Towards Robust Evidence-Aware Fake News Detection via Improving Semantic Perception

Yike Wu[1,2], Yang Xiao[3], Mengting Hu[3*], Mengying Liu[3], Pengcheng Wang[3], Mingming Liu[3]
[1] School of Journalism and Communication, Nankai University, Tianjin, China
[2] Convergence Media Research Center, Nankai University, Tianjin, China
[3] School of Software, Nankai University, Tianjin, China
{wuyike, mthu, liumingming}@nankai.edu.cn

Abstract

Evidence-aware fake news detection aims to determine the veracity of a given news (i.e., claim) with external evidences. We find that existing methods lack sufficient semantic perception and are easily blinded by textual expressions. For example, they still make the same prediction after we flip the semantics of a claim, which makes them vulnerable to malicious attacks. In this paper, we propose a model-agnostic training framework to improve the semantic perception of evidence-aware fake news detection. Specifically, we first introduce two kinds of data augmentation to complement the original training set with synthetic data. The semantic-flipped augmentation synthesizes claims with similar textual expressions but opposite semantics, while the semantic-invariant augmentation synthesizes claims with the same semantics but different writing styles. Moreover, we design a novel module to learn better claim representation which is more sensitive to the semantics, and further incorporate it into a multi-objective optimization paradigm. In the experiments, we also extend the original test set of benchmark datasets with the synthetic data to better evaluate the model perception of semantics. Experimental results demonstrate that our approach significantly outperforms the state-of-the-art methods on the extended test set, while achieving competitive performance on the original one. Our source code are released at https://github.com/Xyang1998/RobustFND.

Keywords: Evidence-Aware Fake News Detection, Robustness, Contrastive Learning

1. Introduction

With the rapid development of the Internet, fake news is easily fabricated and rapidly propagated through online media, which manipulates public opinion and threatens the security of networks and society (Allcott and Gentzkow, 2017; Naeem and Bhatti, 2020). In recent years, some researchers are committed to automatically determining the veracity of a given claim with the guidance of external evidences (Popat et al., 2018; Ma et al., 2019; Vo and Lee, 2021; Xu et al., 2022), i.e., evidence-aware fake news detection, and achieve continuous improvement on benchmark datasets, such as Snopes and PolitiFact (Popat et al., 2018) .

Despite the considerable progress in evidence-aware fake news detection, we observe that existing methods still lack sufficient awareness of semantics, which makes them vulnerable to malicious attacks. To provide a more intuitive illustration, we conduct a case study using GET (Xu et al., 2022). As shown in Figure 1, we simply add negation to the original claim "Mccain supports repeal death tax" to generate a new claim, leaving everything else unchanged. The semantics of two claims are diametrically opposite, and the evidences that support the original one definitely disagree with the generated one. However, the detection model ignores the semantic flip and gives the same output for two claims. To further analyze the model perception of semantics in a quantitative manner, we also extend the original test set of the benchmarks Snopes and PolitiFact via two kinds of data augmentation introduced in this paper. We find that the detection performance of state-of-the-art methods significantly decrease on this extended test set. This pilot experiment is elaborated in Section 3.

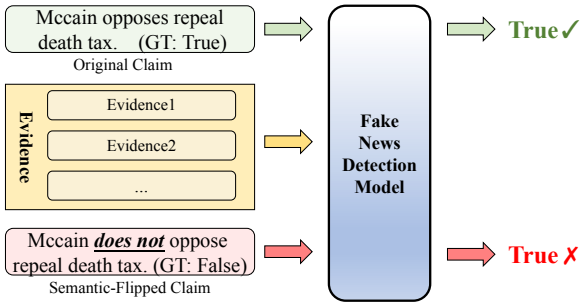

Figure 1: An example on the lack of sufficient semantic perception. We flip the semantics of the original claim as another one, but the model still gives the same output and makes wrong prediction.

We argue that there are mainly two reasons for this unsatisfactory behavior. First, in the original datasets collected from a single data source, the

*Corresponding author.

LREC-COLING 2024, pages 16607–16618

statements are usually quite different from each other in terms of the semantics and textual expressions. During the training process on such datasets, the model can still make correct predictions even without fully capturing the semantic details, which hinders the further improvement of the model's semantic perception ability. Second, the claim representation for final prediction is not sensitive to semantics. Existing methods usually only apply a cross-entropy loss for model training, which cannot explicitly ensure that in the embedding space the claims with different semantics but similar textual expressions are completely separated, and the ones with the same semantics but different writing styles are fully aggregated.

In this paper, we propose a model-agnostic training framework to improve the model semantic perception for robust evidence-aware fake news detection. Specifically, we introduce two kinds of data augmentation to complement the original dataset with our synthetic data. Given an instance in the original dataset, the ***semantic-flipped augmentation*** flips its semantics by just adding negation to it, while the ***semantic-invariant augmentation*** keeps its semantics unchanged and rewrite it in another style. Inspired by the recent advances in LLMs, we also partially employ ChatGPT to implement both two kinds of augmentation. By adding the synthetic data into the training process, we force the model to concentrate on the semantics of a claim behind its superficial textual expression.

Moreover, we design the ***semantic-sensitive claim representation learning module*** to explicitly encourage the separation of claims with different semantics and the closeness of claims with the same semantics in the embedding space. Firstly, we treat each instance in the original training set as an anchor sample, and perform the semantic-flipped and semantic-invariant augmentation on it to obtain its positive and negative counterparts respectively. Secondly, in the embedding space, we close the distance between the positive sample and the anchor sample, and meanwhile separate the negative sample far away from the anchor sample in a contrastive-learning manner.

To sum up, our contributions are as follows:

- We find that the existing evidence-aware fake news detection models lack sufficient semantic perception, and conduct a pilot experiment on the extended test set to quantitatively demonstrate this observation.
- We propose a model-agnostic framework for robust evidence-aware fake news detection. In the framework, we complement the original dataset with two kinds of data augmentation, namely semantic-invariant augmentation and semantic-flipped augmentation, and learn the semantic-sensitive claim representation via our designed module.
- Experimental results show that the proposed method significantly outperforms the state-of-the-art methods on our extended test set, while achieving competitive performance on the original one.

2. Related Work

2.1. Fake News Detection

In recent years, researchers have proposed many approaches (Popat et al., 2016; Dou et al., 2021b; Popat et al., 2018; Ma et al., 2019; Wu et al., 2021b; Vo and Lee, 2021; Wu et al., 2021a; Xu et al., 2022) to automatically detect fake news, which can be mainly divided into three groups:

Pattern-Based Approaches. This group typically directs its attention toward the claim itself. For instance, Popat et al. (2016) utilize stylistic features and the stance of an article to ascertain the veracity of a claim. Similarly, Przybyla (2020) employ writing style for news classification. Recently, researchers also identify emotional bias in fake news and explore emotion mining for detection (Zhang et al., 2021; Giachanou et al., 2019).

Evidence-Based Approaches. This group typically integrates external evidences into their analysis of a claim. To the best of our knowledge, DeClarE (Popat et al., 2018) is the first to incorporate evidences into fake news detection. They employ BiLSTMs to extract semantic features and introduce an attention mechanism to calculate the attention score of each word in the evidence. Subsequently, some researchers have ventured into this direction and present models such as HAN (Ma et al., 2019), EHIAN (Wu et al., 2021b), MAC (Vo and Lee, 2021), CICD (Wu et al., 2021a), GET (Xu et al., 2022). These works explore different methods for extracting semantic features and propose various attention mechanisms to capture the interaction between the claim and evidences.

PLMs in Fake News Detection. Pre-trained Language Models (PLMs) have consistently demonstrated their superiority in the realm of natural language processing. In the field of fake news detection, some works (Dou et al., 2021a; De and Desarkar, 2022) have also leveraged their capabilities. However, there are few studies exploring the integration of PLMs into the evidence-aware fake news detection, where the application of PLMs poses particular challenges. The excessive length of the combination of a claim with its corresponding evidences makes it impractical to input the entire text sequence into Transformer-based models. To address this limiation, we first selectively retrieves the most pertinent sentences from the evidences, effectively curtailing the input length and ensuring com-

pliance with the input constraints of Transformer-based models, as elaborated in Sec. 4.5.2.

2.2. Robustness on Fake News Detection

Recently, the robustness of neural models has drawn much attention. Previous works design different simulated attack schemes to evaluate whether the system can successfully resist attacks (Abdelnabi and Fritz, 2022; Schuster et al., 2021; Du et al., 2022). Du et al. (2022) adversarially exploit GROVER (Zellers et al., 2019) to generate fake articles and then inject them into the retrieval database (e.g. FEVER DB). Abdelnabi and Fritz (2022) evaluate fake news detection under various attacks, such as lexical variation, and omitting paraphrase. Schuster et al. (2021) propose a new dataset named VITAMINC. This dataset is structured adversarially, featuring pairs of evidence for each claim with nearly identical language and content, yet one supports the claim while the other does not. They demonstrate empirically that this dataset improves the model robustness.
Nevertheless, Hansen et al. (2021) find some serious issues in several fake news detection models. For example, evidence-based models may not have the reasoning ability. Their experiments demonstrate that only using the claim or the evidence can yield better performance in certain cases. These findings point out the importance of trustworthy tools for fake news detection.

2.3. Contrastive Learning

In recent years, contrastive learning has been successfully and widely used in both computer vision (Chen et al., 2020; Grill et al., 2020) and natural language processing domains (Fang et al., 2020; Wang et al., 2021; Chuang et al., 2022). This approach focuses on constructing pairs of training samples with the objective of minimizing the distance between positive samples while simultaneously widening the gap between positive and negative samples in the embedding space. This ensures that similar samples yield similar representations, while dissimilar samples produce distinct representations. Through this methodology, the model effectively learns the intrinsic structure and patterns of the data, thereby enhancing its generalization capability and performance.

3. Pilot Experiment

We conduct a pilot experiment to quantitatively analyze the semantic perception of two state-of-the-art methods MAC (Vo and Lee, 2021) and GET (Xu et al., 2022) in evidence-aware fake news detection. For completeness, we also extend this analysis to include two pre-trained language models BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), and specifically adapt them to this task. The detailed setup is provided in Sec. 5.3.1.

Dataset	Model	F1-ma	F1-mi	F1-T	F1-F
Snopes	MAC	78.7	83.3	68.7	88.6
	GET	80.0	84.6	70.5	89.5
	BERT	72.8	78.5	60.4	85.2
	RoBERTa	72.2	77.8	59.8	84.7
Snopes -hard	MAC	58.8	60.3	51.2	66.5
	GET	58.7	60.8	49.5	67.9
	BERT	54.4	59.6	39.1	69.7
	RoBERTa	54.2	58.7	40.0	68.5

Table 1: Performance (%) of state-of-the-art methods on Snopes and Snopes-hard.

Moreover, We extend the original test set of the benchmark dataset Snopes into a hard version one, i.e., Snopes-hard, via two kinds of data augmentation introduced in this paper. This extension places a heightened demand on the model perception of semantics. Details could be found in Sec. 5.1.
The results are displayed in Table 1. Notably, when contrasting the performance of compared models on Snopes-hard with that on the standard Snopes dataset, a significant decline is observed across all evaluation metrics. This suggests that the current SOTA methods exhibit reduced robustness and struggle to effectively discern the semantic nuances within a claim based on its textual expression. This limitation hampers the further enhancement of detection performance and renders them susceptible to malicious attacks.

4. Methodology

4.1. Preliminaries

We first introduce the classical paradigm of evidence-aware fake news detection. Given a claim $\boldsymbol{c}$ and its associated evidences $\mathbf{E} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$, a detection model aims to determine the veracity of the claim based on the provided evidences. It is commonly practiced to formulate the detection task as a binary classification problem:

$$\hat{y} = f(\mathbf{c}, \mathbf{E}, \Theta), \tag{1}$$

where $\hat{y} \in \mathbb{R}^2$ is the predicted probability distribution and Θ represents the trainable model parameters. Given the ground-truth veracity y of the claim $\boldsymbol{c}$, the detection model is optimized by minimizing the cross-entropy loss between y and $\hat{y}$.

4.2. Overview

The overview of the proposed framework is shown in Figure 2. Given a claim $\boldsymbol{c}$ with its corresponding evidences $\boldsymbol{E} = \{\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_n\}$, we first generate its semantic-flipped augmentation $\boldsymbol{c}^-$ and semantic-invariant augmentation $\boldsymbol{c}^+$ respectively. Then we encode the text sequences via the text encoding layer. Next, we feed the encoded representation into the semantic-sensitive representation learning

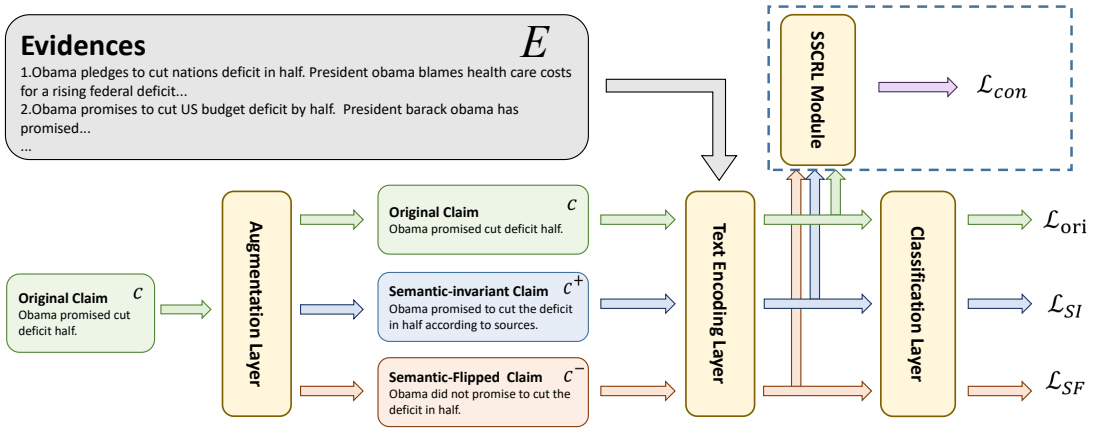

Figure 2: The Overview of the proposed framework. It consists of four parts: the *augmentation layer* that synthesizes semantic-flipped and semantic-invariant counterparts of the input claim, the *text encoding layer* for encoding the input claim with its corresponding evidences, the *semantic-sensitive claim representation learning* (SSCRL) module, and the *classification layer* for claim veracity prediction.

module to calculate the loss $\mathcal{L}_{con}$, and simultaneously, into the classification layer for claim veracity classification. Finally, a two-layer MLP is employed to obtain predicted probability distributions, and the cross-entropy losses $\mathcal{L}_{ori}$, $\mathcal{L}_{SF}$, and $\mathcal{L}_{SI}$ are computed for $\boldsymbol{c}$, $\boldsymbol{c}^-$, and $\boldsymbol{c}^+$, respectively. The overall loss of the proposed framework is defined as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{ori} + \lambda\mathcal{L}_{con} + \mu\mathcal{L}_{SF} + \nu\mathcal{L}_{SI}, \quad (2)$$

where λ, μ and ν are weighting coefficients that regulate the influence of individual losses.

4.3. Semantic-Flipped Augmentation

For a claim in the original dataset, the semantic-flipped augmentation generates a new one with a similar textual expression but opposite semantics. Specifically, we achieve this augmentation in two ways. The first is based on SpaCy[1], a widely used toolkit for NLP. We exploit SpaCy to analyze the lexicality of each word in the claim, and then add negation before the verb (or after the aux). For example, the claim *"Mccain opposes repeal death tax"* will be transformed into *"Mccain does not oppose repeal death tax"*. If the claim already contains negation, we simply remove the negation to flip its semantics. Note that there are also some claims where we could not directly add negation. In this scenario, we settle for second best and just select one piece of evidence of the current claim or another arbitrary claim as the augmented result, which only guarantees a noticeable semantic change. The details are elaborated in Alg. 1.

The second is based on ChatGPT. We have developed a specialized prompt that enables ChatGPT to generate the semantic-flip version of a claim:

[1] https://spacy.io/

Algorithm 1 Semantic-Flipped Augmentation based on SpaCy

Require: The original claim $\boldsymbol{c}$ with evidences $\boldsymbol{E}$ and ground truth y; A batch of claims $\mathcal{B}$, $\boldsymbol{c} \in \mathcal{B}$
Ensure: The new generated claim $\boldsymbol{c}^-$ for $\boldsymbol{c}$

```
if c contains negation then
    c⁻ ← Remove the negation from c
    return c⁻
if c can be added negation then
    c⁻ ← Add negation to c
    return c⁻
if y is False then
    c⁻ ← Randomly select e from E
else
    c⁻ ← Randomly select c̃ from B, c̃ ≠ c
return c⁻
```

> *Add negation to the following sentences to flip their semantics.*
> *For example.*
> *Mccain opposes repeal death tax.*
> *Mccain does not oppose repeal death tax.*

In the above two ways, we finally obtain the semantic-flipped version $\boldsymbol{c}^-$ for each claim $\boldsymbol{c}$.

4.4. Semantic-Invariant Augmentation

For a claim in the original dataset, the semantic-invariant augmentation generates a new one with the same semantics but a different writing style. We also achieve this goal in two ways. The first is based on the *paraphrase* technique, which rewrites sentences with the original semantics reserved. Concretely, we choose PEGASUS (Zhang et al., 2019) to automatically generate the semantic-invariant version for a claim. For example, the original claim is "*Obama signs bill forgiving student loan debt*",

and its paraphrase could be "*The student loan debt has been forgiven by Obama through the signing of a bill*".
The second is still based on ChatGPT. We have also designed a specialized prompt to generate the semantic-invariant version of a claim:

> *Rewrite each following sentence in a different writing style. Note that you need to keep the original semantics.*

With the above two approaches, we finally obtain the semantic-invariant version $\boldsymbol{c}^+$ for each claim $\boldsymbol{c}$.

4.5. Text Encoding Layer

After obtaining two kinds of augmentation for a claim, we exploit the text encoding layer to encode the original claim $\boldsymbol{c}$ and its augmentation $\boldsymbol{c}^-$, $\boldsymbol{c}^+$ into high-level representation $\boldsymbol{h}$, $\boldsymbol{h}^-$ and $\boldsymbol{h}^+$ for further veracity prediction. The corresponding evidences $\boldsymbol{E}$ are also encoded and incorporated into the representation. For completeness, we construct the text encoding layer in two ways, i.e., a graph-based one and a PLM-based one respectively. Next, we will take the original claim $\boldsymbol{c}$ as an example to illustrate the encoding process, and its augmentation $\boldsymbol{c}^-$, $\boldsymbol{c}^+$ is encoded in the same manner.

4.5.1. Graph-Based Text Encoding

For the graph-based one, we first represent the claim $\boldsymbol{c}$ as a graph. Specifically, we treat each word in the claim as a graph node and construct the adjacency matrix by sliding a fixed-sized window over the text sequence. Then, we further encode the obtained graph into the low-level representation $\boldsymbol{g}$ via Graph-Gated Neural Networks (GGNN). Finally, we incorporate the evidences into the representation via an attentive readout layer, resulting in the high-level representation $\boldsymbol{h}$. The implementation of this encoder adheres to the previous work (Xu et al., 2022) and please refer to it for a more comprehensive understanding.

4.5.2. PLM-Based Text Encoding

In the context of evidence-aware fake news detection, it is challenging to employ a PLM-based structure for text encoding. This is because the combined word count of the input claim $\boldsymbol{c}$ and its corresponding evidences $\boldsymbol{E}$ far exceeds of the input limit of Transformer-based models. To address this issue, we devise a two-stage strategy in which we first retrieve the most relevant sentences from the evidences and then encode the concatenation of the claim with these selected sentences.
Retrieval. We employ BERT as the backbone of our sentence retrieval model, following the previous work (Liu et al., 2020). For each sentence $\boldsymbol{s}$ from the evidences $\boldsymbol{E}$, we first pair it with the claim $\boldsymbol{c}$ and then utilize BERT to encode the sentence pair $(\boldsymbol{c}, \boldsymbol{s})$. Subsequently, we feed the $[\mathrm{CLS}]$ representation into an MLP layer to derive a ranking score as follows:

$$\mathrm{ranking\_score} = \tanh(\mathrm{MLP}(\mathrm{BERT}(\boldsymbol{c}, \boldsymbol{s}))). \quad (3)$$

We optimize the retrieval model using Pairwise Loss function. Finally, we sort the ranking scores of all the sentence pairs and concatenate the top 5 sentences as $\boldsymbol{S}$ for the second stage.
Encoding. We first concatenate the claim with the five top-ranked sentences to form a text sequence $\{[\mathrm{CLS}], \boldsymbol{c}, [\mathrm{SEP}], \boldsymbol{S}, [\mathrm{EOS}]\}$, and then feed it into a pre-trained language model as input. In this paper, we employ RoBERTa to encode the text sequence, and take the last hidden state of $[\mathrm{CLS}]$ as the high-level representation $\boldsymbol{h}$.

4.6. Claim Verification Prediction

After obtaining the high-level representation $\boldsymbol{h}$ for the claim, we feed it into a two-layer MLP to predict the probability distribution $\hat{y}$. Finally, we compute the cross-entropy loss $\mathcal{L}_{ori}$ as follows:

$$\mathcal{L}_{ori} = -(y\, log\, \hat{y} + (1-y)\, log(1-\hat{y})). \quad (4)$$

Note that in the above procedure, we can simply replace the claim representation $\boldsymbol{h}$ with its augmentation $\boldsymbol{h}^-$ or $\boldsymbol{h}^+$, and assign the corresponding ground-truth veracity to calculate $\mathcal{L}_{SF}$ or $\mathcal{L}_{SI}$.

4.7. Semantic-Sensitive Claim Representation Learning Module

We argue that the claim representation learned by existing models is not sensitive to the semantics, which leads to the lack of semantic perception and thus hinders the detection performance. To solve this problem, we propose a semantic-sensitive representation learning approach to explicitly promote differentiation among claims with distinct semantics and meanwhile facilitate the aggregation of those with similar semantics in the embedding space. Specifically, after obtaining the low-level claim representation $\boldsymbol{g}$, $\boldsymbol{g}^+$ and $\boldsymbol{g}^-$ in the graph-based encoding process, we learn the semantic-sensitive claim representation in a contrastive-learning manner as follows:

$$\mathcal{L}_{con} = -\log \frac{e^{sim(\boldsymbol{g}, \boldsymbol{g}^+)/\tau}}{e^{sim(\boldsymbol{g}, \boldsymbol{g}^+)/\tau} + \sum_{\tilde{\boldsymbol{g}} \in \mathcal{N}} e^{sim(\boldsymbol{g}, \tilde{\boldsymbol{g}})/\tau}}, \quad (5)$$

where τ is a temperature hyper-parameter, and $sim(\cdot, \cdot)$ denotes the cosine similarity function.
In Eq. 5, for an original sample $\boldsymbol{g}$, the positive sample is its semantic-invariant version $\boldsymbol{g}^+$, while the set $\mathcal{N}$ of negative samples includes its semantic-flipped version $\boldsymbol{g}^-$. Inspired by the self-supervised contrastive learning, we also add the other samples in the same batch with their augmentation into $\mathcal{N}$ during the training phase, since all of them have

Algorithm 2 Semantic-Sensitive Claim Representation Learning on Graph-Based Text Encoding

Require: The original sample $\boldsymbol{g}$ with its augmentation $\boldsymbol{g}^+$, $\boldsymbol{g}^-$; A batch of samples $\mathcal{B}$, where $\boldsymbol{g} \in \mathcal{B}$; The empty set $\mathcal{N}$
Ensure: $\mathcal{L}_{con}$

```
initialize N = {g^-}
for each sample g̃ ∈ B do
    if g̃ ≠ g then
        g̃^+ ← semantic-invariant version of g̃
        g̃^- ← semantic-flipped version of g̃
        Add g̃, g̃^+, g̃^- into N
Compute L_con by Eq. 5
return L_con
```

different semantics from the original sample. The details are elaborated in Alg. 2.

In the PLM-based encoding process, the model directly obtains the high-level claim representation $\boldsymbol{h}$, without the presence of its low-level counterpart $\boldsymbol{g}$ as in the graph-based encoding process. To facilitate adaptive adjustments, we conduct the semantic-sensitive claim representation learning based on $\boldsymbol{h}$ as follows:

$$\mathcal{L}_{con} = -\log \frac{e^{sim(\boldsymbol{h},\boldsymbol{h}^+)/\tau}}{e^{sim(\boldsymbol{h},\boldsymbol{h}^+)/\tau} + e^{sim(\boldsymbol{h},\boldsymbol{h}^-)/\tau}}. \quad (6)$$

5. Experiments

5.1. Experimental Setup

Datasets We evaluate the model performance on two common benchmark datasets, i.e., Snopes and PolitiFact (Popat et al., 2018). We allocate 10% of the entire dataset as the fixed validation set, while the remaining data is split into training and test sets in a 4:1 ratio. All experimental results were averaged over five-fold cross-validation, consistent with prior studies (Vo and Lee, 2021; Xu et al., 2022).

To facilitate a more accurate and efficient assessment of the semantic perception, we also extend the two datasets into their corresponding hard versions, i.e., Snopes-hard and PolitiFact-hard for model evaluation. Specifically, for each claim within the original dataset, we generate augmentations in both semantic-invariant and semantic-flipped forms, subsequently integrating them into the dataset.

Hyperparameter Settings For the graph-based model, we set the temperature parameter τ to 0.1, and the weight coefficients λ, μ, and ν to 0.5, 0.2, and 0.1 respectively. In the case of the PLM-based model, the temperature parameter τ is set to 0.3, while the weight coefficients λ, μ, and ν are set to 0.5, 1.0, and 1.0 respectively. Further implementation details are available in the appendix.

Evaluation The evaluation metrics include macro F1 (F1-ma), micro F1 (F1-mi), F1 score on true category (F1-T) and false category (F1-F).

Data Version	Augmentation	Rating Score
conv-hard	Invariant	136.4±17.2
	Flipped	182.7±8.8
gpt-hard	Invariant	174.8±12.6
	Flipped	185.0±8.1

Table 2: Human evaluation on Politifact-hard. "conv-hard" means data generated with conventional approaches, and "gpt-hard" means data generated using ChatGPT. We report the average score with the standard deviation.

5.2. Human Evaluation on Extend Test Sets

To ensure the quality of our extend test sets, we randomly selected from Politifact-hard 100 claims as well as their corresponding semantic-invariant and semantic-flipped augmentations for human evaluation. We invite 7 English proficient evaluators to rate the two kinds of augmentations. The rating scale ranges from 0 to 2. Here, we employ the semantic-flipped augmentation as an illustrative example to explain the scoring criteria, with the criteria for the semantic-invariant augmentation following a similar pattern. Firstly, a score of "2" is assigned when the sentence's semantics are entirely reversed. Secondly, when the semantics shift without becoming the opposite, a score of "1" is assigned. Thirdly, a score of "0" is assigned when the semantics remain entirely unchanged.

The results of human evaluation are displayed in Table 2, underscoring the overall quality of our extended test sets. Additionally, as mentioned in Sec. 4.3 and Sec. 4.4, we generate each kind of augmentation in two ways. We observe that data generated using ChatGPT exhibit higher quality compared to the conventional approach.

5.3. Comparison with State-of-the-Arts

5.3.1. Baselines

We compare the proposed method with various state-of-the-art methods in Table 3, including two traditional models **MAC** (Vo and Lee, 2021) and **GET** (Xu et al., 2022), and two pre-trained language models **BERT** (Devlin et al., 2019) and **RoBERTa** (Liu et al., 2019). Note that we employ the large versions of BERT and RoBERTa, and the encoding process of textual content adheres to Sec. 4.5.2.

For completeness, we also introduce several additional baselines on the ChatGPT-version extended test set. To investigate whether the effectiveness of our method solely arises from exposure to synthetic data, we incorporate two kinds of augmentations into the original training set to optimize the GET model (i.e., **GET+DA**) and the RoBERTa model (i.e., **RoBERTa+DA**), respectively. Additionally, we include **ChatGPT** as a baseline for comparison.

Data Version	Model	PolitiFact				Snopes			
		F1-ma	F1-mi	F1-T	F1-F	F1-ma	F1-mi	F1-T	F1-F
original	MAC	68.6	69.1	71.8	65.5	78.7	83.3	68.7	88.6
	GET	69.1	69.4	72.3	66.0	80.0	84.6	70.5	89.5
	BERT	62.8	63.1	64.0	61.5	72.8	78.5	60.4	85.2
	RoBERTa	62.1	65.3	64.6	59.7	72.2	77.8	59.8	84.7
	Ours (conv aug & graph enc)	68.8	69.2	72.2	65.3	76.8	81.9	66.0	87.6
	Ours (gpt aug & graph enc)	68.0	68.3	70.6	65.3	77.2	81.8	66.9	87.5
conv-hard	MAC	56.2	56.2	56.4	56.0	58.1	60.6	48.2	68.0
	GET	56.9	57.1	58.9	54.9	58.1	60.6	48.0	68.2
	BERT	54.8	55.2	54.8	54.9	55.1	59.5	41.1	69.1
	RoBERTa	53.5	54.0	53.5	53.5	54.6	58.9	40.6	68.6
	Ours (conv aug & graph enc)	61.6	61.6	62.5	62.1	77.8	78.3	74.6	81.0
	Ours (conv aug & PLM enc)	**64.3**†	**64.4**†	**65.3**†	**63.4**†	**78.6**†	**79.3**†	**74.7**†	**82.4**†
gpt-hard	MAC	55.4	55.6	57.8	53.0	58.8	60.3	51.2	66.5
	GET	56.5	56.7	58.1	54.9	58.7	60.8	49.5	67.9
	BERT	53.6	54.1	55.2	52.0	54.4	59.6	39.1	69.7
	RoBERTa	53.1	53.8	51.8	54.4	54.2	58.7	40.0	68.5
	GET+DA	56.7	56.7	56.7	56.7	74.0	74.6	69.8	78.1
	RoBERTa+DA	56.7	57.2	54.2	59.3	75.7	76.4	71.9	79.6
	ChatGPT	58.4	58.5	57.8	59.2	58.5	61.5	47.3	69.7
	Ours (gpt aug & graph enc)	61.2	61.2	61.8	60.5	**80.1**†	**80.6**†	**77.0**†	**83.2**†
	Ours (gpt aug & PLM enc)	**62.1**†	**62.2**†	**63.0**†	**61.3**†	77.9	78.5	74.2	81.6

Table 3: Performance (%) on Snopes and Politifact. The notation "Data Version" refers to the version of test sets, where "original" means the original test sets of Politifact and Snopes, and "conv-hard" and "gpt-hard" means their corresponding hard-version test sets extended by conventional approaches and ChatGPT respectively. The notation † indicates that **the performance improvement is significant** with p**-values**< 0.05.

This model utilizes a meticulously designed prompt to determine the veracity of a claim. Please consult the appendix for details regarding the prompt.

In the proposed framework, both the augmentation layer and the text encoding layer are achieved in two ways. Therefore, we implement several variants of our method **Ours** for comparison. The notations "conv aug" and "gpt aug" denote augmentation based on the conventional techniques and ChatGPT respectively, while the notations "graph enc" and "PLM enc" represent that text sequences are encoded in a graph-based and PLM-based manner respectively.

5.3.2. Results Analysis

We evaluate various methods on different versions of test sets, and the results are displayed in Table 3. First, we observe that our method outperforms all the baselines on both kinds of extended test sets. We utilize the competitive ChatGPT as a reference to conduct a thorough performance comparison on the extended test sets of the "gpt-hard" version. On Snopes, our method with a graph-based encoder achieves significant improvement by +21.6% on F1-ma and +19.1% on F1-mi respectively. On PolitiFact, our method with a PLM-based encoder exhibits a notable increase of +3.7% on both F1-ma and F1-mi. This demonstrates the effectiveness of our approach and shows that it really improves the model perception of semantics.

Second, the proposed method exhibits competitive performance on the original test sets, which indicates that the model achieves a favorable balance between the original data and the sythetic data, showing effective generalization without being overly influenced by data biases.

Thirdly, despite employing a similar encoding architecture, GET+DA generally performs better than GET, and RoBERTa+DA significantly outperforms RoBERTa. This observation validates the efficacy of the proposed data augmentation. Moreover, our approaches with a graph-based encoder and a PLM-based encoder exhibit remarkable superiority over GET+DA and RoBERTa+DA, respectively. For instance, on the "gpt-hard" version of Snopes, our method improves over GET+DA by +6.1% on F1-ma and +6.0% on F1-mi, respectively. Similarly, on the "gpt-hard" version of PolitiFact, it outperforms RoBERTa+DA by +5.4% on F1-ma and +5.0% on F1-mi, respectively. This underscores that the superiority of our method does not solely rely on exposure to synthetic data during training, and the full potential of data augmentation can only be realized through our multi-objective optimization paradigm.

Finally, when employing a similar PLM-based encoding architecture, our approach consistently outperforms traditional models like MAC and GET, whereas BERT and RoBERTa typically exhibit inferior performance compared to these two models. This observation demonstrates the effectiveness of our proposed framework in fully harnessing the potential of pre-trained language models.

text encoding	#	$\mathcal{L}_{ori}$	$\mathcal{L}_{con}$	$\mathcal{L}_{SF}$	$\mathcal{L}_{SI}$	F1-ma	F1-mi	F1-T	F1-F
graph-based	1	✓				58.7	60.8	49.5	67.9
	2	✓	✓			59.3	61.5	49.8	68.8
	3	✓	✓	✓		71.2	71.8	67.1	75.2
	4	✓	✓	✓	✓	**80.1**	**80.6**	**77.0**	**83.2**
PLM-based	5	✓				54.4	59.6	39.1	69.7
	6	✓	✓			76.9	77.1	74.4	79.3
	7	✓	✓	✓		78.2	78.8	74.6	81.7
	8	✓	✓	✓	✓	**78.6**	**79.4**	**74.8**	**82.5**

Table 4: Ablative performance (%) on the ChatGPT version of Snopes-hard.

Model		PolitiFact				Snopes			
		F1-ma	F1-mi	F1-T	F1-F	F1-ma	F1-mi	F1-T	F1-F
MAC	base	55.4	55.6	57.8	53.0	58.8	60.3	51.2	66.5
	+Ours	59.4 (↑4.0)	59.4(↑3.8)	59.4(↑1.6)	59.4(↑6.4)	78.3(↑19.5)	78.9(↑18.6)	74.8(↑23.6)	81.8(↑15.3)
GET	base	56.5	56.7	58.1	54.9	58.7	60.8	49.5	67.9
	+Ours	61.2(↑4.7)	61.2(↑4.5)	61.8(↑3.7)	60.5(↑5.6)	80.1(↑21.4)	80.6(↑19.8)	77.0(↑27.5)	83.2(↑15.3)
BERT	base	53.6	54.1	55.2	52.0	54.4	59.6	39.1	69.7
	+Ours	61.7(↑8.1)	61.8(↑7.7)	60.8(↑5.6)	62.7(↑10.7)	78.6(↑24.2)	79.4(↑19.8)	74.8(↑35.7)	82.5(↑12.8)
RoBERTa	base	53.1	53.8	51.8	54.4	54.2	58.7	40.0	68.5
	+Ours	62.1(↑9.0)	62.2(↑8.4)	63.0(↑11.2)	61.3(↑6.9)	77.9(↑23.7)	78.5(↑19.8)	74.2(↑34.2)	81.6(↑13.1)

Table 5: Performance (%) of different model architectures on the ChatGPT version of PolitiFact-hard and Snopes-hard. The notation "base" denotes the original method, and "+Ours" denotes applying the proposed framework to the corresponding architecture.

5.4. Ablation Study

As shown in Table 4, we conduct an ablation study to demonstrate the contributions of individual losses in the proposed multi-objective optimization paradigm. For completeness, we employ two variants of our method with different text encoding architectures for this study.

In the PLM-based variant, comparing Row 6 with Row 5, we observe a substantial improvement in model performance with the introduction of $\mathcal{L}_{con}$. For example, the inclusion of $\mathcal{L}_{con}$ results in an impressive +22.5% enhancement on F1-ma and a notable +17.5% increase on F1-mi. This substantiates the effectiveness of semantic-sensitive claim representation learning.

In the graph-based variant, comparing Row 2 with Row 1, we observe that the detection performance slightly improves by adding $\mathcal{L}_{con}$. Furthermore, comparing Row 2 with both Row 3 and Row 4, both $\mathcal{L}_{SF}$ and $\mathcal{L}_{SI}$ further significantly improve the detection performance. This suggests that despite the construction of semantic-sensitive claim representations, further explicit supervisory signals are still required to fully exploit these representations.

In summary, in both two variants, the model performance gradually improves as one more loss is incorporated, which indicates that each individual loss is indispensable and demonstrates the superiority of the multi-objective optimization.

5.5. Model Agnostic Study

Since the proposed framework is model-agnostic, we incorporate it into different model architectures to evaluate its effectiveness. The results are displayed in Table 5. Our method consistently and significantly improves the performance on both Politifact-hard and Snopes-hard. For instance, when compared to RoBERTa, our method yields remarkable improvements on PolitiFact-hard, with a substantial increase of +9.0% on F1-ma and +8.4% on F1-mi, respectively. On Snopes-hard, our method demonstrates even more substantial gains, with a significant enhancement of +23.7% on F1-ma and +19.8% on F1-mi. This indicates that our framework can be effectively integrated into different architectures to enhance the model robustness.

5.6. Case Study

We present qualitative examples in Figure 3 for comparing our method with GET and RoBERTa. We observe that the veracity of the original claim is consistently predicted accurately across various models. However, when dealing with semantic-invariant and semantic-flipped versions of the claim, GET and RoBERTa consistently make incorrect predictions. This underscores the insufficient semantic awareness of existing methods, leading to inaccuracies in their predictions. Our approach, incorporating data augmentation and Semantic-Sensitive Claim Representation Learning Module, successfully capture semantics, resulting in accurate predictions.

Claim Version	The Content of the Claim	GET	Ours
Original	Photograph shows bathroom unusually painted floor.	False ✓	False ✓
Semantic-invariant	The photograph depicts a bathroom with an unusually painted floor.	True ✗	False ✓
Semantic-Flipped	Photograph does not show a bathroom with an unusually painted floor.	False ✗	True ✓

(a) An example on models with a graph-based text encoder

Claim Version	The Content of the Claim	RoBERTa	Ours
Original	Obama charge 28 percent tax on home sales.	False ✓	False ✓
Semantic-invariant	Obama imposes a 28% tax on home sales.	True ✗	False ✓
Semantic-Flipped	Obama does not charge 28 percent tax on home sales	False ✗	True ✓

(b) An example on models with a PLM-based text encoder

Figure 3: Case study on the veracity predictions of different claim versions, which are made by compared methods. *Green* indicates correct predictions, while *red* indicates incorrect predictions.

6. Conclusion

In this paper, we present a model-agnostic training framework for robust evidence-aware fake news detection, aiming to enhance the model semantic perception. We introduce semantic-flipped augmentation and semantic-invariant augmentation to complement the original datasets. Additionally, we propose the semantic-sensitive claim representation learning module, which improves the sensitivity of claim representation to semantics. These components collaboratively form a multi-objective paradigm. Experiments show that our method significantly outperforms the state-of-the-art methods on our extended test sets, while achieving competitive performance on the original one.

Acknowledgements

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

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

Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31(2):211–36.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljacic, Shang-Wen Li, Scott Yih, Yoon Kim, and James Glass. 2022. DiffCSE: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle, United States. Association for Computational Linguistics.

Arkadipta De and Maunendra Sankar Desarkar. 2022. Multi-context based neural approach for covid-19 fake-news detection. In *Companion Proceedings of the Web Conference 2022*, WWW '22, page 852–859, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yingtong Dou, Kai Shu, Congyin Xia, Philip S. Yu, and Lichao Sun. 2021a. User preference-aware fake news detection. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Yingtong Dou, Kai Shu, Congying Xia, Philip S. Yu, and Lichao Sun. 2021b. User preference-aware fake news detection. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2051–2055, New York, NY, USA. Association for Computing Machinery.

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

Hongchao Fang, Sicheng Wang, Meng Zhou, Jiayuan Ding, and Pengtao Xie. 2020. Cert: Contrastive self-supervised learning for language understanding. *ArXiv*, abs/2005.12766.

Anastasia Giachanou, Paolo Rosso, and Fabio Crestani. 2019. Leveraging emotional signals for credibility detection. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 877–880, New York, NY, USA. Association for Computing Machinery.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. 2020. Bootstrap your own latent a new approach to self-supervised learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Casper Hansen, Christian Hansen, and Lucas Chaves Lima. 2021. Automatic fake news detection: Are models learning to reason? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 80–86, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2020. Fine-grained fact verification with kernel graph attention network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7342–7351, Online. Association for Computational Linguistics.

Jing Ma, Wei Gao, Shafiq Joty, and Kam-Fai Wong. 2019. Sentence-level evidence embedding for claim verification with hierarchical attention networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2561–2571, Florence, Italy. Association for Computational Linguistics.

Salman Bin Naeem and Rubina Bhatti. 2020. The covid-19 'infodemic': a new front for information professionals. *Health Information & Libraries Journal*, 37(3):233–239.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2016. Credibility assessment of textual claims on the web. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, page 2173–2178, New York, NY, USA. Association for Computing Machinery.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. DeClarE: Debunking fake news and false claims using evidence-aware deep learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 22–32, Brussels, Belgium. Association for Computational Linguistics.

Piotr Przybyla. 2020. Capturing the style of fake news. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(01):490–497.

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin c! robust fact verification with contrastive evidence. *ArXiv*, abs/2103.08541.

Nguyen Vo and Kyumin Lee. 2021. Hierarchical multi-head attentive network for evidence-aware fake news detection. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 965–975, Online. Association for Computational Linguistics.

Dong Wang, Ning Ding, Piji Li, and Haitao Zheng. 2021. CLINE: Contrastive learning with semantic negative examples for natural language understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2332–2342, Online. Association for Computational Linguistics.

Lianwei Wu, Yuan Rao, Yuqian Lan, Ling Sun, and Zhaoyin Qi. 2021a. Unified dual-view cognitive model for interpretable claim verification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 59–68, Online. Association for Computational Linguistics.

Lianwei Wu, Yuan Rao, Xiong Yang, Wanzhen Wang, and Ambreen Nazir. 2021b. Evidence-aware hierarchical interactive attention networks for explainable claim verification. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20.

Weizhi Xu, Junfei Wu, Qiang Liu, Shu Wu, and Liang Wang. 2022. Evidence-aware fake news detection with graph neural networks. In *Proceedings of the ACM Web Conference 2022*, WWW '22, page 2501–2510, New York, NY, USA. Association for Computing Machinery.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. *Defending against Neural Fake News*. Curran Associates Inc., Red Hook, NY, USA.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *ArXiv*, abs/1912.08777.

Xueyao Zhang, Juan Cao, Xirong Li, Qiang Sheng, Lei Zhong, and Kai Shu. 2021. Mining dual emotion for fake news detection. In *Proceedings of the Web Conference 2021*, WWW '21, page 3465–3476, New York, NY, USA. Association for Computing Machinery.

A. Experimental Details

A.1. Software and Hardware

We use Python 3.9.13 and PyTorch 1.12.1, and conduct experiments on a Linux server equipped with AMD Ryzen 9 5900X and NVIDIA RTX A6000 and 64GB of RAM.

A.2. Implementation Details

For the graph-based model, we primarily adhere to the methodology outlined in previous research (Xu et al., 2022). We employ GloVe word embeddings with an embedding dimension of 300. Additionally, the embedding dimension for claim speakers and evidence publishers is set to 128. The attention mechanism utilizes 3 heads for claims and 1 head for evidence on PolitiFact, while on Snopes, it employs 5 heads for claims and 2 heads for evidence. We train with a batch size of 32 for a maximum of 200 epochs, utilizing the Adam optimizer with a learning rate of 0.0001.

As for the PLM-based model, we use a smaller batch size of 4 and train for a maximum of 10 epochs. The learning rate is set to $3 \times e^{-6}$ to fine-tune the pre-trained language model.

A.3. Evaluation Metrics

We use the API of the sklearn library to calculate our evaluation metrics. F1-macro, F1-micro, F1-True, F1-False are calculated by calling sklearn.metrics.f1_score().

A.4. Links to Baseline Methods

- **BERT:** https://github.com/google-research/bert
- **MAC:** https://github.com/nguyenvo09/EACL2021
- **GET:** https://github.com/CRIPAC-DIG/GET

A.5. The Prompt for ChatGPT Baseline

We demonstrate the prompt for directly utilizing ChatGPT in fake news detection as follows:

> *You are a fact checker and your task is to determine the validity of a claim based on the evidence provided. You must provide a reasoning process to support your decision, whether claim be True or False.*
> *Here are some rules for labeling.*
> *1.If the evidence proves the claim is true, label it as "True".*
> *2.If the facts described in the evidence conflict with the claim, it should be classified as "False".*
> *Using the following format:*
> *Reasoning: <reasoning process>*
> *Label: <label>*
> *There is the claim and evidence. Claim:[the input claim] Evidences:[the input evidences, separated by "\n"]*

A.6. Datasets

	Snopes	PolitiFact
True	1164	1867
False	3177	1701
Speakers	N/A	664
Evidences	29242	29556
Publishers	12234	4542

Table 6: Dataset statistics.

The data statistics of original Snopes and Politifact are shown in Table 6. Both datasets are publicly available at https://github.com/nguyenvo09/EACL2021/tree/main/formatted_data/declare.

Snopes The claims and their corresponding labels from Snopes (Popat et al., 2018) are obtained by crawling the fact-checking website[2], while the evidences and their publishers are collected by querying the claims through a search engine.

PolitiFact The claims and their corresponding labels from PolitiFact (Popat et al., 2018) are from another fact-checking website[3], while the evidences and their publishers are also obtained through search engines. Unlike Snopes, PolitiFact also provides information about speakers for claims. We follow previous work (Popat et al., 2018) and merge the *true*, *mostly true*, *half true* into *true* and *false*, *mostly false*, *pants on fire* into *false*.

[2] https://www.snopes.com/
[3] https://www.politifact.com/