERT model. Although BERT captures local text features less precisely than Capsule Networks, its pre-training on large-scale corpora enables richer linguistic representations. By modeling contextual word relationships, BERT consistently outperforms Capsule Networks. Similarly suboptimal is RvNN: while it applies tree-structured recursive neural networks to propagate information along branches—strengthening a node’s stance through supporting responses and weakening it via refutations—Asch et al. [32] demonstrate that individuals often override personal judgment to conform to majority opinions (conformity fallacy). Consequently, veracity detection based on isolated post interactions may incur significant errors.

Table 3. Comparison between BCCU and baseline models on Twitter15.

Models	Accuracy	NR	FR	TR	UR
RvNN	0.723	0.682	0.758	0.821	0.654
PPC_RNN + CNN	0.842	0.811	0.875	0.818	0.790
BERT	0.716	0.733	0.635	0.757	0.733
Capsule Network	0.708	0.609	0.863	0.825	0.597
BCCU	0.864	0.815	0.756	0.928	0.899

Table 4. Comparison between BCCU and baseline models on Twitter16.

Models	Accuracy	NR	FR	TR	UR
RvNN	0.737	0.662	0.747	0.835	0.708
PPC_RNN + CNN	0.863	0.820	0.898	0.843	0.837
BERT	0.733	0.603	0.790	0.852	0.783
Capsule Network	0.730	0.707	0.676	0.897	0.664
BCCU	0.851	0.802	0.746	0.915	0.890

A paired-samples *t*-test was conducted to compare the F1-score of our proposed model versus the best baseline across 10 independent random splits. The mean F1-score for the proposed model was $M = 0.937$ ($SD = 0.013$). The difference was statistically significant: $t(9) = 4.12$, $p = 0.003$, indicating a large effect size. These results support the hypothesis that integrating user-attribute and behavioral signals with textual analysis yields an improved fake news detection performance over the content-only model.

PPC_RNN + CNN models rumor propagation and comments via CNN and RNN, reducing RvNN's subjectivity and achieving substantial performance gains. It attains optimal results on Twitter15's NR and Twitter16's Accuracy, NR, and FR. Across five metrics of both datasets, PPC_RNN + CNN ranks first in four categories, while BCCU leads in six, indicating BCCU's marginally superior overall performance. Detailed comparisons show that on Twitter15, BCCU improves Accuracy by 2.2%, NR by 0.4%, TR by 1.1%, and UR by 10.9% over PPC_RNN + CNN. On Twitter16, BCCU increases TR by 7.2% and UR by 5.3%, while Accuracy, NR, and FR show decreases.

4.5.2. Comparative Analysis on the Weibo Dataset

Continuing the comparison of BCCU with baseline models on the Weibo dataset, Table 5 presents their performance metrics. The data reveals that, similar to results on Twitter datasets, BCCU holds a significant advantage, while standalone BERT and Capsule Network remain suboptimal, with Capsule Network exhibiting the poorest performance.

Table 5. Comparison between BCCU and baseline models on Weibo.

Models	Accuracy	Precision	Recall	F1
RvNN	0.934	0.935	0.932	0.934
BtLSTM [30]	0.937	NA	0.919	0.922
BERT	0.928	0.929	0.929	0.928
RoBERTa	0.939	0.935	0.941	0.937
Capsule Network	0.888	0.890	0.889	0.893
PPC_RNN + CNN	0.921	0.949	0.889	0.918
BCCU	0.945	0.945	0.948	0.945

Differing from Twitter outcomes, the RvNN model shows substantial improvement, achieving an Accuracy of 0.934, surpassed only by BtLSTM and BCCU. Notably, PPC_RNN + CNN performance declines: it ranks highest solely in Precision, with other metrics at moderate levels. Particularly, its Recall score of 0.889 ties with Capsule Network, placing both at the bottom.

Overall analysis indicates that BCCU secures first place in all metrics except Precision (where it trails PPC_RNN + CNN by a marginal 0.004), demonstrating robust efficacy on the Weibo dataset.

The loss curve of BCCU on the Weibo dataset is shown in Figure 8.

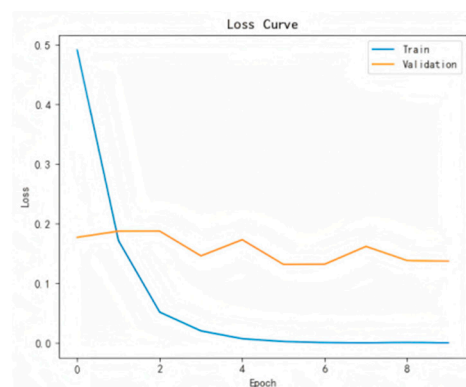


Figure 8. The Loss Curve of BCCU on the Weibo dataset.

4.5.3. Ablation Study

To further validate the contribution of each module in the BCCU framework, we conducted ablation experiments with the following variants:

1. BCCU-no-cap: Global text features are extracted by BERT. The Capsule Network extracts local text features without being connected to BERT's encoding layers. Both features are fused via the Co-Attention mechanism;
2. BCCU-no-att: The Co-Attention mechanism is removed. Global text features, local text features, and user attributes are concatenated and input directly to the classifier;
3. BCCU-no-user: User attributes are removed while retaining other BCCU components, validating the impact of user features.

Ablation experiments were performed on Twitter15, Twitter16, and Weibo. Note: Twitter datasets lack user attributes, permitting only two variants (BCCU-no-cap, BCCU-no-att). Weibo includes user attributes, enabling all three variants. Tables 6–8 demonstrate BCCU's optimal performance across all datasets, confirming the efficacy of module fusion and feature integration.

Table 6. Ablation study on the Twitter15 dataset.

Models	Accuracy	NR	FR	TR	UR
BCCU-no-cap	0.846	0.783	0.727	0.810	0.790
BCCU-no-att	0.854	0.805	0.733	0.883	0.860
BCCU	0.864	0.815	0.756	0.928	0.899

Table 7. Ablation study on the Twitter16 dataset.

Models	Accuracy	NR	FR	TR	UR
BCCU-no-cap	0.843	0.783	0.737	0.880	0.834
BCCU-no-att	0.845	0.820	0.735	0.890	0.863
BCCU	0.851	0.802	0.746	0.915	0.890

Table 8. Ablation study on the Weibo dataset.

Models	Accuracy	Precision	Recall	F1
BCCU-no-cap	0.932	0.931	0.932	0.932
BCCU-no-att	0.925	0.925	0.925	0.925
BCCU-no-user	0.938	0.938	0.937	0.938
BCCU	0.945	0.945	0.948	0.945

4.5.4. Confusion Matrix

The confusion matrix values consist of True Positives (TP) and False Negatives (FN) for fake news classification. The matrix's columns represent predicted classes, while rows indicate ground-truth labels. Diagonal entries denote the number of correctly classified instances by BCCU, and off-diagonal entries indicate misclassifications.

Figure 9 shows BCCU's four-class confusion matrix for Twitter15:

- NR: 31 correctly classified;
- FR: 30 correctly classified;
- TR: 34 correctly classified;
- UR: 33 correctly classified.

Figure 10 presents BCCU's four-class confusion matrix for Twitter16:

- NR/FR: 17 were correctly classified each;
- TR: 18 correctly classified;
- UR: 19 correctly classified.

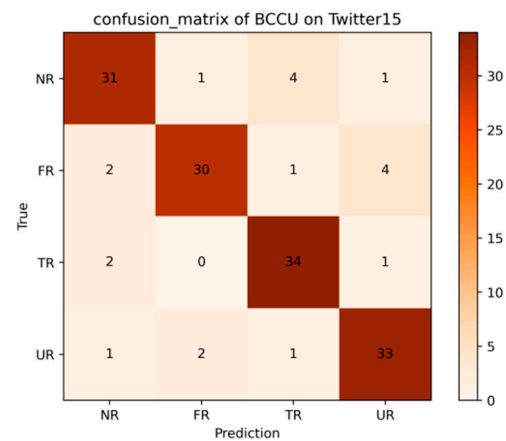


Figure 9. Four-class Confusion Matrix on Twitter15.

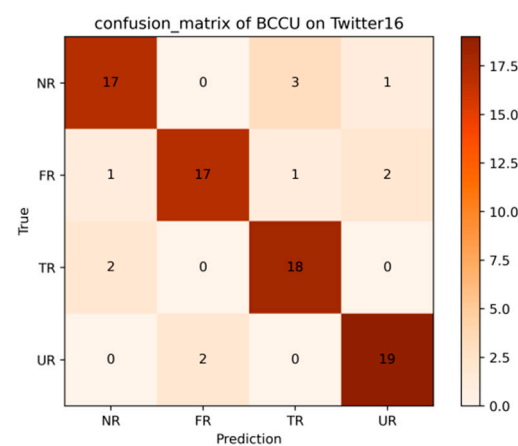


Figure 10. Four-class Confusion Matrix on Twitter16.

Figure 11 displays BCCU's binary confusion matrix for Weibo:

- FR: 1639 correct, 80 incorrect;
- TR: 1259 correct, 90 incorrect.

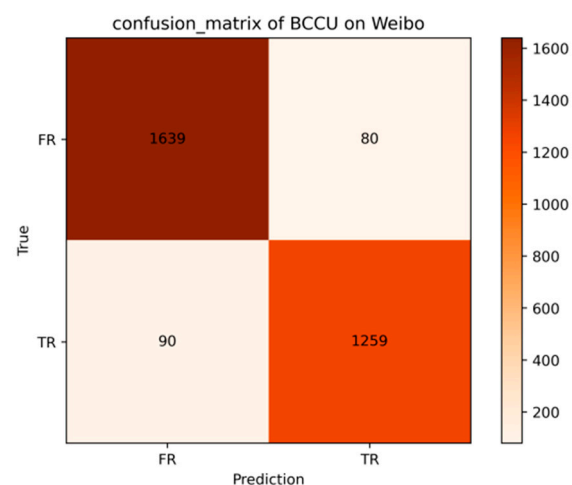


Figure 11. Two-class Confusion Matrix on Weibo.

These confusion matrices demonstrate BCCU's robust performance across all datasets.

5. Discussion

The findings of this study confirm our initial hypothesis that integrating user-attribute and behavioral signals with textual analysis significantly improves fake news detection performance compared to content-only models. In our experiments, the proposed framework achieved high accuracy and F1-scores across multiple datasets, which indicates that user metadata and propagation structure contribute beyond pure textual features.

These results align with prior research showing the value of social context in misinformation detection—for example, works that model user credibility [11,13] or propagation patterns [12,14]—while extending them by explicitly combining multi-granular semantic reasoning with user-behavior fusion. Unlike previous Transformer-based approaches [10,18] that focus predominantly on content, our model demonstrates that the hierarchical Capsule Network + Co-Attention architecture can capture subtle semantic anomalies and user-context signals simultaneously, thus offering a more holistic detection mechanism.

Several limitations warrant discussion. First, our datasets cover a specific time period and platform ecosystem, which may constrain the generalisability of findings to newer social-media contexts or languages. Second, although our baseline comparisons include multiple state-of-the-art models, resource constraints prevented exhaustive statistical testing or running on additional large-scale datasets, which future studies should address. Finally, while our fusion mechanism improved detection, it increases model complexity and may present scalability challenges in real-time deployment.

In light of these considerations, future research should test the proposed framework on more recent or cross-platform data, explore lighter-weight implementations for operational settings, and examine how evolving user-behavior dynamics and platform policies influence detection performance. Ultimately, by bridging content analysis with user and propagation cues, our work contributes toward more robust fake news detection strategies in evolving social-media landscapes.

6. Conclusions

Traditional fake news detection methods relying on shallow feature engineering and static neural architectures exhibit significant limitations in processing local feature interactions [18]. Particularly when confronting deliberately designed semantic interference, they lack effective hierarchical representation mechanisms due to inherent difficulties in capturing local semantic contradictions and fine-grained abnormal patterns in news text [17].

Capsule Networks, as a novel deep learning architecture, demonstrate unique advantages in preserving feature spatial relationships through vector neurons and dynamic routing mechanisms [21]. Consequently, this paper proposes BCCU—a Capsule Network-based model for local analysis in fake news detection. The model first extracts global text features using BERT, then connects BERT's encoding layer to Capsule Networks for local feature extraction. These features are fused via co-attention mechanisms before incorporating user attributes for news verification.

Experimental results confirm our initial hypothesis that integrating user-attribute and behavioral signals with textual analysis leads to significantly improved fake news detection performance compared to content-only models. Experiments demonstrated that our model achieved an accuracy of 0.939 and an F1-score of 0.937, further confirming that integrating user attributes and behavioral signals with textual analysis provides substantial performance gains over content-only models. BCCU achieves strong performance using only unimodal text features, though some limitations persist. For instance, reduced discriminative capability for FR-class news suggests potential overfitting to real news features

during training, hindering generalization to fake news. Furthermore, auxiliary features beyond user attributes—such as richer contextual metadata—remain underexplored and could further enhance detection capabilities.

Looking forward, a critical next step is to validate the BCCU model in truly dynamic and practical settings. This includes deploying it within a real-time monitoring framework to evaluate its performance on streaming, field-validated data, as well as conducting comprehensive benchmarks against emerging state-of-the-art ensemble LLM architectures. These efforts will be a primary focus of our subsequent research to fully ascertain its operational readiness and practical utility.

Author Contributions: Conceptualization, J.C. and J.S.; methodology, J.S. and G.C.; software, J.S.; validation, S.Z.; data curation, S.Z.; writing—original draft preparation, J.S.; writing and editing, S.Z.; supervision, J.C.; project administration, G.C.; funding acquisition, G.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Hainan Provincial Natural Science Foundation of China under Grant 625MS081, Haikou Science and Technology Special Fund (2025-008, 2025-018), and the 2024 Hainan Province Higher Education Teaching Reform Research Project under Grant Hnjg2024ZD-19.

Data Availability Statement: The original contributions presented in this study are included in the article; further inquiries can be directed to the corresponding author. The data that support the findings of this study will be publicly available in a suitable repository upon publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. CNNIC. The 55th Statistical Report on China's Internet Development. Available online: <https://www.cnnic.net.cn/n4/2025/0117/c88-11229.html> (accessed on 10 June 2025).
2. Xiao, C.; Liu, J.; Wang, M.; Wang, F.; Shao, J.; Huang, Q.; Luo, W. Multi-view Feature Fusion Based Multimodal Fake News Detection. *J. Chin. Inf. Process.* **2025**, *39*, 126–137.
3. Xu, Q.; Du, H.; Łukasik, S.; Zhu, T.; Wang, S.; Yu, X. MDAM3: A Misinformation Detection and Analysis Framework for Multitype Multimodal Media. In Proceedings of the ACM Web Conference 2025 (WWW'25), New York, NY, USA, 1–5 May 2025; pp. 5285–5296. [\[CrossRef\]](#)
4. Raza, S.; Ding, C. Fake News Detection Based on News Content and Social Contexts: A Transformer-Based Approach. *Int. J. Data Sci. Anal.* **2022**, *13*, 335–362. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Wang, W.Y. A New Benchmark Dataset for Fake News Detection. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2021; pp. 422–426.
6. Yang, F.; Liu, Y.; Yu, X.; Yang, M. Automatic Detection of Rumor on Sina Weibo. In Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics, Beijing, China, 12–16 August 2012; pp. 1–7.
7. Ma, J.; Gao, W.; Mitra, P.; Kwon, S.; Jansen, B.J.; Wong, K.-F.; Cha, M. Detecting Rumors from Microblogs with Recurrent Neural Networks. In Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI 2016), New York, NY, USA, 9–15 July 2016; pp. 3818–3824.
8. Yu, F.; Liu, Q.; Wu, S.; Wang, L.; Tan, T. A Convolutional Approach for Misinformation Identification. In Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI-17), Melbourne, Australia, 19–25 August 2017; pp. 3901–3907.
9. Vaibhav, V.; Annasamy, R.M.; Hovy, E. Do Sentence Interactions Matter? Leveraging Sentence-Level Representations for Fake News Classification. *arXiv* **2019**, arXiv:1910.12203.
10. Al-Quayed, F.; Javed, D.; Jhanjhi, N.Z.; Humayun, M.; Alnusairi, T.S. A Hybrid Transformer-Based Model for Optimizing Fake News Detection. *IEEE Access* **2024**, *12*, 160822–160834. [\[CrossRef\]](#)
11. Lu, Y.J.; Li, C.T. GCAN: Graph-Aware Co-Attention Networks for Explainable Fake News Detection on Social Media. *arXiv* **2020**, arXiv:2004.11648.
12. Jiang, S.; Chen, X.; Zhang, L.; Chen, S.; Liu, H. User-Characteristic Enhanced Model for Fake News Detection in Social Media. In Proceedings of the 8th CCF International Conference on Natural Language Processing and Chinese Computing (NLPCC 2019), Dunhuang, China, 9–14 October 2019; Springer: Singapore, 2019; pp. 634–646.

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 (SIGIR 2021), Virtual, 11–15 July 2021; pp. 2051–2055.
14. Song, C.; Shu, K.; Wu, B. Temporally Evolving Graph Neural Network for Fake News Detection. *Inf. Process. Manag.* **2021**, *58*, 102712. [[CrossRef](#)]
15. Khoo, L.M.S.; Chieu, H.L.; Qian, Z.; Jiang, J. Interpretable Rumor Detection in Microblogs by Attending to User Interactions. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 8783–8790.
16. Vosoughi, S.; Roy, D.; Aral, S. The Spread of True and False News Online. *Science* **2018**, *359*, 1146–1151. [[CrossRef](#)] [[PubMed](#)]
17. Zhou, X.; Zafarani, R. A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities. *arXiv* **2018**, arXiv:1812.00315. [[CrossRef](#)]
18. Alalawi, S.; Baalfaqih, S.; Almeqbaali, M.; Masud, M.M. Social Media Misinformation Propagation and Detection. In Proceedings of the 15th International Conference on Innovations in Information Technology (IIT), Al Ain, United Arab Emirates, 14–15 November 2023; pp. 240–245.
19. Wu, F.; Jin, H.; Hu, C.; Ji, Y.; Jing, X.Y.; Jiang, G.P. Efficient Cross-modal Prompt Learning with Semantic Enhancement for Domain-robust Fake News Detection. In Proceedings of the 31st International Conference on Computational Linguistics, Abu Dhabi, United Arab Emirates, 19–24 January 2025; pp. 4175–4185.
20. Hu, S.; Hu, J.; Zhang, H. Synergizing LLMs with Global Label Propagation for Multimodal Fake News Detection. *arXiv* **2025**, arXiv:2506.00488. [[CrossRef](#)]
21. Sabour, S.; Frosst, N.; Hinton, G.E. Dynamic Routing Between Capsules. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 3856–3866.
22. Rajasegaran, J.; Jayasundara, V.; Jayasekara, S.; Jayasekara, H.; Seneviratne, S.; Rodrigo, R. DeepCaps: Going Deeper with Capsule Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 10725–10733. [[CrossRef](#)]
23. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2018**, arXiv:1810.04805.
24. Gong, S.; Sinnott, R.O.; Qi, J.; Paris, C. Fake News Detection Through Graph-based Neural Networks: A Survey. *arXiv* **2023**, arXiv:2307.12639. [[CrossRef](#)]
25. Ma, J.; Gao, W.; Wong, K.F. Rumor detection on twitter with tree-structured recursive neural networks. In Proceedings of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; pp. 1980–1989. [[CrossRef](#)]
26. Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J. *Classification and Regression Trees*; Wadsworth International Group: Belmont, CA, USA, 1984.
27. Fix, E.; Hodges, J.L., Jr. *Discriminatory Analysis—Nonparametric Discrimination: Consistency Properties*; Technical Report No. 4; USAF School of Aviation Medicine: Randolph Field, TX, USA, 1952.
28. Vapnik, V.; Golowich, S.E.; Smola, A. Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing. *Adv. Neural Inf. Process. Syst.* **1996**, *9*, 281–287.
29. Cutler, A.; Cutler, D.R.; Stevens, J.R. Random Forests. In *Ensemble Machine Learning: Methods and Applications*; Zhang, C., Ma, Y., Eds.; Springer: Boston, MA, USA, 2012; pp. 157–175.
30. Liu, Y.; Wu, Y.F.B. Early Detection of Fake News on Social Media Through Propagation Path Classification with Recurrent and Convolutional Networks. In Proceedings of the AAAI 2018, New Orleans, LA, USA, 2–7 February 2018; AAAI Press: Palo Alto, CA, USA, 2018; pp. 354–361.
31. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyere, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
32. Asch, S.E. Effects of Group Pressure upon the Modification and Distortion of Judgments. In *Groups, Leadership and Men: Research in Human Relations*; Guetzkow, H., Ed.; Carnegie Press: Pittsburgh, PA, USA, 1951; pp. 177–190.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.