Unseen Domain Fake News Detection via Causal Propagation Substructures

Anonymous ACL submission

Abstract

The spread of fake news through social media poses significant threats. Recent models using text and graph features have shown promising results in specific fake news detection scenarios. However, these data-driven models heavily rely on training data that share similar distribution with inference data, limiting their applicability to fake news from emerging or previously unseen domains, known as *out-of-distribution* (OOD) data. Tackling OOD fake news is a challenging yet critical task. To address the challenge, we propose the **C**ausal **S**ubgraph-oriented **D**omain **A**daptive Fake News Detection (CSDA) model. CSDA extracts causal substructures from news propagation graphs that generalise to OOD data, using a graph neural network-based mask generation process. It uses refined training objectives to ensure high-quality subgraphs. It is further powered by contrastive learning for few-shot scenarios, where a limited amount of OOD data is available for training. Extensive experiments on public social media datasets demonstrate the effectiveness of CSDA effectively handles OOD fake news detection, achieving a 1.23%∼12.23% accuracy improvement over other state-of-the-art models.

1 Introduction

The popularity of social media has enabled rapid news dissemination, for both true and fake news. Given the potential impact of fake news, robust methods are urgently needed to debunk such misinformation in a timely manner. In real-world scenarios, out-of-distribution news from unseen domains emerges over time. This brings substantial challenges to fake news detection models.

Graph-based fake news detection methods using graph neural networks (GNN) have garnered much attention recently for modelling news propagation patterns (Gong et al., 2023a). Despite their success, existing GNN-based methods are generally built upon the assumption that both training and testing data are independently sampled from an identical data distribution (i.i.d.), which often does not hold true nor reflect the real challenges of fake news detection in practical scenarios (Li et al., 2022). Emerging and hitherto unseen fake news and their associated propagation graphs can and do arise. From an empirical perspective, these methods focus on minimising the average training errors and incorporating correlations within the training data (which is considered to be *in-distribution*) to improve fake news detection accuracy (Liu et al., 2021). However, real-world graph-based fake news data is often mixed with biased domain-specific information in the training data. For example, Zhang et al. observed that the veracity of some political news is strongly correlated with specific keywords (e.g., most news propagation involving the terms "White House" and "rainbow" are classified as true). The detection models trained from such data may thus learn domain-specific biases resulting in poor generalisation (Li et al., 2022).

To detect fake news across different domains (e.g., sports and politics), early studies (Ma et al., 2018; Bian et al., 2020) focused on capturing content-independent propagation patterns. However, it has been shown (Min et al., 2022) that not only the news content but also the propagation pat-

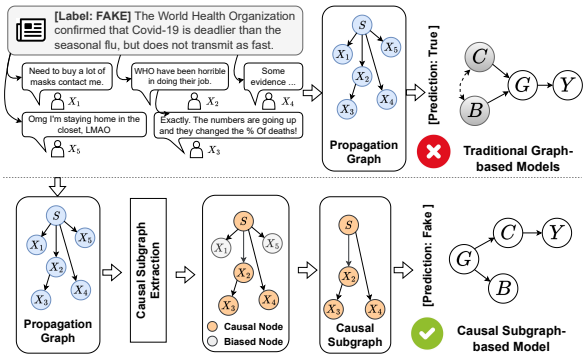


Figure 1: Illustration of causal subgraphs and our casual subgraph-based model (bottom).

terns can vary across different domains. Such cross-domain fake news detection method have limited success. To tackle this, recent methods (Lin et al., 2022; Li et al., 2023) utilise domain adaptation to transfer trained models to emerging news domains with a small amount of data from the emerging domains. However, these methods require labelled data from emerging domains which is typically unavailable. More related works are in Appendix.B.

To address these limitations, we focus on extracting causal subgraphs from news propagation graphs whilst aiming to mitigate domain biases for fake news detection in emerging domains. News from an emerging domain is treated as *out-of-distribution* (OOD) data when its distribution shifts significantly from the in-distribution data. Our model generalises to OOD data by capturing causal subgraphs in an unsupervised manner. From a causal analysis standpoint, each propagation graph comprises a causal subgraph and a domain biased subgraph, which are initially entangled. Our key insight is that not all nodes in propagation graphs contribute to generalisation across unseen domains. Instead, only the causal subgraphs carry critical information should be used for identifying fake news in OOD settings, as illustrated in Fig 1. By identifying and capturing these causal subgraphs, we can enhance the model's generalisation capabilities.

Based on this intuition, our model, the **C**ausal **S**ubgraph Oriented **D**omain **A**daptive Fake News Detection model (CSDA), is proposed. CSDA extracts subgraphs from propagation graphs and performs classification based on particular subgraphs, which are referred to as the *casual subgraphs*. In CSDA, a binary mask is learned for each node and edge of the propagation graph to classify them into *causal* or *biased* elements. For the subgraph formed by each type of element, a graph encoder and a multilayer perceptron (MLP) classifier together encode the subgraphs and classify the news items according to the subgraph embeddings. The classifier predictions of the causal elements are regarded as the domain-invariant predictions. For the biased elements, they are expected to be biased to the training data's domains and harm the generalisation to unseen domains. Since the subgraph of the biased elements can be biased to specific domains, in the training process, we design training objectives to optimise the subgraph split whilst restricting the influence of the biased elements.

Following recent works such as (Lin et al., 2022; Li et al., 2023), we also consider a scenario where limited OOD data becomes available, e.g., through manual labelling. In this scenario, CSDA's performance is further enhanced with a supervised contrastive learning-based approach and achieves state-of-the-art (SOTA) classification accuracy.

In summary, our contributions include:

- We propose a zero-shot unseen domain fake news detection model named CSDA based on domain-invariant causal subgraphs from news propagation patterns.
- We further explore a few-shot scenario where a small number of OOD examples are available, and utilise contrastive learning to enhance CSDA's cross-domain fake news detection performance.
- Extensive experiments are conducted on four real datasets. The results confirm the effectiveness of CSDA for cross-domain fake news detection, outperforming SOTA models by 1.23%$\sim$12.23% in terms of accuracy.

2 Preliminaries

Unseen domain fake news detection aims to transfer a model trained on a labelled (in-distribution) dataset to an OOD dataset that is unlabelled or with a few labelled samples.

Given a set of news items $D_{in} = \{(\mathcal{G}_k^{in}, y_k^{in})\}$ ($k \in [1, n_{in}]$) that comes from some latent distribution $\mathcal{P}$, we aim to train a model to detect fake news in another dataset $D_{out} = \{(\mathcal{G}_k^{out})\}$ ($k \in [1, n_{out}]$) that contains data from an unknown distribution $\mathcal{P}'$ that is different from $\mathcal{P}$. Here, we refer to data D_{in} from $\mathcal{P}$ as in-distribution data and those D_{out} from $\mathcal{P}'$ as OOD data. n_{in} and n_{out} refer to the number of news items in D_{in} and D_{out}, respectively. Our goal is to train a classifier f using the training set D_{in} to determine whether news items in another non-overlapping set D_{out} contains fake news. We assume that both D_{in} and D_{out} share the same label space (i.e. they are labelled as either true news and fake news).

Causal Analysis As shown in Fig 1, we use variables C, B, G and Y to represent the casual subgraph (C), the biased subgraph (B), the observed propagation graph (G), and the news label (Y), according to the recent advances in causal invariant learning (Fan et al., 2022a; Chen et al., 2022). Each link denotes a causal relationship (Fan et al., 2022b). With traditional graph-based models, the

propagation graphs are encoded directly, and hence spurious correlations between C and B are ignored and fused into the graph embedding, leading to inaccurate fake news predictions. In our approach, the causal and biased subgraphs are disentangled, and the prediction improved by referring solely to the causal information.

Data Preparation For each news item from both D_{in} and D_{out}, the propagation graph $\mathcal{G}_k = \langle \mathbf{X}_k, \mathbf{A}_k \rangle$ is extracted and modelled as an undirected acyclic graph. The node set $\mathbf{X}_k = \{x_1, x_2, \ldots, x_{|\mathbf{X}_k|}\}$ contains all posts including the source news posts and all associated comments/reposts which can be used to provide supportive information regarding the post veracity. Each post's embedding is initialised using a pretrained BERT model (Devlin et al., 2019) to compute the text embeddings.

The adjacency matrix $\mathbf{A}_k = \{\alpha_{ij}, i, j \in [1, |C_k|]\}$ is the set of propagation behaviours where an edge exists (i.e., $\alpha_{ij} = 1$) between node i and node j if there is a reply/repost relationship.

3 Proposed Model

In the section, we detail the CSDA model used for unseen fake news detection tasks. CSDA is designed to extract and capitalise on subgraphs from the news propagation graph. The architecture of CSDA is illustrated in Fig 2.

In CSDA, we take a small batch of propagation graphs and apply a mask generator on them to split each propagation graph into a causal subgraph and a biased subgraph. The causal and the biased subgraphs are encoded using two individual graph encoders to produce two separate embeddings. The training objective is to emphasise the impact of the casual subgraphs while eliminating the impact of the biased subgraphs on the classification output.

CSDA is trained on D_{in} and tested on D_{out} in a zero-shot manner. When a few labelled samples are available from D_{out}, they can also be incorporated into the training process to further enhance the model performance on D_{out}.

3.1 Mask Generator

Our mask generator learns a mask to help split each propagation graph $\mathcal{G}$ (i.e., $\mathcal{G}_k$ – now we further drop the subscript 'k' as long as the context is clear) into a causal subgraph $\mathcal{G}_c$ and a biased subgraph $\mathcal{G}_b$. This is achieved by computing node importance scores (denoted as α_i for node i) and edge importance scores (denoted as β_{ij} for the edge between nodes i and j) in the propagation graph $\mathcal{G}$. The aim is to measure the probability of a node or an edge belonging to a causal subgraph.

The mask generator takes graph $\mathcal{G}$ (i.e., its features) as input and outputs the importance of its nodes and edges. A Graph Isomorphism Network (GIN) (Xu et al., 2018) is utilised to encode the graph and map the node features $\mathbf{X}$ to node embeddings $\mathcal{H}$ for the model's graph structure learning capability. After obtaining the graph features $\mathcal{H} = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_N\}$, where N is the size of the node set and $\mathbf{h}_i$ represents the embedding for the i-th node, the node and edge importance scores are computed using an MLP:

$$\alpha_i = \sigma(\text{MLP}([\mathbf{h}_i])), \beta_{ij} = \sigma(\text{MLP}([\mathbf{h}_i, \mathbf{h}_j])). \tag{1}$$

where σ is the activation function.

Since the causal and the biased subgraphs are defined as two non-overlapping substructures of $\mathcal{G}$, the probability of a node and an edge belonging to a biased subgraph can be established by $(1 - \alpha_i)$ and $(1 - \beta_{ij})$, respectively.

Using the importance scores, we construct a causal graph mask $\mathbf{M}_c = [\alpha, \beta]$ and a biased graph mask $\mathbf{M}_b = [(1 - \alpha), (1 - \beta)]$. Finally, the input propagation graph $\mathcal{G}$ is decomposed into a causal subgraph $\mathcal{G}_c = \{\mathbf{M}_c \odot \mathcal{G}\}$ and a biased subgraph $\mathcal{G}_b = \{\mathbf{M}_b \odot \mathcal{G}\}$, where $\odot$ is the filtering operation on graph $\mathcal{G}$ with the corresponding masks. The masks emphasise distinct regions of the propagation graphs, enabling subsequent GNN-based graph encoders to concentrate on different segments of the graphs.

3.2 Graph Encoder

Two subgraph encoders realised as a 2-layer of stacked GCNII (Chen et al., 2020) are used to encode the causal and the biased subgraphs. Given a graph's node features $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ and its adjacency matrix $\mathbf{A}$, the graph embeddings are computed through GCNII by:

$$\mathcal{Z}^{(l+1)} = \sigma\left((\tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}\mathcal{Z}^{(l)} + \mathcal{Z}^{(0)})(\mathbf{I}_n + \mathbf{W}^{(l)})\right), \tag{2}$$

where $l = 0$ or 1, $\mathcal{Z}^{(0)}$ is the initial node features $\mathbf{X}$, $\tilde{\mathbf{A}}$ is the adjacent matrix of the graph with self-loops, $\tilde{\mathbf{D}}$ is the degree matrix of $\tilde{\mathbf{A}}$, $\mathbf{I_n}$ is the identity mappig from GCNII, $\mathbf{W}^{(l)}$ is the learnable parameter matrix, and σ is the activation function. $\mathcal{Z}^{(0)}$ is initiated as the node feature input $\mathbf{X}$.

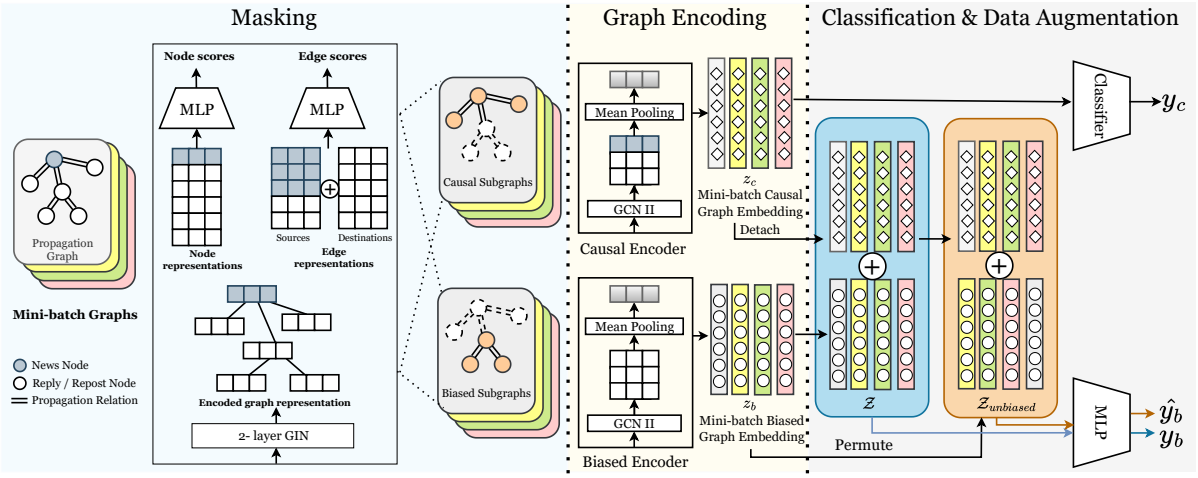


Figure 2: Architecture of CSDA, which is trained with batches of news propagation graphs. A mini-batch of propagation graphs are masked by the Mask Generator and divided into causal and biased subgraphs. Then, the two branches of subgraphs are encoded using two independent graph encoders to produce causal and biased embeddings. Afterwards, the causal embedding is forwarded to an MLP classifier for causal veracity prediction. Meanwhile, the biased embedding is utilised as part of model optimisation for accurate subgraph extraction.

As shown in Fig 2, two parallel subgraph encoders are used to encode the causal subgraph $\mathcal{G}_c$ and the biased subgraph $\mathcal{G}_b$ into a causal embedding $\mathbf{z}_c$ and a biased embedding $\mathbf{z}_b$. These embeddings will subsequently be fed into news classifiers for loss calculation and fake news prediction.

3.3 Classification Module

The classification module (CM) is responsible for predicting the news veracity based on the extracted graph embeddings. It is composed of an MLP that uses a softmax function. Given the graph embedding $\mathcal{Z}$, e.g. the causal graph embedding $\mathbf{z}_c$, the CM computes the prediction through:

$$pred = softmax(\text{MLP}(\mathcal{Z})). \tag{3}$$

Since CSDA focuses on classifying news according to causal features, we design a causal CM and a biased CM in the model. They do not share the parameters and have different input dimensions according to model design. During model training, these two CMs are jointly trained to optimise CSDA to capture causal information accurately. For model inference, only the prediction results from the causal CM are used to detect fake news. More details about the use of the outputs of these two CMs are presented in the next subsection.

3.4 Disentangling Training Objectives

As described, the causal and the biased subgraphs are initially entangled. If a model is optimised only by the prediction results, the model could be trained to extract trivial patterns (e.g. treating the whole graph as the causal subgraph) leading to sub-optimal generalisation outcomes. To disentangle the causal and the biased subgraphs while optimising the prediction results, contrastive learning, data augmentation and hinge loss are utilised in addition to traditional cross-entropy loss.

Loss of the Causal Model Branch The cross-entropy of the causal prediction is established by:

$$\mathcal{L}_{ce}^{c} = CE(y_c, y), \tag{4}$$

where y_c and y are the causal prediction result and the ground truth labels. Training with L_{ce}^{c} only cannot guarantee the accurate extraction of the causal and biased subgraphs.

To enhance the quality of the extracted causal subgraphs, a contrastive loss is introduced due to its data distribution capabilities. The intuition is that since the causal subgraph is based on the direct causality of news labels, the data samples that share the same labels tend to have similar causal subgraph embeddings. Therefore, for two samples n and m sharing the same label in a batch, a contrastive loss can be defined based on:

$$\mathcal{L}_{CL}^{in} = -\frac{1}{N^{in}} \sum_{n=1}^{N^{in}} \frac{1}{N_{y_n^{in}}} \sum_{m=1}^{N^{in}} \mathbb{1}_{[n \neq m]} \mathbb{1}_{[y_n^{in} = y_m^{in}]} \log \frac{\exp(sim(o_n^{in}, o_m^{in})/\tau)}{\sum_{k=1}^{N^{in}} \mathbb{1}_{[n \neq k]} \exp(sim(o_n^{in}, o_k^{in})/\tau)}, \tag{5}$$

where N^{in} is the number of in-distribution data samples in a batch, $N_{y_n^{in}}$ is the number of in-distribution data samples which share the same label y_n^{in} with sample C_n^{in}, $\mathbb{1}$ is the indicator function, o_n^{in}, o_m^{in}, and o_k^{in} are the corresponding extracted casual representations from CSDA, $sim(\cdot)$ is the cosine similarity function, and τ is a hyperparameter that controls the temperature.

The contrastive loss enhances the quality of the extracted causal subgraph. However, the model could still predict the whole input graph as the causal subgraph. Therefore, the biased subgraph also needs to be optimised.

Loss of the Biased Model Branch To emphasise the causal subgraph and restrict the biased subgraph, we allow the causal subgraph to include the biased subgraph, but not versa. To achieve this, we create two embeddings $\mathcal{Z}$ and $\hat{\mathcal{Z}}$ by concatenating the causal subgraph embedding $\mathbf{z}_c$ and the biased subgraph embedding $\mathbf{z}_b$. To prevent back-propagation from the biased classification loss from affecting the causal model branch, the gradients from the causal embedding is detached before concatenation, as shown in Fig. 2. Then we forward the $\mathcal{Z}$ to the biased classifier to output prediction y_b.

In addition, the biased graph embedding $\mathbf{z}_b$ is permuted and concatenated with the causal graph embedding $\mathbf{z}_c$ to form $\hat{\mathcal{Z}}$. The labels y are also permuted to $\hat{y}$ in the same order as $\mathbf{z}_b$ to encourage the biased classification module to focus only on the biased embeddings. The output of this module is $\hat{y_b}$.

Lastly, y_b is the prediction on both of the causal and biased subgraphs, $\hat{y_b}$ is the prediction only on biased subgraphs. Inspired by (Chen et al., 2022), a restricted hinge loss (Crammer and Singer, 2001) is utilised to ensure that the biased subgraph does not include the causal subgraphs:

$$\mathcal{L}_{ce}^{b} = \frac{1}{N} CE(\hat{y}_b, \hat{y})\, \mathbb{1}[CE(y_b, y) \leq CE(\hat{y}_b, \hat{y})], \tag{6}$$

Here, $\mathbb{1}[\cdot]$ denotes the indicator function, which outputs 1 when the specified condition is satisfied. The hinge loss is designed to back-propagate only when the predictions based on both causal subgraphs and biased subgraphs are no worse (i.e., result in a lower or equal loss) than the predictions based solely on the biased subgraphs.

The overall loss $\mathcal{L}$ of CSDA is given as the sum of the loss terms above:

$$\mathcal{L} = \mathcal{L}_{ce}^{c} + \mathcal{L}_{ce}^{b} + \gamma \cdot \mathcal{L}_{CL}^{in}, \tag{7}$$

where γ is a hyperparameter to adjust the contrastive loss.

3.5 Model Fine-tuning with OOD Data

CSDA can be trained using just in-distribution data D_{in}. We can also use a few labelled OOD samples to further fine-tune CSDA on a target domain. In this subsection, we discuss model optimisation given a few OOD samples using contrastive learning.

When OOD samples are available, we can further improve the model performance via aligning the representation space of the data. This is achieved by bringing the representations of in-distribution and OOD samples from the same veracity class together while keeping representations from different classes apart. We use contrastive learning for this purpose.

To fine-tune CSDA with OOD data, another supervised contrastive learning objective is proposed. Here, we aim to draw the embedding space of samples with the same label but from different distributions closer based on:

$$\mathcal{L}_{CL}^{out} = -\frac{1}{N^{out}} \sum_{n=1}^{N^{out}} \frac{1}{N_{y_n^{out}}} \sum_{m=1}^{N^{in}} \mathbb{1}_{[y_n^{out}=y_m^{in}]} \cdot \log \frac{\exp(sim(o_n^{out}, o_m^{in})/\tau)}{\sum_{k=1}^{N^{in}} \exp(sim(o_n^{out}, o_k^{in})/\tau)} \tag{8}$$

where N^{out} is the number of OOD samples in a training batch, N^{in} is the number of in-distribution samples in the batch, $N_{y_n^{out}}$ is the number of in-distribution samples which share the same label y_n^{out} with sample C_n^{out}, and o_n^{out}, o_m^{in}, and o_k^{in} are the corresponding extracted causal representations from CSDA, respectively.

The overall loss $\mathcal{L}'$ of CSDA now becomes:

$$\mathcal{L}' = \mathcal{L}_{ce}^{c} + \mathcal{L}_{ce}^{b} + \gamma \cdot (\mathcal{L}_{CL}^{in} + \mathcal{L}_{CL}^{out}) \tag{9}$$

, where γ is the same hyperparameter used in Equation. 7.

4 Experiment

4.1 Experimental Settings

Datasets Four public datasets collected from Twitter (now called X) and Weibo (a Chinese social media platform like Twitter) are

utilised in the experiments: (1) Twitter (Ma et al., 2017), (2) Weibo (Ma et al., 2016), (3) Twitter-COVID19 (Lin et al., 2022) and (4) Weibo-COVID19 (Lin et al., 2022). The statistics of the datasets are shown in Appendix. D, Table 4. Twitter and Weibo are open-domain datasets. They cover a variety of topics except COVID-19 and are used as the main training set. Twitter-COVID19 and Weibo-COVID19 only contain news related to COVID-19, which represent the OOD data.

To showcase the effectiveness of CSDA, two sets of experiments are designed. In the first set of experiments, the models are trained on in-distribution data (e.g., Twitter) and tested on OOD data (e.g., Twitter-COVID19), to simulate the scenario where no prior knowledge about the OOD data is available. In the second set of experiments, a few OOD samples (e.g., 20% of Twitter-COVID19) are utilised to help optimise the models together with in-distribution data (e.g., Twitter), to simulate the scenario where we have a small number of manually labelled OOD samples. The remaining OOD data (e.g, 80% of Twitter-COVID19) are used for model testing.

Baselines We compare with 14 models including recent models UCD-RD (Ran and Jia, 2023), CADA (Li et al., 2023), DELL (Wan et al., 2024) and graph OOD generalisation methods G-mixup (Han et al., 2022), CIGA (Chen et al., 2022).

Baseline models that are trained with in-distribution data only include: **LSTM** (Ma et al., 2016) which uses an LSTM-based model to learn feature representations of relevant posts over time;**CNN** (Yu et al., 2017) uses a CNN model for misinformation identification by modelling the relevant posts as a fixed-length sequence; **RvNN** (Ma et al., 2018) which learns the propagation of news by exploiting a tree structured recursive neural network; **PLAN** (Khoo et al., 2020) which uses a Transformer (Vaswani et al., 2017)-based model for fake news detection by capturing long-distance interactions between tweets (source posts and associated comments); **RoBERTa** (Liu et al., 2019) which encodes the text information of a news item and classifies the news based on the text classification; **BiGCN** (Bian et al., 2020) which models news propagation by representing social media posts as nodes in a graph, and then it utilises a GCN-based model to encode the graph and classifies whether a given news item is true or fake; **GACL** (Sun et al., 2022) which enhances BiGCN by generating adversarial training samples and training with contrastive learning; **SEAGEN** (Gong et al., 2023b) which models the news propagation process by encoding the temporal propagation graph with a temporal graph network (TGN) and a neural Hawkes process; **UCD-RD** (Ran and Jia, 2023) which uses prototype-based contrastive learning to initialise prototypes via in-distribution samples, and then it aligns the OOD data features with the corresponding prototypes. In addition to these traditional fake news detection methods, we also compare with graph OOD methods including **G-mixup** (Han et al., 2022), **CIGA** (Chen et al., 2022).

Baseline models trained with both in-distribution and low-resource OOD data include: **ACLR** (Lin et al., 2022) which utilises adversarial contrastive learning to transfer pre-trained BiGCN (Bian et al., 2020) models from a source domain to a target domain for fake news detection; **CADA** (Li et al., 2023) which serves as a plug-in module that adapts pre-trained models from a source domains to a target domain based on label-aware domain adversarial neural networks (Ganin and Lempitsky, 2015). In our experiments, CADA uses BiGCN, RoBERTa, SEAGEN and GACL as the pre-trained models. The web-retrieval and Large Language Model (LLM) prompt-based method **DELL** (Wan et al., 2024) is also compared.

All baselines and CSDA are implemented in Pytorch[1] and trained using an A100 GPU. The baseline models use the default hyperparameter settings from their original papers. Hyperparameter γ, τ of the CSDA model are set to 0.2, 0.1 respectively in the experiments. The hyperparameters are selected empirically based on a grid search shown in Fig 3c. The best parameters are selected on the training of Twitter dataset and applied to the training of Weibo dataset.

4.2 Results

Table 1 and Table 2 present the model performance on the four dataset settings (from Twitter, Weibo to Twitter-COVID19, Weibo-COVID19[2].

In Table 1, the models are categorized into two groups. The upper group consists of sequence-based models (LSTM, CNN, RvNN, PLAN, and RoBERTa), while the bottom group includes graph-

[1]https://pytorch.org/

[2]All reported results are averaged of five runs

Table 1: Zero-shot Fake News Detection on Twitter-COVID19 and Weibo-COVID19 (Acc: Accuracy score on fake news detection; F-F1: F1 score on fake news detection; T-F1: F1 score on true news detection).

Source	Twitter						Weibo					
Target	Twitter-COVID19			Weibo-COVID19			Twitter-COVID19			Weibo-COVID19		
Method	Acc	T-F1	F-F1	Acc	T-F1	F-F1	Acc	T-F1	F-F1	Acc	T-F1	F-F1
LSTM	0.412	0.426	0.340	0.463	0.329	0.498	0.510	0.243	0.533	0.416	0.428	0.416
CNN	0.406	0.450	0.285	0.445	0.328	0.476	0.498	0.249	0.528	0.421	0.438	0.382
RvNN	0.436	0.458	0.401	0.514	0.426	0.538	0.540	0.247	0.534	0.479	0.548	0.437
PLAN	0.455	0.432	0.476	0.532	0.414	0.578	0.573	0.298	0.549	0.384	0.283	0.461
RoBERTa	0.479	0.430	0.531	0.623	0.459	0.711	0.603	0.585	0.619	0.680	0.714	0.637
BiGCN	0.468	0.546	0.356	0.569	0.429	0.586	0.616	0.252	0.577	0.612	0.681	0.441
SEAGEN	0.494	0.448	0.494	0.555	0.406	0.583	0.578	0.320	0.650	0.586	0.613	0.424
GACL	0.541	0.545	0.536	0.601	0.410	0.616	0.621	0.345	0.666	0.688	0.635	0.727
UCD-RD	0.665	0.453	**0.767**	0.631	0.510	0.621	0.591	0.371	0.583	0.689	0.451	0.783
G-mixup	0.395	0.319	0.259	0.549	0.530	0.555	0.388	0.319	0.263	0.431	0.375	0.324
CIGA-gcn	0.492	0.476	0.500	0.672	**0.648**	0.672	0.542	0.542	0.539	0.694	0.669	0.693
CIGA-gin	0.475	0.471	0.459	0.627	0.544	0.596	0.450	0.418	0.382	0.732	0.685	0.718
CSDA (ours)	**0.713**	**0.556**	0.763	**0.701**	0.586	**0.723**	**0.697**	**0.651**	**0.732**	**0.741**	**0.721**	**0.809**
↑ (%)	+7.21	+1.83	-0.52	+4.32	-9.57	+1.69	+12.23	+11.28	+9.91	+1.23	+5.26	+3.32

Table 2: Few-shot Fake News Detection on Twitter-COVID19 (trained on Twitter) and Weibo-COVID19 (trained on Weibo) (Acc: Accuracy score on fake news detection; F-F1: F1 score on fake news detection; T-F1: F1 score on true news detection).

Method	Twitter-COVID19			Weibo-COVID19		
	Acc	T-F1	F-F1	Acc	T-F1	F-F1
$\text{CADA}_{\text{BiGCN}}$	0.681	0.621	0.725	0.716	0.552	0.792
$\text{CADA}_{\text{RoBERTa}}$	0.711	0.540	0.790	0.839	0.783	0.878
$\text{CADA}_{\text{SEAGEN}}$	0.669	0.383	0.785	0.662	0.471	0.752
$\text{CADA}_{\text{GACL}}$	0.641	0.511	0.716	0.684	0.402	0.786
ACLR	0.741	0.607	**0.799**	0.897	0.847	0.917
DELL	0.446	0.384	0.497	0.800	0.743	0.852
$\text{CSDA}_{\text{Fine-Tuned}}$	**0.772**	**0.767**	0.797	**0.922**	**0.884**	**0.940**
↑ (%)	+4.18	+26.36	-0.25	+2.79	+4.37	+2.51

based models (BiGCN, SEAGEN, GACL, UCD-RD, G-Mixup, CIGA, and CSDA). Overall, the graph-based models outperform the sequence-based ones, underscoring the effectiveness of leveraging propagation graphs for fake news detection. Among the graph-based models, CSDA consistently achieves the best performance across both datasets in terms of accuracy and F1 scores.

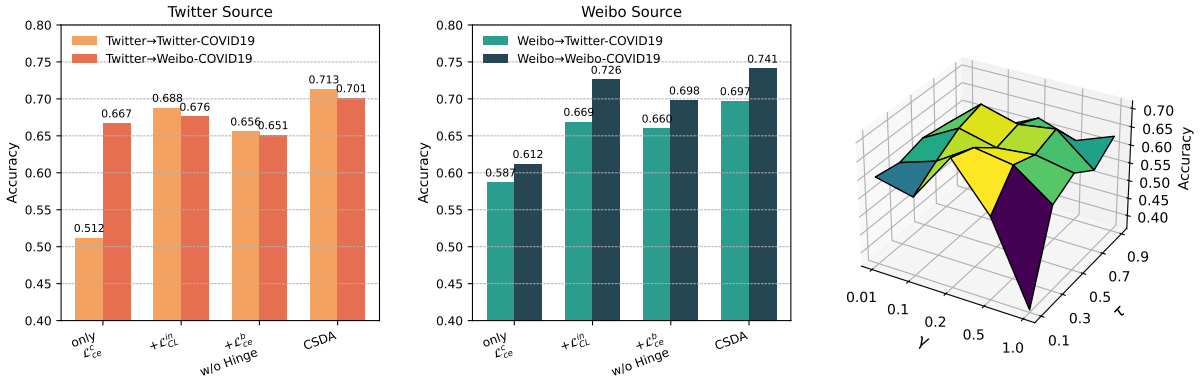
Baseline models that do not account for OOD data generally exhibit poor performance. These models are trained on open-domain in-distribution datasets and are therefore biased by domain-specific information. UCD-RD seeks to align the representations of in-distribution and OOD news samples belonging to the same class. However, it fails to address domain biases, making it less effective than CSDA. The graph OOD generalisation method, CIGA, demonstrates significant improvements only on the Weibo-COVID19 dataset, whereas G-Mixup fails to deliver any notable improvements. The reason can be these methods are designed for more sophisticated graph structures and are less suited to news propagation graphs, which feature simpler structures but more complex node attributes.

As shown in Table 2, when labelled OOD data is available, the baseline models (BiGCN, RoBERTa, SEAGEN and GACL) powered by CADA can learn features from the OOD data and achieve better accuracy than their vanilla versions. ACLR, which is designed for domain adaptation, achieves even better performance. However, these models are still outperformed by CSDA using fine-tuning with a performance improvement of $2.79 \sim 4.18\%$. DELL has good performance on Weibo-COVID19 dataset but performs poorly on Twitter-COVID19, showing both promising results and limitations of LLMs in fake news detection, which could provide some inspirations for future work.

4.3 Ablation Study

To show the effectiveness of the causal subgraph extraction module and impact on the loss functions,

four variants of CSDA are trained and the averaged results are shown in Fig 3a and Fig 3b. In the first variant "Only $\mathcal{L}_{ce}^{c}$", the subgraph extraction and classification module are trained purely based on the prediction loss. The remaining three variants all use causal subgraph extraction. They each add one additional loss component, with the final



(a) Ablation 1 (trained on `Twitter`). (b) Ablation 2 (trained on `Weibo`). (c) Parameter Sensitivity.

Figure 3: Ablation experiment and parameter sensitivity.

Table 3: Examples from `Twitter-COVID19`.

News, Comments, Node Scores and Edge Scores
Source news 1: The World Health Organization confirmed that Covid-19 is deadlier than the seasonal flu, but does not transmit as flu... [Node Score: <0.001] [Label: **FAKE**] **Comment 1** on source news: Need to buy a lot of masks contact me. [Node Score: 0.154] [Edge 0→1 Score: 0.152] **Comment 2** on source news: Because of their more rigorous testing protocols, South Korea's mortality rate of 0.6% is the most accurate. [Node Score: 0.393] [Edge 0→2 Score: 0.515] **Comment 3** on source news: why don't you look at implementing #Covid_19 travel health cards that confirm the person has been... [Node Score: 0.514] [Edge 0→3 Score: 0.462] **Comment 4** on source news: WHO is also omitting mild cases from their stats. [Node Score: 0.556] [Edge 0→4 Score: 0.574]
Source news 2: Rumours are no less infectious than #coronavirus! This looks like a meticulous list, but a fake one too... URL [Node Score: 0.1] [Label: **True**] **Comment 1** on source news: Yeah, this is fake coz you guys have totally disallowed stores from delivering essent... URL [Node Score: 0.446] [Edge Score: 0.554] **Comment 2** on source news: Why are police personnel beating up vegetable vendors and delivery guys... URL [Node Score: <0.01] [Edge Score: 0.036] **Comment 3** on comment 2: We have forwarded your query to the xxx. You can contact them on xxx-xxxxxx. [Node Score: 0.190] [Edge Score: 0.809]

model being the complete `CSDA` model. The results show the importance of each model component, especially the disentangling training objectives which guide the causal subgraph extraction module.

4.4 Case Study

The effectiveness of the `CSDA` model is further demonstrated through a case study using the `Twitter` and `Twitter-COVID19` datasets. The mask generator, trained on `Twitter`, is applied to `Twitter-COVID19` to filter out biased subgraphs while preserving causal ones. As shown in Table 3, source news with an official tone receives low node scores, indicating limited standalone value for classification (The indexes of news/comments are specified by the index number. The node and edge scores are calculated by `CSDA`'s mask generator). In contrast, comments revealing the news veracity are scored higher, while irrelevant or propagandistic content is down-weighted. This allows the Graph Encoder to focus on causal signals, enhancing detection.

This case study highlights a latent link between graph OOD generalisation and semantic reasoning: semantically meaningful content tends to receive higher scores, suggesting the model implicitly captures domain-invariant, informative semantics.

5 Conclusions

We presented the `CSDA` model for detecting fake news across domains by extracting and leveraging causal substructures in new propagation graphs. `CSDA` addresses the limitations of existing models in handling domain biases and OOD data, highlighting the importance of causal elements in news propagation graphs. Through extensive experiments, we show that `CSDA` outperforms not only sequence-based models but also other graph-based models, achieving higher accuracy, particularly in cross-domain scenarios. We also show that the integration of a fine-tuning process with low-resource OOD data further enhances `CSDA`'s robustness and adaptability. Interestingly, the indicated connection between graph OOD generalisation and semantic reasoning revealed in the case study also points future direction to reason on the propagation graph.

Limitations

While the paper introduces a novel and effective approach, it lacks concrete validation of the causal subgraph assumptions. Further work needs to be done to prove the identified subgraphs are truly causal rather than robust correlates. More interpretability could be gained. Furthermore, the evaluation is limited to propagation graphs from Twitter and Weibo, leaving open the question of generalisability to other domains or platforms.

Besides, in the era of large language models (LLMs), many fake news detection systems leverage powerful language understanding and retrieval capabilities, often without relying on propagation structures. This poses a limitation for CSDA, which requires structured propagation graphs, potentially making it less applicable when only raw text is available. However, CSDA's causal subgraph approach can complement LLMs by serving as a structure-aware module—for instance, its causal masks can be used to select or highlight informative user comments for LLMs to summarize or verify, which could be promising future work.

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 *AAAI*.

Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. 2020. Simple and deep graph convolutional networks. In *ICML*.

Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, Kaili Ma, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. 2022. Learning causally invariant representations for out-of-distribution generalization on graphs. In *NeurIPS*.

Koby Crammer and Yoram Singer. 2001. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of machine learning research*, 2(Dec):265–292.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Shaohua Fan, Xiao Wang, Yanhu Mo, Chuan Shi, and Jian Tang. 2022a. Debiasing graph neural networks via learning disentangled causal substructure. In *Advances in Neural Information Processing Systems*.

Shaohua Fan, Xiao Wang, Yanhu Mo, Chuan Shi, and Jian Tang. 2022b. Debiasing graph neural networks via learning disentangled causal substructure. In *NeurIPS*.

Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Syntactic stylometry for deception detection. In *ACL*.

Wenzheng Feng, Jie Zhang, Yuxiao Dong, Yu Han, Huanbo Luan, Qian Xu, Qiang Yang, Evgeny Kharlamov, and Jie Tang. 2020. Graph random neural networks for semi-supervised learning on graphs. In *NeurIPS*.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *ICML*.

Shuzhi Gong, Richard O Sinnott, Jianzhong Qi, and Cecile Paris. 2023a. Fake news detection through graph-based neural networks: A survey. *arXiv preprint arXiv:2307.12639*.

Shuzhi Gong, Richard O Sinnott, Jianzhong Qi, and Cecile Paris. 2023b. Fake news detection through temporally evolving user interactions. In *PAKDD*.

Shurui Gui, Meng Liu, Xiner Li, Youzhi Luo, and Shuiwang Ji. 2024. Joint learning of label and environment causal independence for graph out-of-distribution generalization. In *NeurIPS*.

Xiaotian Han, Zhimeng Jiang, Ninghao Liu, and Xia Hu. 2022. G-Mixup: Graph data augmentation for graph classification. In *ICML*.

Ling Min Serena Khoo, Hai Leong Chieu, Zhong Qian, and Jing Jiang. 2020. Interpretable rumor detection in microblogs by attending to user interactions. In *AAAI*.

Jingqiu Li, Lanjun Wang, Jianlin He, Yongdong Zhang, and Anan Liu. 2023. Improving rumor detection by class-based adversarial domain adaptation. In *ACM-MM*.

Xiner Li, Shurui Gui, Youzhi Luo, and Shuiwang Ji. 2024. Graph structure extrapolation for out-of-distribution generalization. In *ICML*.

Zenan Li, Qitian Wu, Fan Nie, and Junchi Yan. 2022. Graphde: A generative framework for debiased learning and out-of-distribution detection on graphs. In *NeurIPS*.

Hongzhan Lin, Jing Ma, Liangliang Chen, Zhiwei Yang, Mingfei Cheng, and Chen Guang. 2022. Detect rumors in microblog posts for low-resource domains via adversarial contrastive learning. In *Findings of the ACL: NAACL*.

Jiashuo Liu, Zheyan Shen, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. 2021. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*.

Yang Liu, Xiang Ao, Fuli Feng, Yunshan Ma, Kuan Li, Tat-Seng Chua, and Qing He. 2023. FLOOD: A flexible invariant learning framework for out-of-distribution generalization on graphs. In *KDD*.

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. *arXiv preprint arXiv:1907.11692*.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. In *IJCAI*.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. In *ACL*.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor detection on twitter with tree-structured recursive neural networks. In *ACL*.

Siqi Miao, Miaoyuan Liu, and Pan Li. 2022. Interpretable and generalizable graph learning via stochastic attention mechanism. *ICML*.

Erxue Min, Yu Rong, Yatao Bian, Tingyang Xu, Peilin Zhao, Junzhou Huang, and Sophia Ananiadou. 2022. Divide-and-conquer: Post-user interaction network for fake news detection on social media. In *WWW*.

Ahmadreza Mosallanezhad, Mansooreh Karami, Kai Shu, Michelle V. Mancenido, and Huan Liu. 2022. Domain adaptive fake news detection via reinforcement learning. In *WWW*.

Van-Hoang Nguyen, Kazunari Sugiyama, Preslav Nakov, and Min-Yen Kan. 2020. FANG: Leveraging social context for fake news detection using graph representation. In *CIKM*.

Hyeonjin Park, Seunghun Lee, Sihyeon Kim, Jinyoung Park, Jisu Jeong, Kyung-Min Kim, Jung-Woo Ha, and Hyunwoo J. Kim. 2022. Metropolis-hastings data augmentation for graph neural networks. In *NeurIPS*.

Hongyan Ran and Caiyan Jia. 2023. Unsupervised cross-domain rumor detection with contrastive learning and cross-attention. In *AAAI*.

Chris Samarinas, Wynne Hsu, and Mong Li Lee. 2021. Improving evidence retrieval for automated explainable fact-checking. In *NAACL*.

Kai Shu, Suhang Wang, and Huan Liu. 2019. Beyond news contents: The role of social context for fake news detection. In *WSDM*.

Amila Silva, Ling Luo, Shanika Karunasekera, and Christopher Leckie. 2021. Embracing domain differences in fake news: Cross-domain fake news detection using multi-modal data. In *AAAI*.

Tiening Sun, Zhong Qian, Sujun Dong, Peifeng Li, and Qiaoming Zhu. 2022. Rumor detection on social media with graph adversarial contrastive learning. In *WWW*.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

Herun Wan, Shangbin Feng, Zhaoxuan Tan, Heng Wang, Yulia Tsvetkov, and Minnan Luo. 2024. DELL: Generating reactions and explanations for LLM-based misinformation detection. In *ACL-Findings*.

Lirong Wu, Haitao Lin, Yufei Huang, and Stan Z. Li. 2022a. Knowledge distillation improves graph structure augmentation for graph neural networks. In *NeurIPS*.

Yingxin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. 2022b. Discovering invariant rationales for graph neural networks. In *ICLR*.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2018. How powerful are graph neural networks? In *ICLR*.

Feng Yu, Qiang Liu, Shu Wu, Liang Wang, Tieniu Tan, and 1 others. 2017. A convolutional approach for misinformation identification. In *IJCAI*.

Junchi Yu, Jian Liang, and Ran He. 2023. Mind the label shift of augmentation-based graph ood generalization. In *CVPR*.

Zhenrui Yue, Huimin Zeng, Ziyi Kou, Lanyu Shang, and Dong Wang. 2022. Contrastive domain adaptation for early misinformation detection: A case study on covid-19. In *CIKM*.

Jiajun Zhang, Zhixun Li, Qiang Liu, Shu Wu, Zilei Wang, and Liang Wang. 2024. Evolving to the future: Unseen event adaptive fake news detection on social media. In *CIKM*.

Tong Zhao, Yozen Liu, Leonardo Neves, Oliver Woodford, Meng Jiang, and Neil Shah. 2022. Data augmentation for graph neural networks. In *AAAI*.

A Domain Difference

The domain differences between the in-distribution data and the out-of-distribution data are examined from two perspectives: text content and graph statistics.

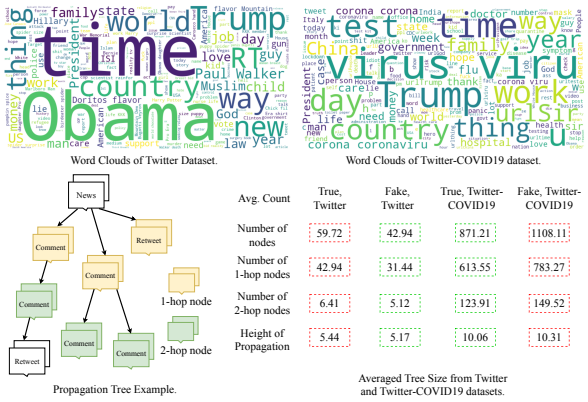


Figure 4: Visualisation of domain differences.

As illustrated in Fig. 4, the Word Clouds of the Twitter and Twitter-COVID19 datasets highlight the text content disparities between the two domains. Consequently, it is challenging to transfer linguistic-based models from Twitter to Twitter-COVID19. In addition to text content, we also analyse the statistics of the propagation graph, focusing on the number of nodes and the heights of the propagation trees. The results reveal that true news on Twitter generally exhibits larger propagation, whereas fake news on Twitter-COVID19 shows greater propagation, posing challenges for traditional graph-based fake news detection models. To address these issues, we propose our model, CSDA, designed to capture the causal aspects of propagation.

B Related Work

Fake News Detection Traditional fake news detection methods have explored *news content*, *social context* and *social environment* aspects. Content-based methods learn content or style features from the text or multi-media content (Feng et al., 2012). They may also leverage external knowledge for fact checking (Samarinas et al., 2021). Social context-based methods exploit user features (Shu et al., 2019) and user interactions that occur in news propagation. They use both sequence modelling (Ma et al., 2016; Khoo et al., 2020) and graph modelling (Bian et al., 2020; Gong et al., 2023b)models. Environment-based methods (Nguyen et al., 2020) consider associations across multiple news domains to extract broader contextual information.

Cross-domain fake news detection aims to train a model in one domain (the *source domain*) and apply the model to a different domain (the *target domain*). This is achieved using *sample-level* and *feature-level* methods. Sample-level methods identify domain-invariant samples in the training set and assign larger weights to them (Silva et al., 2021; Yue et al., 2022). Feature-level methods focus on weighting or extracting domain-independent features. For example, Mosallanezhad et al. utilise reinforcement learning to select domain-invariant attributes from news features. Inspired by domain-adaptive neural networks (Ganin and Lempitsky, 2015), studies such as (Min et al., 2022; Li et al., 2023) train an additional domain discriminator adversarially by attempting to generate news embeddings that cannot be recognised by a domain discriminator. However, these works require knowledge of the target domains and they have not explored the challenges in fake news detection from unseen news domains.

Graph Out-of-Distribution Generalisation Despite the success of graph machine learning, most methods assume that training and testing data share the same distribution (the in-distribution hypothesis). In practice, this assumption is often unrealistic, especially for nascent fake news. Traditional graph methods struggle with OOD generalisation, causing performance degradation. Recent advancements improve OOD generalisation through two main strategies: data-centric methods (Feng et al., 2020; Park et al., 2022; Wu et al., 2022a; Zhao et al., 2022; Li et al., 2024), which modify the training graph data to improve robustness, and invariant learning (Chen et al., 2022; Miao et al., 2022; Wu et al., 2022b; Liu et al., 2023; Yu et al., 2023; Gui et al., 2024), which focus on identifying consistent feature-label relationships across distributions while eliminating environment-specific correlations.

However, to the best of our knowledge, no existing graph-based OOD generalisation methods have been successfully adapted to fake news, because of the complex language semantics in the news propagation graph. Our experiments have shown that the direct application of graph OOD generalisation methods leads to low fake news detection accuracy.

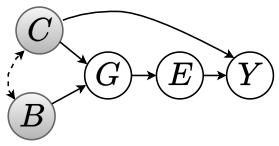
C Causal Analysis

In this paper, Structural Causal Models (SCMs) are employed to characterize the key features of the fake news detection problem and to elucidate the interactions among these features. We conduct a causal analysis of several variables to assess the differences and effectiveness of our CSDA model.

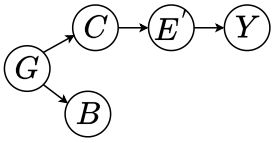
As depicted in Fig.5a, we consider five variables: the unobserved causal subgraph variable C, the unobserved biased subgraph variable B, the observed graph G, the graph embedding E, and the ground truth or prediction Y. Since the prediction is optimized to match the ground truth, we use the same variable to represent both. Fig.5a illustrates the SCM, where each link denotes a causal relationship.

The link $C \rightarrow Y$ indicates that C is the sole endogenous parent responsible for generating the ground truth label Y. For instance, C represents the oracle propagation subgraph, which precisely explains why the label is assigned as Y. However, the observed graph data G is generated by both the causal variable C and the bias variable B, leading to the fusion of biased subgraph information into the embedding E, which can result in incorrect predictions.

Our objective, therefore, is to decompose the observed graph G to uncover the unobserved variables C and B, and to utilize only the causal subgraph C to generate a causal embedding E', as illustrated in Fig. 5b. This approach ensures that the prediction Y is uncorrelated with the biased information B.



(a) Structural Causal Model of the union of the data generation process and the prediction process of traditional graph-based fake news detection methods. The grey and white variables represent unobserved and observed variables.



(b) Structural Causal Model of our CSDA model.

Figure 5: Structural Causal Models.

Table 4: Experimental Dataset Statistics ("Avg. depth" refers to the average number of layers of the news propagation graphs, i.e., trees)

	Twitter	**Twi-COVID**	**Weibo**	**Wei-COVID**
# news	1,154	400	4,649	399
# graph nodes	60,409	406,185	1,956,449	26,687
# true news	579	148	2,336	146
# fake news	575	252	2,313	253
Avg. depth	11.67	143.03	49.85	4.31
Avg. # posts	52	1,015	420	67
Domain	Open	COVID-19	Open	COVID-19
Language	English	English	Chinese	Chinese

D Data Statistics

The statistics of the datasets are shown in Table. 4.

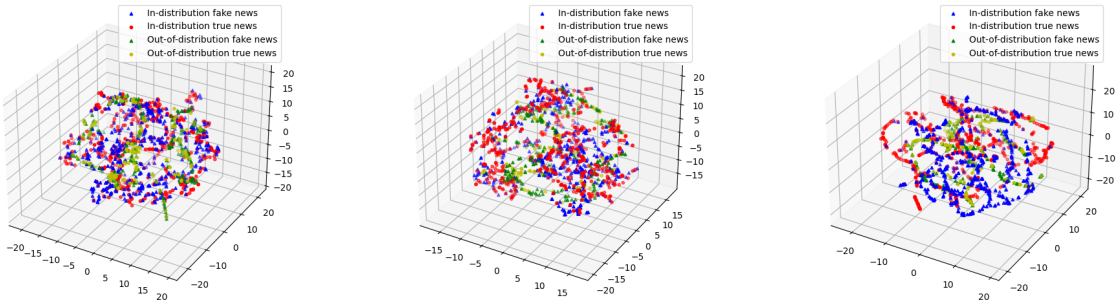
E Training Algorithm

To give more details about the training process, the pseudo code of the training is given in the following Algorithm 1. The training algorithm is discussed from two aspects: With the low-resource OOD data available in training (few-shot) and only in-distribution data available in training (zero-shot).

F Feature Visualisation

Fig. 6 shows the T-SNE (Van der Maaten and Hinton, 2008) visualisation of learned news embeddings from three representative models: BiGCN, UCD-RD and our CSDA. These models are utilised to learn embeddings for news items from Twitter and Twitter-COVID datasets. The computed embeddings are visualised through T-SNE under the same settings.

From Fig. 6, we observe that CSDA learns more discriminative representations, leading to better separations between the clusters of fake news and true news. This reaffirms that CSDA can effectively extract the causal information from the news propagation graphs for fake news detection.



(a) BiGCN representation. (b) UCD-RD representation. (c) CSDA representation.

Figure 6: TSNE feature visualisation of three representative models.

Algorithm 1 CSDA Training Algorithm

Input: A set of in-distribution (ID) news C_i^{in};
Optional Input: A set of out-of-distribution (OOD) news C_i^{out}.
Output: Assign news veracity labels $y \in \{0, 1\}$ to given unlabelled test data samples.

if OOD data is provided **then**
  for each mini-batch N^{out} from the OOD data **do**
    for each mini-batch N^{in} from the ID data **do**
      Combine the N^{out} and the N_{in} to a integrated mini-batch as C_i^*
      Pass C_i^* to the Mask Generator to get the causal subgraph $\mathcal{G}_{ic}^*$ and the biased subgraph $\mathcal{G}_{ib}^*$;
      Encode $\mathcal{G}_{ic}^*$ and $\mathcal{G}_{ib}^*$ to embedding $\mathcal{Z}_c$ and $\mathcal{Z}_b$ by corresponding causal and biased graph encoders;
      Permute the $\mathcal{Z}_b$ and the corresponding label y to $\widetilde{\mathcal{Z}_b}$ and $\widetilde{y}$;
      Calculate the label predictions through causal MLP and biased MLP;
      Calculate the enhanced overall loss with $\mathcal{L}^{'}$ with contrastive learning loss $\mathcal{L}_{CL}$ based on the embedding Z, predictions and training labels;
      Jointly optimize parameters given loss $\mathcal{L}^{'}$;
    end for
  end for
else
  for each mini-batch N^{in} from the ID data **do**
    Treat N_{in} as a mini-batch C_i^*
    Pass C_i^* to the Mask Generator to get the causal subgraph $\mathcal{G}_{ic}^*$ and the biased subgraph $\mathcal{G}_{ib}^*$;
    Encode $\mathcal{G}_{ic}^*$ and $\mathcal{G}_{ib}^*$ to embedding $\mathcal{Z}_c$ and $\mathcal{Z}_b$ by corresponding causal and biased graph encoders;
    Permute the $\mathcal{Z}_b$ and the corresponding label y to $\widetilde{\mathcal{Z}_b}$ and $\widetilde{y}$;
    Calculate the label predictions through causal MLP and biased MLP;
    Calculate the enhanced overall loss loss $\mathcal{L}$ based on the predictions and training labels;
    Jointly optimize parameters given loss $\mathcal{L}$;
  end for
end if