2025 International Joint Conference on Neural Networks (IJCNN) | 979-8-3315-1042-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/IJCNN64981.2025.11228022

Enhancing Fake News Detection by Incorporating Evidence Credibility

Yike Wu[1,4], Mengying Liu[2], Mingming Liu[2,*], Jiahao Sun[3], Yang Xiao[2], Mengting Hu[2]
[1]School of Journalism and Communication, Nankai University, Tianjin, China
[2]School of Software, Nankai University, Tianjin, China
[3]College of Computer Science, Nankai University, Tianjin, China
[4]Laboratory of Intelligent Publishing Technology and Standards, Nankai University, Tianjin, China
{wuyike, liumingming, mthu}@nankai.edu.cn

***Abstract*—The evidence-aware fake news detection aims to determine the veracity of claims under the guidance of external evidences. However, existing methods often neglect the credibility of evidences, making them vulnerable to misinformation in real-world scenarios where the evidence credibility is not always guaranteed. In this paper, we incorporate evidence credibility into fake news detection and propose a novel framework named ECFEND, which explicitly models the varying credibility of different evidences. Moreover, we present a new benchmark, SnopesCG, designed to simulate more realistic and challenging scenarios. Each claim in the benchmark is associated with noisy evidences retrieved from web pages as well as generated interference ones. Experimental results demonstrate the superiority of ECFEND over state-of-the-art methods, particularly on SnopesCG. We have open-sourced the code at: https://github.com/nffxdhd88/ECFEND.**

***Index Terms*—Fake News Detection, Evidence Credibility, Iterative Cross Verification**

I. Introduction

The rapid spread of fake news on the Internet poses serious threats to society, affecting politics, economics, and public trust. To solve the problem, researchers have proposed a variety of methods to automatically detect fake news in recent years. Among these, evidence-aware fake news detection [14], [18], [24] has gained prominence, which predicts the veracity of claims by leveraging external evidences retrieved from the search engine.

Despite significant progress, existing methods tend to focus on maximizing the use of evidence but often neglect its credibility. As shown in Figure 1, this leads to two major challenges: firstly, in real-world applications, the evidences for a claim are top-ranked results retrieved from a search engine without manual inspection, and the credibility is not always guaranteed. As a result, models can make incorrect predictions if they rely on incredible evidences. Secondly, deceptive individuals can easily sway the model predictions by falsifying evidences, exposing them to malicious attacks.

To address these challenges, we introduce the concept of evidence credibility, which measures how much the model should trust evidence to make the prediction, and we propose to incorporate this concept into evidence-aware fake news detection. A naive approach would be to manually label the credibility of evidences and use these labels to guide model training. However, this is both time-consuming and labor-intensive, making it impractical for large-scale real-world applications. Instead, we propose a method for modeling evidence credibility without requiring additional annotations.

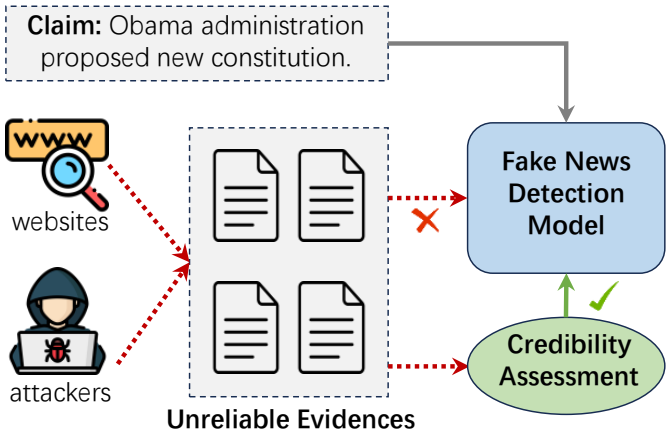


Fig. 1. A realistic scenario in evidence-aware fake news detection. Unreliable evidences, whether crawled from websites or injected by attackers, can expose detection systems to security vulnerabilities. To mitigate this, we propose assessing the credibility of evidence before making predictions. Red dashed arrows indicate unreliable processes, while the green solid arrow represents credible ones.

We hypothesize that most retrieved evidences for a claim are credible, as it is unlikely that incredible evidences would dominate top search results on the Internet. Based on this assumption, we propose a novel framework named ECFEND, which leverages a multi-head evidence credibility module to explicitly model the varying credibility of different evidences. Specifically, we perform cross verification among multiple evidences for a claim, which allows them to assess the credibility of each other. To enhance reliability, we iteratively duplicate this process to ensure robust credibility assessments. The obtained credibility scores are then used to modulate the attention weights of each evidence to refine the final prediction.

Moreover, we argue that existing datasets, such as Snopes

*Corresponding Author.

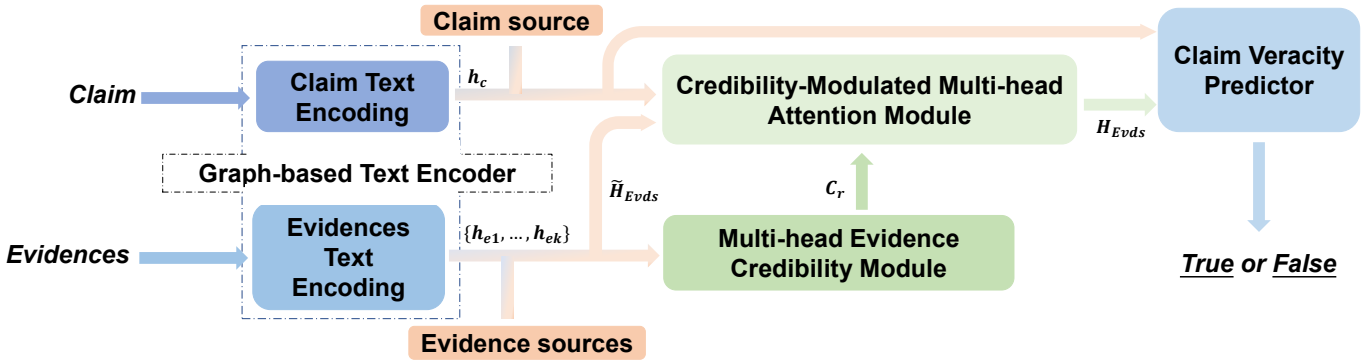


Fig. 2. The overview of the proposed ECFEND framework.

and PolitiFact [14], are excessively cleaned and fail to capture the complexity of real-world scenarios, which leaves models trained on them poorly equipped to handle noisy or malicious data. To overcome this limitation, we present a new benchmark SnopesCG to simulate more realistic conditions. SnopesCG differs from previous datasets in two ways: first, it introduces more noise by applying a more relaxed cleaning process to the crawled evidences; second, it incorporates generated interference evidences to simulate malicious attacks. The construction and specifics of SnopesCG are detailed in Section IV.

To sum up, the main contributions of this paper are as follows:

- We propose ECFEND, a novel framework that models the evidence credibility without labeled credibility data.
- We introduce SnopesCG, a new benchmark that simulates more realistic and challenging scenarios, fostering further research in evidence-aware fake news detection.
- Extensive experiments on Snopes, PolitiFact, and SnopesCG demonstrate the superiority of ECFEND over state-of-the-art methods, particularly in handling real-world complexities.

II. Related Work

A. Evidence-aware Fake News Detection

Fake news detection models can be broadly categorized into two types: those based solely on news content and those that incorporate auxiliary information such as user behavior, propagation patterns, or external evidence. The former group typically extracts linguistic features to determine veracity [2], [7], [9], [15]. However, a significant focus has been placed on evidence-aware approaches, where models leverage external evidence to verify claims.

In evidence-aware fake news detection, retrieved external evidence is used to establish the truthfulness of claims. DeClarE [14] is a prominent example, employing Bi-LSTM and attention mechanisms to align claims with their corresponding evidence. HAN [13] takes this further by using dual attention mechanisms to ensure topic coherence between claims and evidence, while other approaches focus on hierarchical relationships [18], [22]. To handle redundant information in retrieved evidence, GET [24] introduces text graphs and Gated Graph Neural Networks to improve evidence selection and processing.

Despite these advancements, most existing methods neglect the critical aspect of evidence credibility. They often assume that all retrieved evidence is trustworthy, without explicitly modeling its credibility. While some studies have examined user credibility or the reliability of news sources [11], [20], they do not directly evaluating the credibility of the evidence itself.

Our work bridges this gap by introducing the concept of evidence credibility into fake news detection. We propose a multi-head evidence credibility module that evaluates the trustworthiness of evidence without requiring additional labels. This allows the model to effectively filter out unreliable evidence, significantly enhancing robustness and resilience to noisy real-world conditions.

B. Evidence Attack

Evidence attacks, aimed at undermining evidence-aware models, have received increasing attention. Zellers et al. [25] introduced Grover, a controllable text generation model capable of producing highly convincing fake news articles. Du et al. [5] further demonstrated that injecting or modifying evidence in the document pool can significantly degrade the performance of fact verification models. Abdelnabi and Fritz [1] systematically explored evidence manipulation attacks, categorizing them based on attacker goals, constraints, and capabilities. Additionally, Huang et al. [8] developed a method to generate deceptive articles that mimic human writing styles, posing further challenges to detection systems.

In this paper, we introduce SnopesCG, a new benchmark designed to better mirror real-world complexities. SnopesCG incorporates both noisy real-world evidence and generated attacking evidence to simulate more realistic and challenging scenarios. Our proposed approach, which integrates evidence credibility, demonstrates robustness against noise and malicious attacks, achieving superior performance on this benchmark.

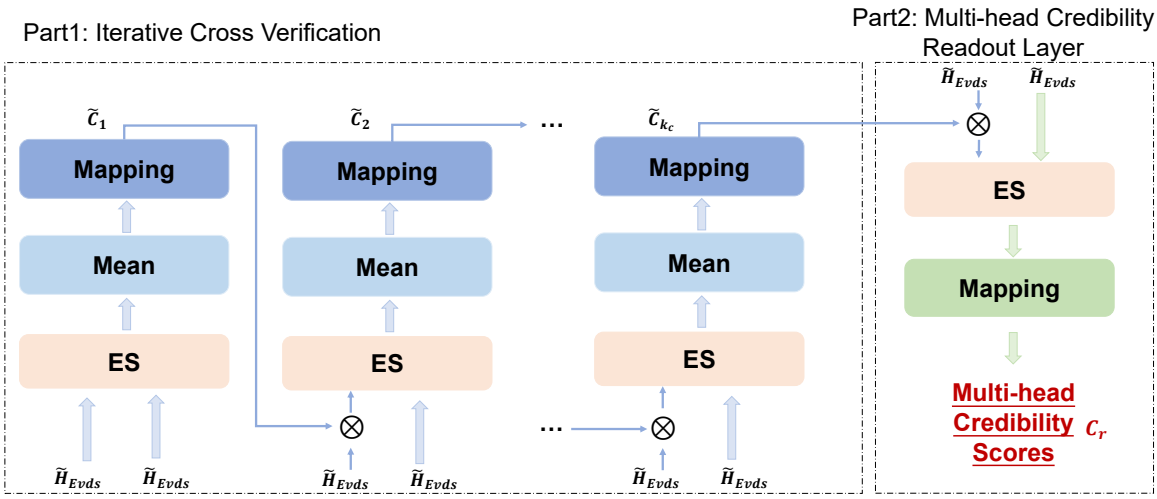


Fig. 3. The architecture of Multi-head Evidence Credibility Module. "ES" denotes a scaled dot-product, "Mean" represents a mean pooling operation, and "Mapping" indicates a linear layer with exponential and tanh activation functions. Note that the two "Mapping" operations in different colors have distinct parameters.

III. ECFEND Framework

A. Overview

Given a claim c from source s_c, along with its evidence set Evds, the evidence-aware fake news detection model $\mathcal{F}$ predicts the veracity of the claim as follows: $p_r, p_f = \mathcal{F}(c, s_c, \text{Evds})$, where p_r and p_f represent the probabilities of the claim being true or false, respectively. The evidence set $\text{Evds} = \{(e_1, s_{e1}), (e_2, s_{e2}), \ldots, (e_k, s_{ek})\}$ consists of k evidence items, each paired with its source s_e.

The ECFEND framework, illustrated in Figure 2, consists of four main components: a graph-based text encoder, a multi-head evidence credibility module, a credibility-modulated multi-head attention module, and a claim veracity predictor. First, the claim and its associated evidence are encoded by the graph-based text encoder, producing the claim representation $\boldsymbol{h}_c$ and each evidence representation $\boldsymbol{h}_e$. Then, the evidence representations along with their sources, forming the matrix $\tilde{\boldsymbol{H}}_{\text{Evds}}$, are processed by the multi-head evidence credibility module to generate credibility scores $\boldsymbol{C}_r$. Next, these scores are then used by the credibility-modulated multi-head attention module to adjust the attention weights across evidences. The adjusted attention weights refine $\tilde{\boldsymbol{H}}_{\text{Evds}}$, yielding the final evidence representation $\boldsymbol{H}_{\text{Evds}}$. Finally, the claim representation $\boldsymbol{h}_c$, the refined evidence $\boldsymbol{H}_{\text{Evds}}$, and the claim source are passed to the claim veracity predictor, which outputs the probabilities p_r and p_f, determining the veracity of the claim.

B. Graph-based Text Encoder

Given a claim and its associated evidences, we encode them into document-level representations using a graph-based text encoder. Each word in the document is represented as a node in a graph, with edges established between the central word and others within a sliding window of size 5. The resulting claim graph and evidence graphs are then passed through a two-layer Gated Graph Neural Network (GGNN) [12], followed by mean pooling to produce the document-level representations $\boldsymbol{h}_c$ and $\boldsymbol{h}_e$. Note that for evidence graphs, a new adjacency matrix is constructed in the second GGNN layer to remove redundant nodes, initially mitigating the inherent noise and redundancy in crawled evidence. The implementation of this encoder follows the prior work [24], please refer to it for details.

C. Multi-head Evidence Credibility Module

This module comprises two key components: iterative cross-verification and a multi-head credibility readout layer, as shown in Figure 3. It takes the encoded evidences $\tilde{\boldsymbol{H}}_{\text{Evds}} = \{[\boldsymbol{h}_{e1}; \boldsymbol{s}_{e1}], \ldots, [\boldsymbol{h}_{ek}; \boldsymbol{s}_{ek}]\}$ as input, where each evidence is concatenated with its source vector. The output is a set of credibility scores $\boldsymbol{C}_r \in \mathbb{R}^{k \times h}$, where h is the number of heads in the readout layer.

1) Iterative Cross Verification: To estimate credibility scores without explicit labels, we hypothesize that most evidence for a claim is credible. The credibility of each evidence increases when it receives substantial support from others. Consequently, we use the cross verification among evidences to assess their credibility. However, a single pass of cross-verification may be unreliable due to noise and uncertainties in the initial assessment. Therefore, we refine this process through iterative cross-verification, ensuring a more robust evaluation of evidence credibility.

In each iteration t, pairwise relationships among the evidence in $\tilde{\boldsymbol{H}}_{\text{Evds}}$ are computed using a scaled dot-product operation denoted as ES, followed by mean pooling to derive scalar values for each evidence. These scalars are mapped into credibility scores $\tilde{\boldsymbol{C}}_t \in \mathbb{R}^k$ using a linear layer with exponential and $\tanh$ activations. The credibility scores guide the next iteration, and the process is repeated for k_c iterations:

$$\tilde{\boldsymbol{M}}_t = \text{ES}\left(\tilde{\boldsymbol{H}}_{\text{Evds}}, \tilde{\boldsymbol{C}}_{t-1}\tilde{\boldsymbol{H}}_{\text{Evds}}\right), \quad (1)$$

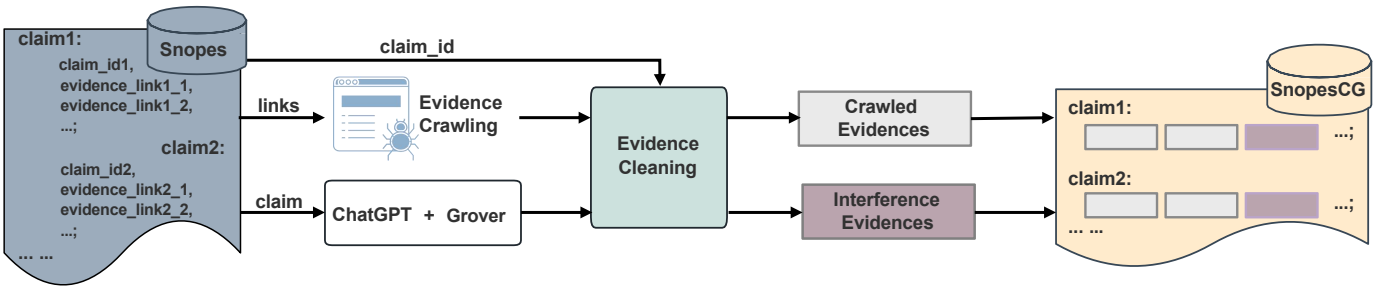


Fig. 4. The overall construction process of SnopesCG.

$$\tilde{\boldsymbol{C}}_t = \tanh\left(\exp\left(\text{Mean}\left(\tilde{\boldsymbol{M}}_t\right)\boldsymbol{W}_t\right)\right), \tag{2}$$

where $\text{ES}(\boldsymbol{Q},\boldsymbol{K}) = a * \frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d_k}}$, with d_k representing the dimension of $\boldsymbol{Q}$ and $\boldsymbol{K}$, and a and $\boldsymbol{W}_t \in \mathbb{R}^{k\times k}$ are learnable parameters. Initial credibility scores $\tilde{\boldsymbol{C}}_0$ are set to $\mathbf{1} \in \mathbb{R}^k$.

2) Multi-head Credibility Readout Layer: This layer is designed to determine the final credibility scores $\boldsymbol{C}_r \in \mathbb{R}^{k\times h}$ after undergoing k_c iterations of cross verification. To provide a multi-perspective evaluation, we generate h credibility scores for each evidence. This process mirrors the previous cross-verification step, followed by a linear mapping:

$$\tilde{\boldsymbol{M}}_r = \text{ES}\left(\tilde{\boldsymbol{H}}_{\text{Evds}}, \tilde{\boldsymbol{C}}_{k_c}\tilde{\boldsymbol{H}}_{\text{Evds}}\right), \tag{3}$$

$$\boldsymbol{C}_r = \tanh\left(\exp\left(\tilde{\boldsymbol{M}}_r\boldsymbol{W}_r\right)\right), \tag{4}$$

where $\boldsymbol{W}_r \in \mathbb{R}^{k\times h}$ is a learnable parameter.

D. Credibility-Modulated Multi-head Attention Module

Previous work usually focus on utilizing attention mechanisms to assess evidence importance, but they overlook the credibility. To address this limitation, we incorporate credibility scores $\boldsymbol{C}_r$ into the attention process.

First, multi-head attention weights $\boldsymbol{O}$ between the claim and evidence are computed as follows:

$$\boldsymbol{O} = \text{softmax}\left(\left(\tanh\left(\left[\tilde{\boldsymbol{H}}_{\text{Evds}}; \boldsymbol{H}_c\right]\boldsymbol{W}_{o1}\right)\right)\boldsymbol{W}_{o2}\right), \tag{5}$$

where $\boldsymbol{H}_c$ is derived from the claim representation along with its source $[\boldsymbol{h}_c; \boldsymbol{s}_c]$, repeated k times. Both $\boldsymbol{W}_{o1}$ and $\boldsymbol{W}_{o2}$ are learnable parameters.

Second, we leverage the credibility scores $\boldsymbol{C}_r$ to modulate these attention weights via a Gated Linear Unit (GLU) [3]. The modulated attention weights are then used to refine each evidence representation in $\tilde{\boldsymbol{H}}_{\text{Evds}}$, resulting in the final evidence representations $\boldsymbol{H}_{\text{Evds}}$. The details are outlined as follows:

$$\boldsymbol{H}_{\text{Evds}} = \text{flatten}\left(\tilde{\boldsymbol{H}}^{\text{T}}_{\text{Evds}}\text{GLU}\left(\boldsymbol{C}_r, \boldsymbol{O}\right)\right), \tag{6}$$

$$\text{GLU}\left([\boldsymbol{A}; \boldsymbol{B}]\right) = \text{gating}\left([\boldsymbol{A}; \boldsymbol{B}]\,\boldsymbol{W}_g + \boldsymbol{b}_g\right), \tag{7}$$

$$\text{gating}\left(\boldsymbol{X}\right) = \boldsymbol{X}_{[:,:h]} \odot \sigma\left(\boldsymbol{X}_{[:,h:]}\right), \tag{8}$$

where flatten denotes an operation that concatenates the outputs from the multi-head attention. The parameters $\boldsymbol{W}_g \in \mathbb{R}^{2h\times 2h}$ and $\boldsymbol{b}_g \in \mathbb{R}^{2h}$ are learnable, and the notation $\boldsymbol{X}_{[:,:h]}$ and $\boldsymbol{X}_{[:,h:]}$ represents the column-wise splitting of $\boldsymbol{X}$ into two parts of equal dimension. Additionally, the function σ denotes the sigmoid function.

E. Claim Veracity Prediction

Finally, using the claim representation $\boldsymbol{h}_c$, source vector $\boldsymbol{s}_c$, and refined evidence representation $\boldsymbol{H}_{\text{Evds}}$, we predict claim veracity via a linear layer followed by softmax:

$$p_r, p_f = \text{softmax}\left(\left[\boldsymbol{H}_{\text{Evds}}; \boldsymbol{h}_c; \boldsymbol{s}_c\right]\boldsymbol{W}_p + \boldsymbol{b}_p\right), \tag{9}$$

$$\text{label} = \text{argmax}(p_r, p_f). \tag{10}$$

IV. SnopesCG Benchmark

Snopes and PolitiFact are among the most widely used datasets for evidence-aware fake news detection. However, both datasets undergo extensive cleaning, resulting in evidence that does not accurately reflect real-world conditions. Specifically, claims are used as queries to retrieve top-ranked articles from search engines, and a 100-word snippet with the highest relevance is extracted as evidence. Less relevant evidence is excluded based on predefined relevance thresholds, leading to a dataset with highly pertinent and relatively noise-free evidence. While this approach simplifies training, it fails to capture the complexity and noise inherent in real-world misinformation, where evidence is often inconsistent, diverse, or intentionally misleading. Moreover, these datasets do not account for the risk of generating false narratives to deceive models.

To address these limitations, we introduce SnopesCG, a benchmark crafted to more accurately reflect real-world challenges. As shown in Figure 4, SnopesCG is built on the foundation of Snopes, retaining its core structure while introducing two key innovations. First, a more relaxed data cleaning process is employed, allowing for the inclusion of noise in the retrieved evidence. Second, adversarial interference evidence is generated to simulate potential attacks, significantly enhancing the realism and complexity of the benchmark.

A. Evidence Crawling and Cleaning

Unlike the Snopes dataset, we adopt a more lenient approach to cleaning the retrieved articles. After retrieving top-ranked articles based on the provided evidence links in Snopes, we split the *claim_id* attribute into keywords (e.g., *claim_id*

"1998-trump-people-quote" is split into ['1998', 'trump', 'people', 'quote']). These keywords are individually searched within the crawled articles, and once located, we extract a 100-word snippet centered around the first occurrence (10 words to the left and 90 words to the right). If no keywords are found, we extract the first 100 words of the article. This process results in noisier evidence that more accurately reflects the variability and inconsistency of real-world data.

B. Interference Evidence

To simulate attack scenarios, we generate interference evidence using the API of ChatGPT and Grover [5]. ChatGPT first generates five interference sentences for each claim, with opposing semantics for true claims and similar semantics for false claims. For true claims, the interference evidence is generated using the following prompt:

> *Produce 5 A for opposite meaning sentences generation of Q.*
> *Q: {the content of a claim}*
> *A: <opposite-sentences>*

For false claims, the interference evidence is generated using the following prompt:

> *Produce 5 A for similar meaning sentences generation of Q.*
> *Q: {the content of a claim}*
> *A: <similar-sentences>*

However, since ChatGPT tends to generate concise and coherent sentences, we use Grover to expand these sentences into more verbose paragraphs, better mimicking the noise and verbosity of real-world articles. Following the prior work [5], we specify the domain as *wikipedia.com* in Grover's input to produce more realistic content. The expanded paragraphs are then processed using the same cleaning method as the retrieved articles to extract evidence snippets.

Interference evidence is only added to claims with more than five pieces of crawled evidence, ensuring that interference does not dominate the evidence pool. To maintain consistency with existing datasets, we limit the total number of evidence per claim to 30.

C. Comparison with Snopes and PolitiFact

Table I compares the SnopesCG, Snopes, and PolitiFact datasets. SnopesCG offers the most comprehensive coverage, containing 62,181 evidence items—more than double the amount in Snopes and PolitiFact. This is partly due to the more relaxed data cleaning process in SnopesCG, resulting in an average of 10.2 crawled evidence items per claim, compared to 6.7 in Snopes and 8.3 in PolitiFact. Moreover, SnopesCG is the only dataset that incorporates generated interference evidence, with an average of 3.9 per claim. Overall, SnopesCG provides a more robust and challenging benchmark that better reflects real-world conditions. Its richer and noisier evidence enhances the dataset's utility for evaluating model resilience against both noise and malicious attacks.

TABLE I
COMPARISON OF SNOPES, POLITIFACT, AND SNOPESCG DATASETS. "CLAIMS" REPRESENTS THE TOTAL NUMBER OF CLAIMS, WHILE "EVDS" REFERS TO THE TOTAL EVIDENCE. "CRAWLED" INDICATES THE AVERAGE NUMBER OF RETRIEVED EVIDENCE PER CLAIM, AND "GENERATED" REFLECTS THE AVERAGE NUMBER OF INTERFERENCE EVIDENCE PER CLAIM.

Dataset	Claims	Evds	Evds per Claim	
			Crawled	Generated
Snopes	4341	27827	6.7	0
PolitiFact	3568	28439	8.3	0
SnopesCG	4476	62181	10.2	3.9

TABLE II
HYPERPARAMETER SETTINGS IN EXPERIMENTS. **1**, **2** AND **3** STAND FOR TRAINING-RELATED HYPERPARAMETERS, INPUT-RELATED ONES, AND ARCHITECTURE-RELATED ONES RESPECTIVELY.

Hyperparameters		Snopes	PolitiFact	SnopesCG
1	Learning Rate	$1 \times e^{-4}$	$2.5 \times e^{-5}$	$1 \times e^{-4}$
	Batch Size	32	16	32
	Epochs	100		
	Optimizer	Adam		
2	k	30		
	d	300		
3	h	2	1	2
	k_c	4	5	5

V. EXPERIMENTS

A. Experimental Setup

Datasets. We conduct our experiments on two widely used datasets for evidence-aware fake news detection: Snopes and PolitiFact [14]. To evaluate the robustness of our approach under noisy and adversarial conditions, we also use the newly constructed SnopesCG dataset.

Evaluation. Consistent with previous studies [14], [18], [24], we perform five-fold cross-validation and report the average results across the folds. Our evaluation metrics include macro F1 (F1-Ma), micro F1 (F1-Mi), F1 score for the false category (F1-False), and F1 score for the true category (F1-True).

Implementation Details. The experiments are implemented using python 3.7, Pytorch 1.12, and CUDA 11.2 on an NVIDIA A6000 with 48G of memory, operating on Ubuntu 18.04. For the Snopes dataset, we set the head number h of credibility scores to 2, with k_c (the iteration count for cross-verification) set to 4. The Adam optimizer [10] is used with a learning rate of 1×10^{-4} and a batch size of 32. Training is conducted for up to 100 epochs, with early stopping applied if the F1-macro score did not improve over 10 consecutive epochs. Detailed hyperparameter configurations for Snopes, PolitiFact, and SnopesCG are provided in Table II.

B. Baselines

We compare ECFEND with three groups of baselines: (1) *Content-based methods*, which rely solely on the claim content

TABLE III
EXPERIMENT RESULTS (%) ON POLITIFACT AND SNOPES. † DENOTES OUR IMPLEMENTATION.

Model	PolitiFact				Snopes			
	F1-Ma	F1-Mi	F1-False	F1-True	F1-Ma	F1-Mi	F1-False	F1-True
LSTM [16]	60.6	60.9	59.3	61.8	62.1	71.9	81.2	43.0
TextCNN [19]	60.4	60.7	59.2	61.5	63.1	72.0	81.2	45.0
BERT [4]	59.7	59.8	58.6	60.8	62.1	71.6	81.0	43.1
DeClarE [14]	65.3	65.2	63.1	67.5	72.5	78.6	85.7	59.4
HAN [13]	66.1	66.0	64.3	67.9	75.2	80.2	86.8	63.6
EHIAN [22]	67.6	67.9	65.5	68.9	78.4	82.8	88.5	68.4
CICD [21]	68.2	68.5	65.7	70.2	78.9	83.7	89.3	69.1
MAC† [18]	67.9	68.3	64.8	71.1	78.5	83.4	88.7	68.2
GET† [24]	68.3	68.6	65.2	71.4	80.1	84.5	89.5	70.6
RobustSP [23]	68.0	68.3	65.3	70.6	77.2	81.8	87.5	66.9
LLaMA2-7B [17]	52.8	53.1	56.7	48.9	56.6	66.6	77.4	35.7
LLaMA3-13B-instruct [6]	63.8	63.9	65.7	61.9	70.0	80.6	87.7	52.2
ECFEND (Ours)	**69.1**	**69.3**	**66.2**	**71.9**	**80.5**	**84.7**	**89.6**	**71.5**

TABLE IV
EXPERIMENT RESULTS (%) ON SNOPESCG.

Model	F1-Ma	F1-Mi	F1-False	F1-True
MAC [18]	64.9	71.4	80.0	49.8
GET [24]	66.2	73.4	81.8	50.7
LLaMA2-7B [17]	51.2	57.0	68.1	34.3
LLaMA3-13B-instruct [6]	68.0	76.9	84.9	51.1
ECFEND (Ours)	67.1	75.1	83.3	51.0

for predictions. This includes models such as LSTM [16], TextCNN [19], and BERT [4]; (2) *Evidence-aware methods*, which, like our approach, incorporate external evidence. These include DeClarE [14], HAN [13], EHIAN [22], CICD [21], MAC [18], GET [24], and RobustSP [23]; (3) To provide a more comprehensive comparison, we also evaluate large language models (LLMs) such as LLaMA2-7B [17] and LLaMA3-13B-instruct [6] . The prompt used to determine claim veracity for both models is as follows:

> *News Content:*
> {*the content of a claim*}
> *Evidences:*
> {*evidences*}
> *Verdict: Is the news item true or false?*
> *[Options]:*
> *True*
> *False*
> *Answer:*

In this case, {*the content of a claim*} is filled with the claim itself, while {*evidences*} is filled with the merged content of the associated evidences, with a maximum token limit of 2048. After inputting the prompt into the LLM, we determine the veracity of the claim based on whether the output contains "True" or "False."

C. Comparison Results

The comparison results for PolitiFact and Snopes are presented in Table III. First, we observe that incorporating external evidence significantly improves performance over content-based models, with ECFEND consistently outperforming all baselines on both datasets. For instance, on PolitiFact, ECFEND surpasses MAC by 1.2% in F1-Ma and 1.0% in F1-Mi, and outperforms GET by 0.8% and 0.7% in the same metrics. These results highlight the value of modeling evidence credibility for enhancing fake news detection.

Notably, LLaMA2 and LLaMA3 do not perform as well as smaller models in this task. This could be because the datasets do not fully capture the complexities where LLMs typically excel, suggesting that more focused models like ECFEND may be better suited for tasks with limited data and more structured scenarios.

To assess performance in a more challenging, real-world scenario, we evaluate ECFEND on SnopesCG, comparing it with competitive baselines such as MAC and GET. As shown in Table IV, ECFEND consistently outperforms both baselines, with gains of 2.2% in F1-Ma and 3.7% in F1-Mi over MAC, and improvements of 0.9% and 1.7% over GET. These larger performance gains on SnopesCG, compared to Snopes, indicate that SnopesCG effectively simulates more complex and realistic environments. Additionally, LLaMA3 demonstrates strong performance on SnopesCG, suggesting that it excels in handling noisy and adversarial conditions. However, ECFEND remains competitive, delivering comparable performance with a more lightweight and cost-effective model, making it easier for real-world deployment.

D. Case Study

Figure 5 presents a case study using samples from the Snopes test set to qualitatively evaluate the interpretability of the credibility scores generated by the Multi-head Evidence Credibility (MEC) module in ECFEND. Each claim is shown alongside its ground-truth label and a subset of associated evidence, allowing us to assess how well the credibility scores reflect the reliability of each evidence.

Our analysis reveals that the credibility scores align well with the actual reliability of the evidence. First, in the case of Claim 1, which is false, Evd1 supports the claim, while Evd2 refutes it. For a false claim, evidence that refutes it should be more credible than evidence that supports it. This is precisely

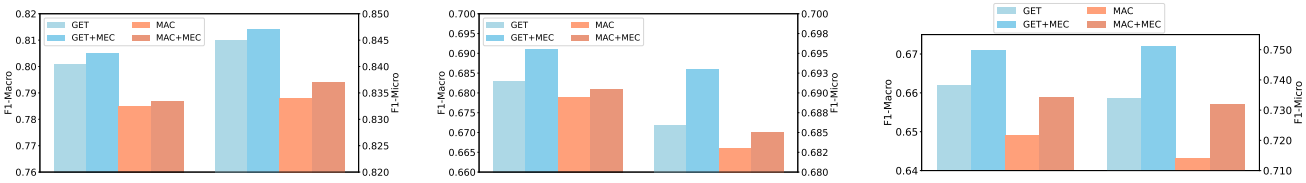
Claim1: obama administration proposed new constitution **[False]**

Evd1: …obama administration proposes new constitution proving it is a total … **(0.67)**
Evd2: …i don't think obama wants to write a new constitution i think he would rather replace the constitution with the communist … **(1.00)**

Claim2: nascar champion tony stewart hit killed fellow driver accident race trac **[True]**

Evd1: … tony stewart nascar sprint cup champion hit and killed fellow driver kevin … **(0.92)**
Evd2: … nascar champion tony stewarts car struck his fellow racer …**(0.94)**

Fig. 5. Case study on the interpretability of credibility scores. Each claim is followed by its ground-truth label in square brackets. "Evd" represents the associated evidence, and the value in parentheses indicates its credibility score. For simplicity, only the first-head score from the multi-head credibility readout layer is displayed.



(a) Snopes (b) PolitiFact (c) SnopesCG

Fig. 6. Performance comparison between the original models (MAC and GET) and those enhanced with the MEC module (MAC+MEC and GET+MEC) on Snopes, PolitiFact, and SnopesCG. For each dataset, the bars on the left side represents the F1-Macro score, and the bars on the right side represents the F1-Micro score.

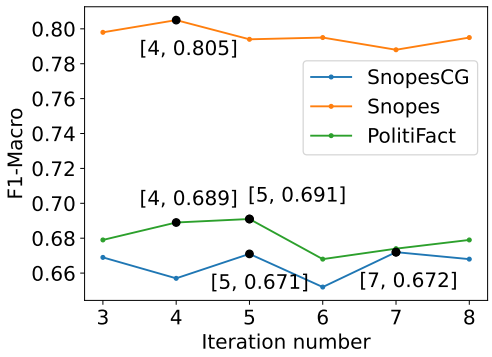


Fig. 7. Impact of iteration numbers on ECFEND's performance.

reflected in the credibility scores: Evd2 receives a perfect score of 1.00, significantly higher than Evd1's score of 0.67.

Similarly, in the case of Claim 2, which is true, both Evd1 and Evd2 support the claim. In this scenario, supporting evidence is expected to have high credibility, and the credibility scores confirm this, with Evd1 and Evd2 receiving high and nearly identical scores of 0.92 and 0.94, respectively.

Overall, this case study demonstrates that the evidence credibility scores produced by ECFEND are highly interpretable. The MEC module effectively distinguishes between more and less credible evidence, allowing the model to make accurate veracity predictions in line with human reasoning.

E. Model-Agnostic Analysis

To investigate the adaptability of the Multi-head Evidence Credibility (MEC) module, we integrate it into two baseline models, MAC and GET, and evaluate their performance on Snopes, PolitiFact, and SnopesCG. Figure 6 shows the F1-Macro and F1-Micro scores for both models with and without the MEC module across all three datasets. The results indicate consistent performance improvements with the addition of the MEC module, with the most notable gains observed on the SnopesCG dataset, which features noisier evidence.

These results demonstrate that the MEC module enhances evidence credibility assessment across different model architectures, suggesting its potential for broader application in other evidence-aware fake news detection models.

F. Effect of Iteration Number in MEC

We examine the impact of varying the iteration number k_c in the MEC module on ECFEND's performance. Figure 7 shows how changing k_c from 3 to 8 affects results across the Snopes, PolitiFact, and SnopesCG datasets. The results indicate that the optimal k_c falls between 4 and 7, depending on the dataset.

ECFEND achieves its best performance with k_c set to 4 on Snopes and 5 on PolitiFact. For SnopesCG, which contains noisier evidence, the model performs optimally with $k_c = 7$. This highlights the need for more cross-verification iterations on datasets with higher levels of noise.

VI. Conclusion

In this paper, we propose a model-agnostic framework, ECFEND, that incorporates evidence credibility into fake news detection. Specifically, we design a multi-head evidence

credibility module to derive the credibility score of each piece of evidence through iterative cross-verification. This credibility score is then used to adjust the attention weights assigned to each evidence. Additionally, we present a new dataset, SnopesCG, which simulates more realistic and challenging scenarios involving less credible evidence. Experimental results demonstrate that ECFEND significantly improves detection performance, particularly in noisy environments, and achieves comparable results to the latest large language model LLaMA3, while being more lightweight and cost-efficient. We hope this work encourages further research into modeling evidence credibility to strengthen fake news detection systems.

ACKNOWLEDGEMENTS

This research is supported by the Ministry of Education of the People's Republic of China Humanities and Social Sciences Youth Foundation (Grant No. 23YJCZH240), and the National Natural Science Foundation of China (Grant No. 62302245, 62406151, 62406150).

REFERENCES

[1] Sahar Abdelnabi and Mario Fritz. Fact-saboteurs: A taxonomy of evidence manipulation attacks against fact-verification systems. *ArXiv*, abs/2209.03755, 2022.

[2] Mingxi Cheng, Shahin Nazarian, and Paul Bogdan. Vroc: Variational autoencoder-aided multi-task rumor classifier based on text. In *Proceedings of The Web Conference 2020*, WWW '20, page 2892–2898, New York, NY, USA, 2020. Association for Computing Machinery.

[3] Yann Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *International Conference on Machine Learning*, 2016.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2019.

[5] Y. Du, Antoine Bosselut, and Christopher D. Manning. Synthetic disinformation attacks on automated fact verification systems. In *AAAI Conference on Artificial Intelligence*, 2022.

[6] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[7] Bilal Ghanem, Simone Paolo Ponzetto, Paolo Rosso, and Francisco Rangel. FakeFlow: Fake news detection by modeling the flow of affective information. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 679–689, Online, April 2021. Association for Computational Linguistics.

[8] Kung-Hsiang Huang, Preslav Nakov, Yejin Choi, and Heng Ji. Faking fake news for real fake news detection: Propaganda-loaded training data generation. In *Annual Meeting of the Association for Computational Linguistics*, 2022.

[9] Gihwan Kim and Youngjoong Ko. Graph-based fake news detection using a summarization technique. In *Conference of the European Chapter of the Association for Computational Linguistics*, 2021.

[10] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[11] Quanzhi Li, Qiong Zhang, and Luo Si. Rumor detection by exploiting user credibility information, attention and multi-task learning. In *Annual Meeting of the Association for Computational Linguistics*, 2019.

[12] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard S. Zemel. Gated graph sequence neural networks. *CoRR*, abs/1511.05493, 2015.

[13] Jing Ma, Wei Gao, Shafiq R. Joty, and Kam-Fai Wong. Sentence-level evidence embedding for claim verification with hierarchical attention networks. In *Annual Meeting of the Association for Computational Linguistics*, 2019.

[14] Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. Declare: Debunking fake news and false claims using evidence-aware deep learning, 2018.

[15] Piotr Przybyła. Capturing the style of fake news. In *AAAI Conference on Artificial Intelligence*, 2020.

[16] Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Conference on Empirical Methods in Natural Language Processing*, 2017.

[17] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[18] Nguyen Vo and Kyumin Lee. Hierarchical multi-head attentive network for evidence-aware fake news detection. *ArXiv*, abs/2102.02680, 2021.

[19] William Yang Wang. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Annual Meeting of the Association for Computational Linguistics*, 2017.

[20] Lingwei Wei, Dou Hu, Wei Zhou, Zhaojuan Yue, and Songlin Hu. Towards propagation uncertainty: Edge-enhanced bayesian graph convolutional networks for rumor detection. *ArXiv*, abs/2107.11934, 2021.

[21] Lianwei Wu, Yuan Rao, Yuqian Lan, Ling Sun, and Zhao Qi. Unified dual-view cognitive model for interpretable claim verification. In *Annual Meeting of the Association for Computational Linguistics*, 2021.

[22] Lianwei Wu, Yuan Rao, Xiong Yang, Wanzhen Wang, and Ambreen Nazir. Evidence-aware hierarchical interactive attention networks for explainable claim verification. In *International Joint Conference on Artificial Intelligence*, 2020.

[23] Yike Wu, Yang Xiao, Mengting Hu, Mengying Liu, Pengcheng Wang, and Mingming Liu. Towards robust evidence-aware fake news detection via improving semantic perception. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16607–16618, 2024.

[24] Weizhi Xu, Jun Wu, Qiang Liu, Shu Wu, and Liang Wang. Evidence-aware fake news detection with graph neural networks. *Proceedings of the ACM Web Conference 2022*, 2022.

[25] Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. Defending against neural fake news. In *Neural Information Processing Systems*, 2019.