Evolving to the Future: Unseen Event Adaptive Fake News Detection on Social Media

Anonymous ACL submission

Abstract

With the rapid development of social media, the wide dissemination of fake news on social media is increasingly threatening both individuals and society. In the dynamic landscape of social media, fake news detection aims to develop a model trained on news reporting past events. The objective is to predict and identify fake news about future events, which often relate to subjects entirely different from those in the past. However, existing fake detection methods exhibit a lack of robustness and cannot generalize to unseen events. To address this, we introduce **F**uture **AD**aptive **E**vent-based Fake news Detection (**FADE**) framework. Specifically, we train a target predictor through an adaptive augmentation strategy and graph contrastive learning to make more robust overall predictions. Simultaneously, we independently train an event-only predictor to obtain biased predictions. Then we further mitigate event bias by obtaining the final prediction by subtracting the output of the event-only predictor from the output of the target predictor. Encouraging results from experiments designed to emulate real-world social media conditions validate the effectiveness of our method in comparison to existing state-of-the-art approaches.

1 Introduction

With the rapid development of the Internet, social media has become a platform for people to express their opinions and obtain information. While beneficial in many ways, this trend has also led to the proliferation of fake news. Nowadays, fake news has become more and more common in the era of mobile internet and social media since viewing and spreading fake news become much easier. Worse, the spread of fake news has been found to partially shape a country's public opinion, leading to economic loss and serious political consequences. Thus, fake news detection becomes a crucial problem waiting to be solved.

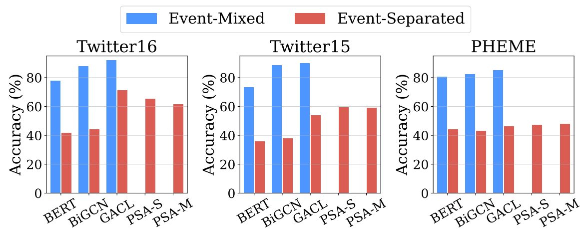

Figure 1: Comparing between event-mixed and event-separated settings, mean accuracy based on 10 different runs of each approach (PSA-S and PSA-M are methods designed specifically for event-separated scenarios, hence, their performance was not tested under event-mixed settings)

In real-world scenarios, a fake news detection model is trained on news reporting past events and expected to detect fake news pieces about future events. In the realm of social media, trending events are inherently dynamic and ever-changing, fake news is often crafted around current hot-button events that capture public attention. In other words, the training and testing data is non-independent and identically distributed (non-iid). The conversation graph of news within different events exhibits entirely distinct propagation structures and node attributions, which places high demands on the robustness of the detecting model. However, most existing methods assume that the training and testing news pieces are sampled iid from the same static news environment. They utilize an experimental setup based on this assumption to test model performance, which we refer to as event-mixed fake news detection. This setup leads to their actual detection capabilities being seriously overestimated.

After comparative experiments, we found that existing methods generally perform well under event-mixed experimental setup. However, in event-separated fake news detection (Wu and Hooi, 2022), where the test data contains news pieces from a set of events unseen during training, their accuracy drops significantly by over 40% as shown in 1. This startling result indicates that current methods lack effective detection capabilities when confronted with fake news from unseen events in real-world social media scenarios. We believe the deficiencies of existing models primarily lie in two aspects: (1) **Insufficient Robustness:** news under different events often exhibit vastly different propagation structures. For instance, news about celebrity gossip or popular culture tends to form flat propagation trees, whereas news on political or social issues often results in trees with greater depth. Additionally, news from different events can have vastly different textual feature distributions. Existing methods inadequately consider these variations, resulting in a lack of robustness when dealing with unseen events. This limitation hinders their ability to effectively detect fake news in such scenarios. (2) **Inadequate Generalization:** within each event, there are numerous highly similar keyword-sharing samples with the same class label. As shown in Figure 2, among all 48 news samples in the event 'E689', they all have the class label 'True', with 46 of them sharing the keywords 'white house' and 'rainbow'. Similarly, the event 'CIKM_1000737' includes 80 news items labeled 'True', of which 78 contain the keyword 'paul walker'. Existing methods utilize these keywords as spurious cues for inference. While such models perform well in event-mixed detection, they lack generalizability when faced with unseen events.

To bridge the event gap between news pieces in different periods and achieve more generalized and robust detection, we propose a FADE framework for fake news detection in this paper. Overall, our framework consists of a target predictor and an event-only predictor, each trained independently. (1) **Target Predictor:** data augmentation is a common training strategy that enhances the robustness of models by generating a diverse range of training samples. We propose an efficient graph augmentation strategy named adaptive augmentation, which generates the most challenging augmented samples in the representation space. We then use high-quality augmented training data to train a target predictor through graph contrastive learning, thereby providing robust predictions. (2) **Event-Only Predictor:** common debiasing methods like adversarial debiasing and reweighting, which are employed during the training stage for debiasing, are not suitable for the task of fake news detection due to the excessive number of event categories involved. To address this challenge, inspired by the Potential Outcomes Model (Sekhon, 2008), we propose to train an event-only predictor and use it for debiasing during the inference stage. Specifically, in training the event-only predictor, we incorporate an average pooling layer for samples under the same event. This enables it to generate predictions driven by event biases. We regard the prediction from the target predictor as a combination of unbiased features and biases inherent in the news. Consequently, we obtain the final debiased prediction by subtracting the event-label biased prediction from the target predictor's prediction during the inference stage.

Overall, the main contributions can be summarized as follows:

- We innovatively propose an adaptive augmentation strategy to produce the most demanding augmentations in the representation space, achieving significant performance gains while avoiding the need for manually designing augmentation strategies and intensities for different news datasets.
- We further introduce an inference stage debiasing method, indirectly obtaining unbiased inferences through the combination of biased predictions. This approach effectively enhances the framework's generalizability when dealing with news within unseen events.
- To our best knowledge, we are the first to effectively address fake news detection in an event-separated setting. Our empirical findings illustrate that our framework markedly surpasses existing state-of-the-art baselines.

2 Related Works

2.1 Fake news Detection Methods

Recently, many methods have been put forward for fake news detection. Yu et al. (2017) propose a Convolutional Neural Network (CNN) based model to extract key features scattered among an input sequence to identify fake news. Liu et al. (2018) andYu et al. (2019), utilizing the attention mechanism, have significantly improved fake news detection accuracy. The RvNN-based rumor detection introduced by Ma et al. (2018) employs both bottom-up and top-down propagation trees to learn the embedding of a fake news propagation structure. Building upon this, Bi-GCN (Bian et al., 2020) integrates a Graph Convolutional Network (GCN) into existing structures, marking the first application of

Event	Content	Label
E689	as sun goes down, **white house** lights up **rainbow** colors to celebrate scotus ruling	True
	the **white house** takes on **rainbow** hues in celebrating	
	there will be cool photos of the **white house** with **rainbow** colors tonight but hard to top this one by chuck kennedy.	
	see the **white house** light up as a **rainbow** to celebrate gay marriage	
	if they can light up the **white house** like a **rainbow** for gay pride, it sure as hell better be red, white & blue for independence day.	
CIKM_1000737	"**paul walker**'s character in fast and the furious was named "brian",brian from family guy also died this week.	True
	my heart goes out to loved ones and fans of **paul walker**, who died in a car wreck saturday.	
	rip roger rodas the man who died with **paul walker** in the fatal car crash	
	paul walker died shortly after attending a charity event for his organization reach out worldwide	
	r.i.p **paul walker**, why are people making jokes about his death? not funny at all!	

Figure 2: In the news content within the same event, there are numerous repeated keywords that can be used as spurious cues between the event and the label. The **bolded** words represent the repeated keywords.

GCN in social media rumor detection, and setting a new standard in performance.

The common shortcoming of the aforementioned methods is their inadequate consideration of model robustness and generalizability. GACL (Sun et al., 2022) makes a groundbreaking move by introducing contrastive learning into fake news detection, which, through the AFT module, enhances the model's robustness. Ma et al. (2022) proposes a hard positive sample pairs generation method (HPG) for conversation graphs, bolstering the model's resistance to interference. Wu and Hooi (2022) improves model performance and generalizability by integrating aggregated Publisher Style features as auxiliary information into their classification model. Furthermore, they introduce a more realistic social media fake news detection task, termed event-separated fake news detection. While these methods have made substantial strides toward improving classification model robustness and generalizability, their performance remains insufficient when dealing with the unseen events of real-world social media scenarios.

2.2 Data Augmentation

Data augmentation has been empirically validated as a highly effective strategy for enhancing the performance of deep learning models, particularly within the scope of classification tasks. For image data, an array of transformation or distortion techniques have been developed to generate a wealth of augmented samples. These techniques include but are not limited to flipping, cropping, rotation, scaling, and injection of noise, as well as transformations within the color space (Krizhevsky et al., 2012; Sato et al., 2015; Simard et al., 2003; Singh et al., 2018). In the realm of text data, augmentation methodologies generally fall into one of three categories: those based on paraphrasing (Madnani and Dorr, 2010; Wang and Yang, 2015), those based on the introduction of noise (Wei and Zou, 2019), and those relying on the sampling of existing data(Min et al., 2020). These data augmentation techniques have found broad application in the realm of deep learning, where they are employed to counteract overfitting and promote the robustness of deep neural network models.

Although image and text augmentations have been widely explored, undertaking augmentations for graphs presents more formidable challenges. Predominant methodologies currently in existence are rooted in the random alteration of graph structures or features, encompassing tactics such as random node dropping, perturbing edges, or feature masking (Hamilton et al., 2017; Wang et al., 2020; You et al., 2020; Rong et al., 2019; Zhu et al., 2021). Nevertheless, while these random transformations have shown some effectiveness on certain benchmark datasets, their performance often falls short when applied to the task of fake news detection.

2.3 Model Debiasing

Task-specific biases have been identified in many tasks, such as pre-trained language models (Meade et al., 2021), fact checking (Schuster et al., 2019; Xu et al., 2023), recommendation (Chen et al., 2023), and the biases present in task datasets can lead to models learning biased predictions. Debiasing methods primarily fall into two categories: data-level processing (Dixon et al., 2018; Wei and Zou, 2019) and model-level balancing strategies (Kaneko and Bollegala, 2019; Kang et al., 2019). For fake news detection, Zhu et al. (2022) proposes a framework to mitigate entity bias from a cause-effect perspective, while Wu and Hooi (2022) is the first to identify event bias, which is the very bias this paper aims to address.

3 Method

3.1 Problem Definition

Fake news detection is a classification task. The objective is to train a classifier using labeled instances and then deploy this trained model to predict the labels of unseen test instances.

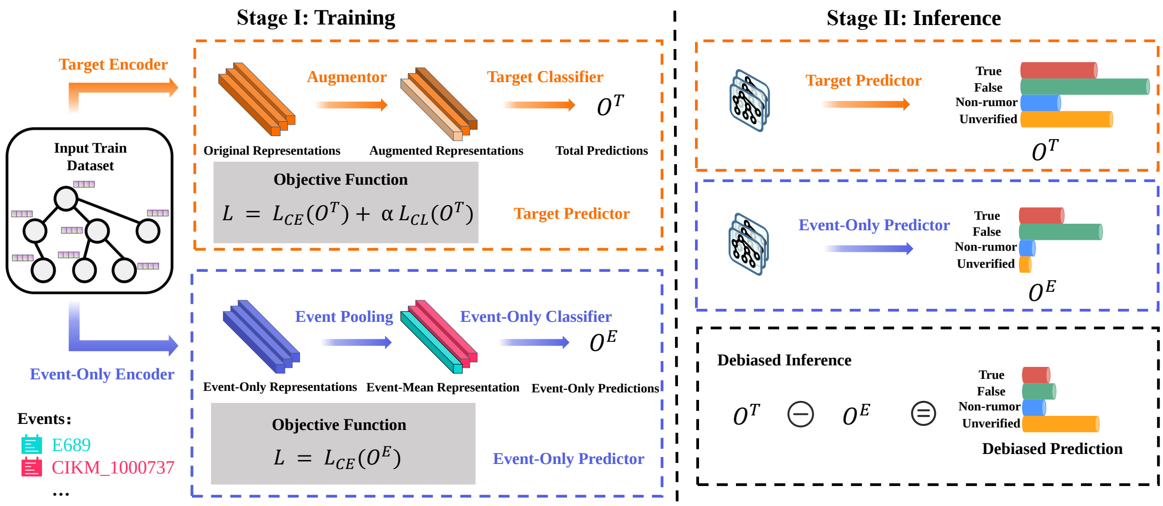

Figure 3: Overview of our FADE framework. In the training stage, given an input batch of data, we simultaneously use it to train both the main classifier and the Event-Only classifier. The main classifier is trained using contrastive loss and cross-entropy loss, while the event-only classifier is trained solely with cross-entropy loss. In the inference stage, each sample is predicted separately using both the target predictor and the event-only predictor. We then subtract the event-only prediction from the target prediction to obtain the debiased prediction, i.e., the final output.

Given an news instance set $\mathcal{C} = \{c_1, c_2, ..., c_m\}$ of size m, each instance c_i can be delineated as $c_i = \{r_i, w_1^i, w_2^i, ..., w_{n_i-1}^i, P_i\}$. Here, n_i denotes the count of posts in c_i with r_i being the source post, each w_j^i denotes the j-th comment post, and P_i denotes the propagation structure.

To each instance c_i, there corresponds a ground-truth label $y_i \in \{R, N\}$ (i.e. Rumor or Non-Rumor) and an event label e_i. In some cases, fake news detection is defined as a four-class classification task, correspondingly, $y_i \in \{N, F, T, R\}$ (i.e. Non-rumor, False Rumor, True Rumor, and Unverified Rumor). The label e_i encapsulates the event associated with the instance c_i.

To represent the propagation structure, we translate each instance c_i to a graph $G_i = (V_i, \mathbf{X_i}, \mathbf{A_i})$. $\mathcal{V}_i = \{r_i, w_1^i, w_2^i, ..., w_{n_i-1}^i\}$ denotes the vertex set. $\mathbf{X_i} \in \mathbb{R}^{n_i \times d}$ denotes the text features of each vertex, which are embedded using a pre-trained BERT model. $\mathbf{A_i} \in \{0, 1\}^{n_i \times n_i}$ is the adjacency matrix. Specifically, $a_{jk}^i = 1$ indicates a reply relationship between post j and post k, else $a_{jk}^i = 0$.

Given these definitions, the dataset for fake news detection can be expressed as $\mathcal{S} = \{(G_1, y_1, e_1), (G_2, y_2, e_2), \ldots, (G_m, y_m, e_m)\}$. We define the set of events in the training set as $\mathcal{E}_{tr}$ and the set of events in the test set as $\mathcal{E}_{te}$. When $\mathcal{E}_{tr} \cap \mathcal{E}_{te} \neq \emptyset$, we refer to such tasks as event-mixed fake news detection. Conversely, when $\mathcal{E}_{tr} \cap \mathcal{E}_{te} = \emptyset$, we term these tasks as event-separated fake news detection.

3.2 Model Overview

Figure 3 illustrates the overview of the FADE framework. It comprises a training stage and an inference stage. In the training stage, the target predictor (a combination of the GCN-based target encoder and classifier) is trained through adaptive augmentation and graph contrastive learning, enabling them to make predictions with strong generalizability and robustness. Meanwhile, the event-only predictor (a combination of the GCN-based event-only encoder and classifier) is trained using event-mean pooling, to ensure that the predictions are predominantly derived from event bias. In the inference stage, we subtract the prediction of the event-only predictor from that of the target predictor to obtain the final debiased prediction.

3.3 GCN-based Encoder

Leveraging the power of Graph Convolutional Network (GCN) (Kipf and Welling, 2016), we extract graph-level representations from structured data. The computational formula for the l-th layer with weight matrix $\mathbf{W}^{(l)}$ is:

$$H^{(l+1)} = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{\frac{1}{2}}H^{(l)}\mathbf{W}^{(l)}\right), \quad (1)$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_n$, is the adjacency matrix of the graph G with added self-connections. $\mathbf{I}_N$ is the identity matrix. $\tilde{\mathbf{D}}$ is the degree matrix of $\tilde{\mathbf{A}}$, $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$, and $H^0 = X$. $\sigma(\cdot)$ denotes an activation function. To get graph-level representations from node-level representations, we use:

$$R = \text{Pooling}(H^L). \quad (2)$$

where L is the number of layers, and the Pooling function is a permutation invariant function, such as mean or add. Additionally, R^O denotes the original graph representations. Furthermore, both the target encoder and the event-only encoder are identical GCN-based Encoders.

3.4 Adaptive Graph Augmentation

Existing data augmentation strategies rely on manually selecting and combining several basic augmentations like node dropping, edge perturbation, attribute masking, and subgraph extraction with manually set intensities. These strategies are not sufficiently powerful and lack universality across different datasets. To address this issue, we propose a powerful, efficient, and versatile augmentation strategy namely adaptive augmentation. Specifically, we perform the augmentation in the representation space by adding a perturbation to the original representation R^O. In our experiment, we first calculate the centroid and the average Euclidean distance between each original representation and the centroid as d by the following formula:

$$d = \frac{1}{N}\sum_{i=1}^{N} \| \frac{1}{N}\sum_{j=1}^{N} R_j^O - R_i^O \|_2. \quad (3)$$

where N denotes the number of samples. Then in the generation process, each time, we stochastically generate multiple random unit vectors. Each unit vector is represented by $\boldsymbol{v}$. Then, we use unit vectors to calculate augmented representations for each news sample. The augmented representation, denoted as R^A, is computed as:

$$R^A = R^O + d\boldsymbol{v}. \quad (4)$$

To ensure the intensity of the perturbation remains within a reasonable range, we use the label y as a constraint. Let the target predictor predict the label of each augmented representation of a news sample, represented as $\hat{y}$. From the pool of augmented representations, we aim to select the most demanding one, i.e., the one that lies closest to the decision boundary of the target classifier, while ensuring that $\hat{y} = y$.

3.5 Target Predictor

In this subsection, we describe the training stage of the target predictor. First, we input R^O into the target classifier for prediction as $O^T = F(R^O)$, where $O^T \in \mathbb{R}^L$ denotes the predicted class distribution by target classifier (L is the number of class) and $F(\cdot)$ denotes the target classifier. The objective function for the target predictor combines both the contrastive loss and the cross-entropy loss. The cross-entropy loss ($\mathcal{L}_{CE}$) is defined as follow:

$$\mathcal{L}_{CE} = -\sum_{(R_i^O, y_i) \in \mathcal{S}} CE(\Phi(F(R_i^O), y_i)), \quad (5)$$

where CE denotes cross-entropy loss, $\Phi(\cdot)$ is Softmax. The contrastive loss ($\mathcal{L}_{CL}$) is defined as:

$$\mathcal{L}_{CL} = \frac{-(P_i^O)^T P_i^A}{\|P_i^O\|_2 \|P_i^A\|_2}. \quad (6)$$

here, we adopt a multi-layer projection head to get projection vectors P^O and P^A from original representations R^O and augmented representations R^A. Combining Eq.5, 6, our overall objective function for the main predictor can be written as follows:

$$\underset{\Theta}{argmin}\, \mathcal{L} = \mathcal{L}_{CE} + \alpha \mathcal{L}_{CL}, \quad (7)$$

where Θ denotes parameters of the target encoder and classifier, α denotes the trade-off hyperparameter to balance contrastive loss and classification loss.

3.6 Event-Only Predictor

In this subsection, we describe the training stage of the event-only predictor. To train an Event-Only model that generates predictions driven by label-event spurious correlations, we incorporate an average pooling layer for samples under the same event. We aggregate the origin representation encoded by the event-only encoder of each sample within event e_i as follows:

$$R^E = Mean(\{R'^O_j\}_{j=1}^{m_i}), \quad (8)$$

where R'^O denotes the original representation encoded by event-only encoder, $Mean$ denotes the average pooling, and R^E denotes the event-average representation for each event.

Subsequently, we use R^E as the representation for each sample, inputting it into the event-only

Statistic	Twitter15	Twitter16	PHEME
#Source tweets	1,490	818	6,425
#Events	298	182	9
#Users	276,663	173,487	48,843
#Posts	331,612	204,820	197,852
#Non-rumors	374	205	4,023
#False rumors	370	205	2,402
#Unverified rumors	374	203	-
#True rumors	372	205	-

Table 1: Statistics of the datasets

classifier for prediction. This process yields predictions that are entirely derived from the bias associated with label-event correlations, as $O^E = F'(R^E)$, where $O^E \in \mathbb{R}^L$ denotes the predicted class distribution by the event-only classifier (L is the number of class) and $F'(\cdot)$ denotes the event-only classifier. Then we define the loss function for the event-only predictor as follows:

$$\mathcal{L}_E = -\sum_{(O_i^E, y_i) \in \mathcal{S}} (CE(\Phi(O_i^E, y_i)) \tag{9}$$

Then, our overall objective function for the event-only predictor can be written as $\underset{\theta}{argmin}\, \mathcal{L}_E$, where θ denotes the parameters of the event-only encoder and classifier.

3.7 Debias in inference stage

After the training stage, we have obtained a target predictor capable of making overall predictions O^T using both unbiased and biased features in news pieces, and an event-only predictor that makes predictions O^E merely based on event biases.

To reduce event-label bias, inspired by the Potential Outcomes Model, we subtract O^E from O^T with a bias coefficient β and obtain the debiased output O^D.

$$O^D = O^T - \beta O^E \tag{10}$$

O^D reduces biased predictions and retains unbiased ones, thereby achieving a debiasing effect.

4 Experiments

4.1 Experiment Settings

4.1.1 Dataset

We put our proposed model to the test using three publicly accessible, real-world fake news detection datasets: Twitter15, Twitter16, and PHEME, detailed statistics are shown in Table 1. all of which

Twitter15					
Method	Acc.	U F1	N F1	T F1	F F1
BERT	36.02$\pm$4.80	40.20$\pm$3.00	60.14$\pm$3.30	10.23$\pm$5.80	25.44$\pm$6.50
BiGCN	37.91$\pm$2.58	43.84$\pm$3.75	51.84$\pm$3.77	17.20$\pm$3.14	27.16$\pm$7.04
GACL	54.01$\pm$1.18	56.13$\pm$2.06	<u>88.14$\pm$1.94</u>	13.24$\pm$8.88	38.22$\pm$2.97
PSA-S	<u>59.36$\pm$1.73</u>	**92.35$\pm$0.91**	45.81$\pm$4.10	36.23$\pm$4.69	<u>52.66$\pm$2.97</u>
PSA-M	58.97$\pm$0.87	<u>88.30$\pm$0.56</u>	41.83$\pm$2.62	<u>42.14$\pm$2.08</u>	52.47$\pm$2.03
FADE	**71.81$\pm$2.50**	56.80$\pm$1.44	**92.10$\pm$1.34**	**66.42$\pm$2.17**	**63.68$\pm$1.97**

Table 2: Metrics $\pm$ STD (%) comparison under our experiment setting, averaged over 10 runs. The highest results are highlighted with **bold**, while the second highest results are marked with <u>underline</u>

Twitter16					
Method	Acc.	U F1	N F1	T F1	F F1
BERT	41.87$\pm$5.60	45.00$\pm$3.00	52.00$\pm$5.02	43.00$\pm$3.61	52.00$\pm$5.30
BiGCN	44.29$\pm$1.34	46.86$\pm$2.90	44.81$\pm$2.34	53.76$\pm$4.49	25.43$\pm$2.97
GACL	<u>71.26$\pm$2.18</u>	79.73$\pm$1.76	<u>81.83$\pm$0.93</u>	59.68$\pm$7.36	<u>58.11$\pm$2.68</u>
PSA-S	65.43$\pm$0.95	**95.05$\pm$0.80**	46.66$\pm$1.64	61.22$\pm$1.49	55.62$\pm$2.35
PSA-M	61.47$\pm$1.74	<u>93.91$\pm$0.28</u>	20.97$\pm$8.51	<u>62.21$\pm$1.86</u>	55.08$\pm$3.93
FADE	**77.72$\pm$0.48**	83.06$\pm$2.26	**83.68$\pm$1.35**	**74.14$\pm$2.19**	**63.01$\pm$3.90**

Table 3: Metrics $\pm$ STD (%) comparison under our experiment setting, averaged over 10 runs.

have been gathered from Twitter, one of the most prominent social media platforms in the US. In the three datasets, graph topologies of posts are constructed based on users, sources, and comments. For all three datasets, we employ the pre-training model BERT to generate node embeddings.

4.1.2 Data Splitting

For all three datasets, we adhere to the principle of event separation, ensuring that events do not overlap among the training, testing, and validation sets. Under this constraint, we strive to allocate approximately 10% of the data for validation. The remaining data is then divided into training and test sets, aiming for a 3:1 ratio based on event IDs. Our data splitting for the Twitter15 and Twitter16 datasets is consistent with the split detailed in Wu and Hooi (2022). For the PHEME dataset, we use the same dataset as in Sun et al. (2022), hence we split the dataset ourselves according to the aforementioned ratio. This data splitting ensures that the data in both the test and validation sets belong to unseen events, making it more closely aligned with real-world scenarios.

4.1.3 Compared Methods

We compare with the following baselines:

BERT (Devlin et al., 2018) is a popular pre-trained model that is used for fake news detection.

	PHEME				
Method	Class	Acc.	Prec	Rec	F1
BERT	R	44.05±3.60	62.43±5.45	20.54±4.87	25.52±3.32
	N		45.81±2.03	78.66±8.12	55.28±4.56
BiGCN	R	43.09±4.10	31.44±4.44	34.28±9.84	30.57±6.26
	N		52.01±2.90	49.10±11.63	48.56±7.24
GACL	R	46.21±0.82	75.88±3.24	12.26±4.14	20.76±3.06
	N		42.23±0.66	93.51±2.83	58.01±1.04
PSA-S	R	47.29±1.24	**79.98±2.44**	15.08±2.90	25.23±4.29
	N		43.19±0.53	**94.38±2.90**	**59.26±0.41**
PSA-M	R	48.08±1.20	77.14±2.87	17.88±1.98	29.03±3.13
	N		43.44±1.10	92.25±3.01	59.07±1.01
FADE	R	**60.13±1.41**	76.18±3.02	**52.71±3.79**	**59.52±4.31**
	N		**52.30±1.70**	72.45±3.91	55.98±2.78

Table 4: Metrics ± STD (%) comparison under our experiment setting, averaged over 10 runs.

Model	Twitter15	Twitter16	PHEME
	Acc.	Acc.	Acc.
FADE	**71.81±1.61**	**77.72±0.48**	**60.18±0.89**
FADE w/o ADA	55.66±4.33	53.86±3.90	50.79±3.01
FADE w/o DBI	63.78±2.35	71.70±1.40	53.20±1.35
FADE w/ MUA	61.98±3.12	66.70±2.18	51.39±2.35
FADE w/ ADV	64.01±2.04	70.92±2.14	53.08±1.97
FADE w/ RWT	62.43±2.36	71.01±3.02	51.14±2.67

Table 5: Accuracy ± STD (%) comparison of ablation study on the Twitter15, Twitter16 and PHEME, averaged over 10 runs

BiGCN (Bian et al., 2020) is a GCN-based model that uses the two key features of news propagation and dispersion to capture the global structure of the news tree.

GACL (Sun et al., 2022) is a GCN-based model using adversarial and contrastive learning for fake news detection.

PSA (Wu and Hooi, 2022) is a text-based fake news classifier that can learn writing style and truth stance, thus enhancing its classification capability. **PSA-S** and **PSA-M** respectively represent the use of sum and mean as pooling functions.

FADE is our proposed framework.

4.1.4 Implementation Details.

We implement our FADE framework and other baselines using PyTorch with CUDA 12.0 on an Ubuntu 20.04 server with NVIDIA RTX 3090 GPU and an AMD EPYC 7763 CPU. For optimization, we use Adam optimizers, with a learning rate of 0.001 across all datasets. Batch sizes are set at 510 for Twitter16, 3851 for PHEME, and 992 for Twitter15. Trade-off hyper-parameters are 10.0 for Twitter15 and Twitter16 and 1.0 for PHEME and the bias coefficient is 0.1 for all three datasets.

4.2 Result and Discussion

To ensure a fair comparison, we adopt the same evaluation criteria as GACL. We adopt the Accuracy (Acc.), Precision (Prec.), Recall (Rec.), and F1-measure (F1) as our evaluation metrics. Table 2,3,4 showcase the performance of all comparison methods on three public real-world datasets following our event-separated data split criteria.

The BERT model, based on a self-attention mechanism, yields the poorest results. The GCN-based models BiGCN and GACL, designed for event-mixed detection, experience significant performance declines in the event-separated setting. BiGCN, focusing on bottom-up and top-down structures, achieves only 41.76% average accuracy across the datasets. Despite its AFT module aimed at enhancing robustness, GACL also struggles in event-separated detection. PSA, specifically developed for event-separated detection, underperforms as well due to its exclusive reliance on textual content and overlooking news propagation structure.

The FADE proposed in this paper outperforms all other compared methods, on all three datasets. Compared to the current best-performing methods, FADE has shown an improvement in accuracy by **12.45%** on Twitter15, **6.46%** on Twitter16, and **12.05%** on the PHEME dataset. The superiority of FADE stems from three reasons: (1) Our adaptive augmentation strategy generates superior augmented samples compared to other manually designed augmentation. These high-quality samples, enhanced through graph contrastive learning, significantly improve the model's classification performance and robustness. (2) In situations where unbiased predictions cannot be directly obtained, we indirectly mitigate the impact of event bias on predictions by subtracting the event-only output, which is derived directly from biases, from the target output that integrates both biased and unbiased features. This effectively alleviates the influence of event bias, enhancing the framework's generalization performance. (3) We leverage the advanced pre-training model, BERT, to generate embeddings.

4.3 Ablation Study

This section evaluates the impact of each module in our study through ablation experiments.

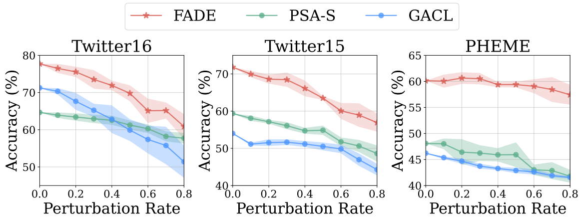

Figure 4: Accuracy ± STD (%) of perturbation experiments on the Twitter15, Twitter16, and PHEME datasets with different data perturbation rates (r), averaged over 10 random perturbation processes.

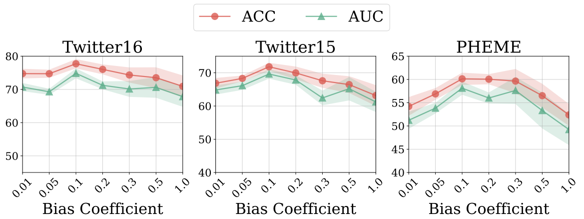

Figure 5: Metrics ± STD (%) results of Hyperparameter Analysis on the Twitter15, Twitter16 and PHEME with different bias coefficients (β)

FADE w/o ADA omits the adaptive augmentation module and the contrastive learning loss, solely utilizing classification loss for training.

FADE w/o DBI removes the step of utilizing the event-only predictor for debiasing during the inference phase.

FADE w/ MUA indicates replacing the adaptive augmentation in the FADE framework with a manually selected augmentation strategy.

FADE w/ ADV denotes switching the debiasing method in FADE to adversarial debiasing.

FADE w/ RWT signifies replacing the debiasing approach in FADE with reweighting debiasing.

Table 5 shows the results of the ablation study. The removal of adaptive graph augmentation in FADE w/o ADA and the removal of the debiasing module in FADE w/o DBI both result in a notable performance drop. In FADE w/ MUA, we replace the adaptive augmentation strategy in FADE with a manually selected augmentation strategy and choose the optimal intensity. However, the performance achieved is far below that of the complete FADE. In FADE w/ ADV and FADE w/ RWT, we respectively replace the debiasing strategy in FADE with adversarial debiasing and reweighting debiasing. However, these two debiasing strategies fail to effectively reduce bias and even result in a certain degree of performance degradation. The above experimental results demonstrate the effectiveness of the two modules in the FADE framework.

4.4 Perturbation Experiments

In this section, we assess the robustness of **FADE** through experiments using perturbed graphs and compare its performance with GACL and PSA-S. GACL and PSA-S were selected for comparison due to their exceptional performance.

We employed two perturbation methods: edge perturbation and node feature masking, which simulate the structural and feature variations that news might have under different events in social media. The perturbation rate, denoted by r, quantifies the intensity of these perturbations.

Results in Figure 4 reveal that FADE outperforms the other models across different perturbation intensities. With disturbances up to 30%, FADE's accuracy remains stable, dropping by less than 4% on Twitter16, under 2% on Twitter15, and 1% on PHEME. Impressively, even when 80% of edges and node features are altered, FADE still achieves 60.86% accuracy on Twitter16, 56.92% on Twitter15, and 57.43% on PHEME. This affirms FADE's robustness against variations in news propagation structures and feature distributions.

4.5 Hyperparameter Analysis

In this section, we analyze the impact of the hyperparameter bias coefficient (β) on model performance. As illustrated in Figure 5, the optimal performance on all three datasets is achieved when the target predictions and event-only predictions are combined with an intensity of 0.1.

5 Conclusion

In this paper, we analyze how event-separated data splitting more closely aligns with real-world social media fake news detection tasks. Then, we demonstrate that current state-of-the-art methods are ineffective in detecting fake news within unseen events. To better address this task, we propose a social media fake news detection framework, FADE, which exhibits sufficient robustness and generalizability when dealing with dynamic and ever-changing events on social media. Specifically, we first trained a robust target predictor using adaptive augmentation and graph contrastive learning. Then, we combined this with an independently trained event-only predictor for further debiasing during the inference stage. Experiments demonstrate that FADE outperforms existing methods on three real-world fake news detection datasets.

6 Limitations

In this part, we discuss two limitations of our work.

Firstly, some events possess only faint bias signatures, making it challenging for our event-only predictor to yield substantially biased predictions in these scenarios. This limitation means that during debiasing, subtracting these weak predictions might not significantly mitigate bias. Instead, it risks omitting valuable information from the target predictions. We leave the task of addressing debiasing under varying levels of bias as an area for future work.

Secondly, the field of Large Language Models (LLMs) like GPT-4 has seen rapid advancement in the past year. These models have demonstrated formidable capabilities in understanding context, generating coherent and relevant text, and even exhibiting a form of reasoning. However, a limitation of our current method is that it doesn't harness these state-of-the-art LLMs to enhance feature quality or assist in predictions. Recognizing the potential of these developments, we aim to integrate LLMs into our future work on fake news detection, leveraging their advanced capabilities to further enhance our approach.

7 Ethics Statement

This article focuses on Twitter social media data. We use publicly available benchmark datasets for classification, which comply with Twitter's regulations and were extracted using the official API. To ensure user privacy and data security, all dataset-related tweets were anonymized and URLs removed. Our research aims to analyze rumor detection methods to enhance information credibility on social media. Experimental results will be reported objectively and transparently, adhering to academic and ethical standards.

References

Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 549–556.

Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems*, 41(3):1–39.

Enyan Dai and Suhang Wang. 2021. Say no to the discrimination: Learning fair graph neural networks with limited sensitive attribute information. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 680–688.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

Masahiro Kaneko and Danushka Bollegala. 2019. Gender-preserving debiasing for pre-trained word embeddings. *arXiv preprint arXiv:1906.00742*.

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. 2019. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Qiang Liu, Feng Yu, Shu Wu, and Liang Wang. 2018. Mining significant microblogs for misinformation identification: an attention-based approach. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 9(5):1–20.

Guanghui Ma, Chunming Hu, Ling Ge, Junfan Chen, Hong Zhang, and Richong Zhang. 2022. Towards robust false information detection on social networks with contrastive learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 1441–1450.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor detection on twitter with tree-structured recursive neural networks. Association for Computational Linguistics.

Nitin Madnani and Bonnie J Dorr. 2010. Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, 36(3):341–387.

Nicholas Meade, Elinor Poole-Dayan, and Siva Reddy. 2021. An empirical survey of the effectiveness of debiasing techniques for pre-trained language models. *arXiv preprint arXiv:2110.08527*.

Junghyun Min, R Thomas McCoy, Dipanjan Das, Emily Pitler, and Tal Linzen. 2020. Syntactic data augmentation increases robustness to inference heuristics. *arXiv preprint arXiv:2004.11999*.

Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. 2019. Dropedge: Towards deep graph convolutional networks on node classification. *arXiv preprint arXiv:1907.10903*.

Ikuro Sato, Hiroki Nishimura, and Kensuke Yokoi. 2015. Apac: Augmented pattern classification with neural networks. *arXiv preprint arXiv:1505.03229*.

Tal Schuster, Darsh J Shah, Yun Jie Serene Yeo, Daniel Filizzola, Enrico Santus, and Regina Barzilay. 2019. Towards debiasing fact verification models. *arXiv preprint arXiv:1908.05267*.

Jasjeet Sekhon. 2008. The neyman—rubin model of causal inference and estimation via matching methods.

Patrice Y Simard, David Steinkraus, John C Platt, et al. 2003. Best practices for convolutional neural networks applied to visual document analysis. In *Icdar*, volume 3. Edinburgh.

Krishna Kumar Singh, Hao Yu, Aron Sarmasi, Gautam Pradeep, and Yong Jae Lee. 2018. Hide-and-seek: A data augmentation technique for weakly-supervised localization and beyond. *arXiv preprint arXiv:1811.02545*.

Tiening Sun, Zhong Qian, Sujun Dong, Peifeng Li, and Qiaoming Zhu. 2022. Rumor detection on social media with graph adversarial contrastive learning. In *Proceedings of the ACM Web Conference 2022*, pages 2789–2797.

William Yang Wang and Diyi Yang. 2015. That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using# petpeeve tweets. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2557–2563.

Yiwei Wang, Wei Wang, Yuxuan Liang, Yujun Cai, and Bryan Hooi. 2020. Graphcrop: Subgraph cropping for graph classification. *arXiv preprint arXiv:2009.10564*.

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196*.

Jiaying Wu and Bryan Hooi. 2022. Probing spurious correlations in popular event-based rumor detection benchmarks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 274–290. Springer.

Weizhi Xu, Qiang Liu, Shu Wu, and Liang Wang. 2023. Counterfactual debiasing for fact verification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6777–6789.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823.

Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2019. Attention-based convolutional approach for misinformation identification from massive and noisy microblog posts. *computers & security*, 83:106–121.

Feng Yu, Qiang Liu, Shu Wu, Liang Wang, Tieniu Tan, et al. 2017. A convolutional approach for misinformation identification. In *IJCAI*, pages 3901–3907.

Yanqiao Zhu, Yichen Xu, Qiang Liu, and Shu Wu. 2021. An empirical study of graph contrastive learning. *arXiv preprint arXiv:2109.01116*.

Yongchun Zhu, Qiang Sheng, Juan Cao, Shuokai Li, Danding Wang, and Fuzhen Zhuang. 2022. Generalizing to the future: Mitigating entity bias in fake news detection. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2120–2125.

A Dataset Events Statistics

As illustrated in Figure 6, severe event-label spurious correlations exist in the Twitter16 and Twitter15 datasets. While large-size events encompass more than 70% of samples in Twitter15 and Twitter16 datasets, each event's samples invariably have the same class label. Meanwhile, the PHEME dataset, comprising only 9 events, does not consistently feature news with the same label within each event. However, it still exhibits a strong tendency for keyword-sharing.

B Class Imbalance

Overall, existing methods appear to be inadequate when facing event-separated fake news detection, and there is a significant class imbalance in their detection capabilities across different categories. For instance, PSA-M on the Twitter16 dataset shows a stark disparity in F1 scores for Unverified news and Non-rumors categories, at 93.91% and 20.97% respectively. This vast difference indicates that the model has a severe bias towards different categories of news. We leave the exploration of this aspect for future work.

C Propagation Structures Analysis

Figure 7 shows that the news in the top 10 events of the Twitter15 dataset have vastly different propagation structures. They exhibit significant variations

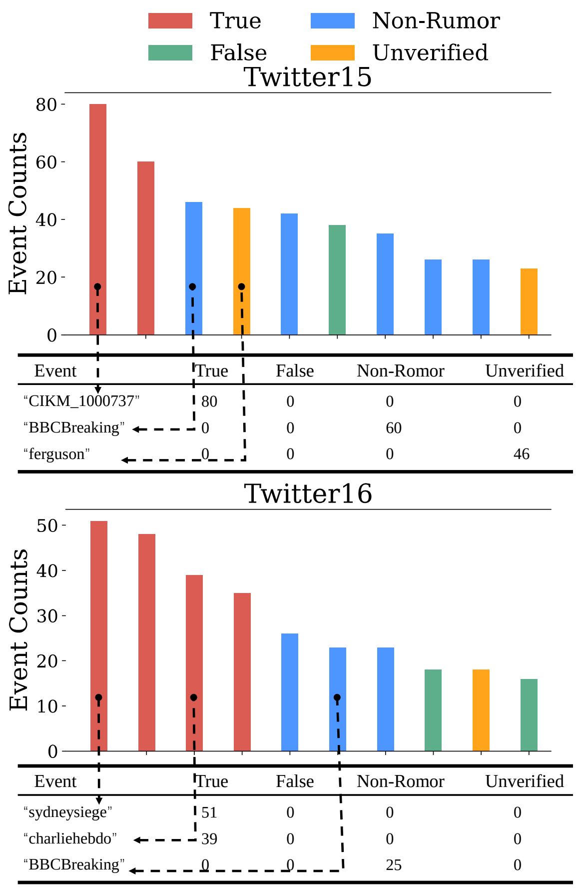

Figure 6: The size of largest events in Twitter15 and Twitter16 datasets, most event labels directly correlate with class labels, which shows strong event bias in fake news detection datasets.

in both their average depth and the average proportion of edges directly connected to the root node in relation to the total number of edges. Additionally, events with a shallower average depth tend to have stronger node centrality.

D Ablation Study Details

In the FADE w/ MUA experiment, we selected three augmentation methods in our designed enhancement strategy: random node dropping, perturbing edges, and feature masking. After repeated experiments, the optimal augmentation intensity used was 0.15.

In the FADE w/ ADV experiment, we replaced our debiasing method with the adversarial debiasing approach designed according to reference Dai and Wang (2021). Specifically, we set up a discriminator f_D to judge the event labels of the news, training it with an objective function as Eq.11. Subsequently, we conducted adversarial training of the encoder using the objective function in Eq.12.

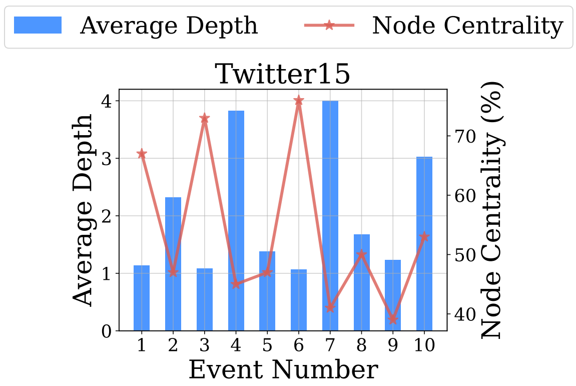

Figure 7: The average depth and the average proportion of edges directly connected to the root node in relation to the total number of edges (%) of the top 10 events.

However, due to the excessive number of event categories in the fake news dataset, the adversarial training was ineffective in reducing bias.

$$\min_{\theta_D} \mathcal{L}_S = -\sum_{(G_i,e_i)\in\mathcal{S}} (CE(\Phi(f_D(G_i), e_i)) \tag{11}$$

where θ_D denotes the parameters of the discriminator.

$$\begin{aligned}\min_{\theta_E,\theta_C} \mathcal{L} = &\sum_{(R_i^O,y_i)\in\mathcal{S}} CE(\Phi(F(R_i^O), y_i)) \\ &- \sum_{(G_i,e_i)\in\mathcal{S}} CE(\Phi(f_D(G_i), e_i))\end{aligned} \tag{12}$$

where θ_E denotes the parameters of the target encoder, θ_C denotes the parameters of the target classifier.

In the FADE w/ RWT experiment, we calculated the weight of each sample according to the method described in Eq.13.

$$w(R_i) = \frac{1}{\sum_{k=1}^{s} \frac{\mathbb{1}(F(R_i)=y_i)}{s} + \gamma} \tag{13}$$

where s denotes event size, and γ is a scale hyperparameter.