Causal Walk: Debiasing Multi-Hop Fact Verification with Front-Door Adjustment

Congzhi Zhang*, **Linhai Zhang***, **Deyu Zhou**†

School of Computer Science and Engineering, Key Laboratory of Computer Network and Information Integration, Ministry of Education, Southeast University, China
{zhangcongzhi, lzhang472, d.zhou}@seu.edu.cn

Abstract

Multi-hop fact verification aims to detect the veracity of the given claim by integrating and reasoning over multiple pieces of evidence. Conventional multi-hop fact verification models are prone to rely on spurious correlations from the annotation artifacts, leading to an obvious performance decline on unbiased datasets. Among the various debiasing works, the causal inference-based methods become popular by performing theoretically guaranteed debiasing such as casual intervention or counterfactual reasoning. However, existing causal inference-based debiasing methods, which mainly formulate fact verification as a single-hop reasoning task to tackle shallow bias patterns, cannot deal with the complicated bias patterns hidden in multiple hops of evidence. To address the challenge, we propose Causal Walk, a novel method for debiasing multi-hop fact verification from a causal perspective with front-door adjustment. Specifically, in the structural causal model, the reasoning path between the treatment (the input claim-evidence graph) and the outcome (the veracity label) is introduced as the mediator to block the confounder. With the front-door adjustment, the causal effect between the treatment and the outcome is decomposed into the causal effect between the treatment and the mediator, which is estimated by applying the idea of random walk, and the causal effect between the mediator and the outcome, which is estimated with normalized weighted geometric mean approximation. To investigate the effectiveness of the proposed method, an adversarial multi-hop fact verification dataset and a symmetric multi-hop fact verification dataset are proposed with the help of the large language model. Experimental results show that Causal Walk outperforms some previous debiasing methods on both existing datasets and the newly constructed datasets. Code and data will be released at https://github.com/zcccccz/CausalWalk.

Introduction

Fact verification aims to verify the given claim based on the retrieved evidence, which is a challenging task. Previous work formulates fact verification as a natural language inference task, where multiple evidence pieces are concatenated together and a single-hop inference is performed (Hanselowski et al. 2018; Nie, Chen, and Bansal 2019). However, in many cases, the process of verifying a claim requires integrating and reasoning over several pieces of evidence (Ostrowski et al. 2021). Therefore, multi-hop fact verification, which performs a multi-hop reasoning process to verify a claim, has become an attractive research topic recently (Zhou et al. 2019; Zhao et al. 2020a; Si, Zhu, and Zhou 2023).

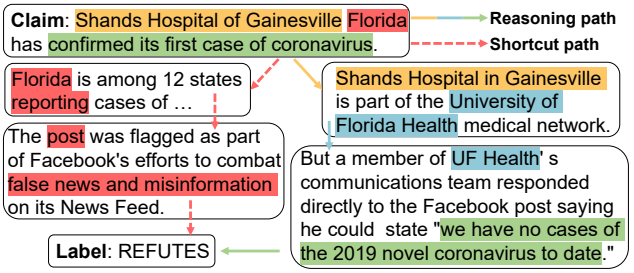

Figure 1: Illustration of an example of bias in multi-hop fact verification dataset, which is taken from the PolitiHop dataset. The solid line indicates the reasoning path while the dashed line indicates the shortcut path.

Though notable progress has been made, most multi-hop fact verification methods focus on learning label-specific features for judging the veracity of the given claim, which may expose the models to hidden data bias. Previous study (Schuster et al. 2019) has shown that there are annotation biases in a commonly used fact verification dataset, FEVER (Thorne et al. 2018), where most of the claims require only a single piece of evidence to verify. As shown in Figure 1, we observe that there are also biases in the multi-hop fact verification datasets such as PolitiHop (Ostrowski et al. 2021). The sentence "The post was flagged as part ..." appears 85 times in the training set of PolitiHop with a high correlation with the REFUTES label. Therefore, it is easy for the black-box neural network methods to learn such shortcut paths instead of multi-hop reasoning to cheat and obtain the right answer. We found obvious performance declines for some popular multi-hop fact verification methods when such biases are removed in a synthetic dataset. The same phenomenon, known as disconnected reasoning, has also been found in multi-hop question answering datasets (Trivedi et al. 2020).

*These authors contributed equally.
† Corresponding author.

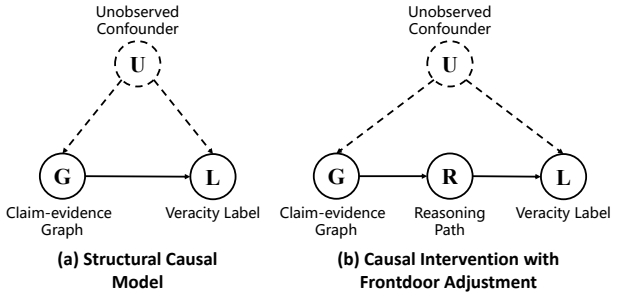

Figure 2: Structural Causal Model for multi-hop fact verification.

Previous methods for debiasing cannot deal with biases in multi-hop datasets as they mainly formulate fact verification as a single-hop reasoning task. For data augmentation-based methods (Wei and Zou 2019; Lee et al. 2021), it is difficult to generate unbiased multi-hop fact verification instances. For reweight-based methods (Schuster et al. 2019; Karimi Mahabadi, Belinkov, and Henderson 2020), it is hard to detect the biased samples as the bias patterns are complicated. Recently, causal inference has become a popular paradigm for debiasing methods because of its theoretical guarantee and generalizability, which aims to calculate the causal effect between the treatments (input examples) and outputs (labels). For causal inference-based methods, the common practices are causal intervention (Tian et al. 2022a) and counterfactual reasoning (Xu et al. 2023). However, these methods mainly focus on shallow bias patterns such as the correlation between specific types of words (e.g. negation words) and specific labels (e.g. REFUTES), and cannot deal with complicated bias patterns hidden in multiple hops of evidence.

To address the above challenge, we propose to debias the multi-hop fact verification by causal intervention based on front-door adjustment. As shown in Figure 2(a), we reflect the causal relationships in the multi-hop fact verification as a Structural Causal Model (SCM), where G is the graph consisting of claim and evidence. U is the unobservable confounder that introduces various biases. R is the reasoning path. L is the corresponding veracity label. The debiasing process is achieved by measuring the causal effect between treatment G and outcome L. As U absorbs various multi-hop biases, making it hard to model or detect, it becomes infeasible to employ back-door adjustment to calculate the causal effect between G and L. As shown in Figure 2(b), we introduce the reasoning path between G and L as a mediator variable R, which is unaffected by U and fully mediates the causal effect between G and L. With R introduced, the front-door adjustment can be employed by measuring the causal effect between G and L as an adding up of the causal effect between G and R and the causal effect between R and L.

In this paper, Causal Walk, a novel debiasing method for multi-hop fact verification based on front-door adjustment, is proposed. Specifically, to measure the causal effect between G and R, the idea of random walk is applied with the graph neural network to calculate the probability of the reasoning path. To measure the causal effect between R and L, normalized weighted geometric mean (NWGM) approximation is utilized with the recurrent neural network to estimate the unbiased outcome based on the reasoning path. Furthermore, we construct an adversarial multi-hop fact verification dataset and a symmetric multi-hop fact verification dataset by extending the PolitiHop dataset with the help of large language models. We conduct experiments on both single-hop and multi-hop fact verification datasets under both original and adversarial scenarios. Experimental results show that the proposed method outperforms previous debiasing methods on both single-hop and multi-hop datasets. The contributions of this work are three-fold.

- As far as we know, we are the first one to debias multi-hop fact verification task using front-door adjustment.
- We propose Causal Walk, a novel method to perform causal intervention with front-door adjustment by introducing the reasoning path between input and output as the mediator.
- Experimental results show the effectiveness of the proposed method on both previous datasets and the proposed datasets.

Related Work

Multi-hop Fact Verification

Early methods of fact verification mainly formulate fact verification as a natural language inference task (Hanselowski et al. 2018; Nie, Chen, and Bansal 2019). Zhou et al. (2019) first propose a graph-based evidence reasoning framework to enable the communication of multiple pieces of evidence in a fully connected graph. Zhong et al. (2020) introduce a semantic-level graph for fact verification. Liu et al. (2020) utilize the Kernel Graph Attention Network for fine-grained fact verification. Zhao et al. (2020a) update Transformer (Vaswani et al. 2017) with extra Hop attention for multi-hop reasoning tasks including fact verification. Chen et al. (2022) propose an evidence fusion network to capture global contextual information from various levels of evidence information. Si, Zhu, and Zhou (2023) cast explainable multi-hop fact verification as subgraph extraction with salience-aware graph learning. Fajcik, Motlicek, and Smrz (2023) present a 2-stage system composed of the retriever and the verifier, while we only focus on the debiasing of the verifier. Recently, some works have used LLM for multi-hop fact verification (Zeng and Gao 2023; Pan et al. 2023). Most of the existing multi-hop fact verification methods focus on modeling the reasoning process while ignoring the hidden bias in the datasets.

Debiasing with Causal Inference

Recently, causal inference has been preferred for the debiasing method, which provides a more principled way of defining a causal model and debiasing the model by measuring the causal effect. Xu et al. (2023) propose to mitigate the spurious correlation between the claims and the labels by subtracting the output of a claim-only model from the output of a claim-evidence fusion model. Tian et al. (2022a) combine the causal intervention and counterfactual reasoning to debias fact verification, where the do-calculus for

causal intervention is estimated by NWGM approximation. Besides fact verification, causal inference is also widely applied in debiasing other Natural Language Processing tasks and Computer Vision tasks. Wang et al. (2022) introduced instrumental variable estimation to debias implicit sentiment analysis. Guo et al. (2023) employed counterfactual reasoning for reducing disconnected reasoning in multi-hop QA. Niu et al. (2021) proposed to use counterfactual reasoning to debias the visual question answering task by subtracting the prediction of the language-only model from the prediction of the vision-language model. Some works apply backdoor adjustment or front-door adjustment to the image caption task (Liu et al. 2022; Yang et al. 2021; Yang, Zhang, and Cai 2021). Zhu et al. (2023) proposed neuron-wise and token-wise backdoor adjustments to mitigate name bias in machine reading comprehension. Chen et al. (2023) proposed to debias multi-modal fake news detection with both causal intervention and counterfactual reasoning. To our best knowledge, we are the first one to utilize front-door adjustment for multi-hop fact verification debiasing.

Preliminaries

Structural Causal Model and Causal Effect

The structural Causal Model reflects the causal relationships between certain variables we are interested in. As shown in Figure 2(a), SCM is often represented as a directed acyclic graph $SCM = \{V, E\}$, where V denotes the set of variables and E denotes the direct causal effect. G is a direct cause of L when variable L is the child of G. As for fact verification, we regard the graph that combines the claim and corresponding evidence as the treatment variable G, and the veracity label as the outcome variable L. The claim and evidence together determine the veracity label. Therefore we have an edge $G \rightarrow L$ to show the direct causal effect of G on L. Except for the input and output variables, there is another unobservable variable that affects both the input and output as a background, which we denote as the unobservable confounder variable U. For fact verification, U can be annotation artifacts that inevitably introduce biases between the input and output. For example, most annotators will follow certain patterns including negation words when generating REFUTES instances. So we have another path $G \leftarrow U \rightarrow L$ from G to L, which is also known as a backdoor path.

The conventional methods often adopt the total effect $P(L|G)$ to measure how input G affects output L. However, from the perspective of causal inference, such total effect does not reflect the real effect of G on L, it involves all paths from G to L including the backdoor path. Therefore, methods based on $P(L|G)$ have poor generalizability on out-of-domain datasets and are vulnerable to adversarial attacks for learning biases hidden in the backdoor path unconsciously. Contrary to conventional methods, causal inference calculates the direct causal effect between G and L with do-calculus $P(L|do(G))$. The do-calculus $P(L|do(G))$ measures the effect on L when intervening the treatment G but keeping other variables unchanged, while $P(L|G)$ only means the probability of L condition on G.

Causal Intervention with Front-door Adjustment

There are four main ways to calculate the do-calculus: randomized controlled trial, backdoor adjustment, front-door adjustment, and instrumental variable estimation. Both the randomized controlled trial method and the instrumental variable estimation method require a thorough control of the relationship between input text and label, which is difficult for multi-hop fact verification. The backdoor adjustment is also infeasible as the confounder U is too polymorphic to be observed exhaustively. Therefore, the front-door adjustment is chosen to perform the causal intervention and calculate $P(L|do(G))$.

To perform the front-door adjustment, a mediator variable is required to fully mediate the causal effect between G and L, i.e. all direct causal paths from G to L go through the mediator. For multi-hop fact verification, we choose the reasoning path in the claim-evidences graph as the mediator variable R. On the one hand, R is only affected by the graph G itself, on the other hand, the only way the graph G can affect the label L is through the reasoning path R. Therefore, we can specify the path $G \rightarrow R \rightarrow L$. With the mediator, based on the front-door adjustment, we have

$$P(L|do(G)) = \sum_r P(L|do(r))P(r|do(G)) \tag{1}$$

where $r \in R$ is the reasoning path between G and L. The causal effect between G and L is decomposed into the causal effect between G and R and the causal effect between R and L.

Since L is a collider of G and R, L blocks the backdoor paths of G and R. Therefore, based on the backdoor adjustment, we have

$$P(r|do(G)) = \sum_l P(r|G, l)P(l) = P(r|G) \tag{2}$$

where $l \in L$ is the veracity label.

Because G blocks $R \leftarrow G \rightarrow U \rightarrow L$, G satisfies the backdoor criterion, and we have

$$P(L|do(r)) = \sum_g P(L|r, g)P(g) \tag{3}$$

where $g \in G$ is the claim-evidence graph.

Methodology

In this section, multihop fact verification is formulated as a graph-based classification task. We first estimate $P(r|do(G))$ by applying the idea of random walk with the graph neural network. We then combine Normalized Weighted Geometric Mean approximation with the recurrent neural network to estimate $P(L|do(r))$. Finally, we use the beam search to estimate $P(L|do(G))$ and train the model.

Task Definition

Given the claim c and the corresponding evidence set $\{e_1, ..., e_n\}$, the model needs to detect the veracity of the claim. Following the previous work (Zhou et al. 2019), we combine the claim and evidence set into a graph G. Each

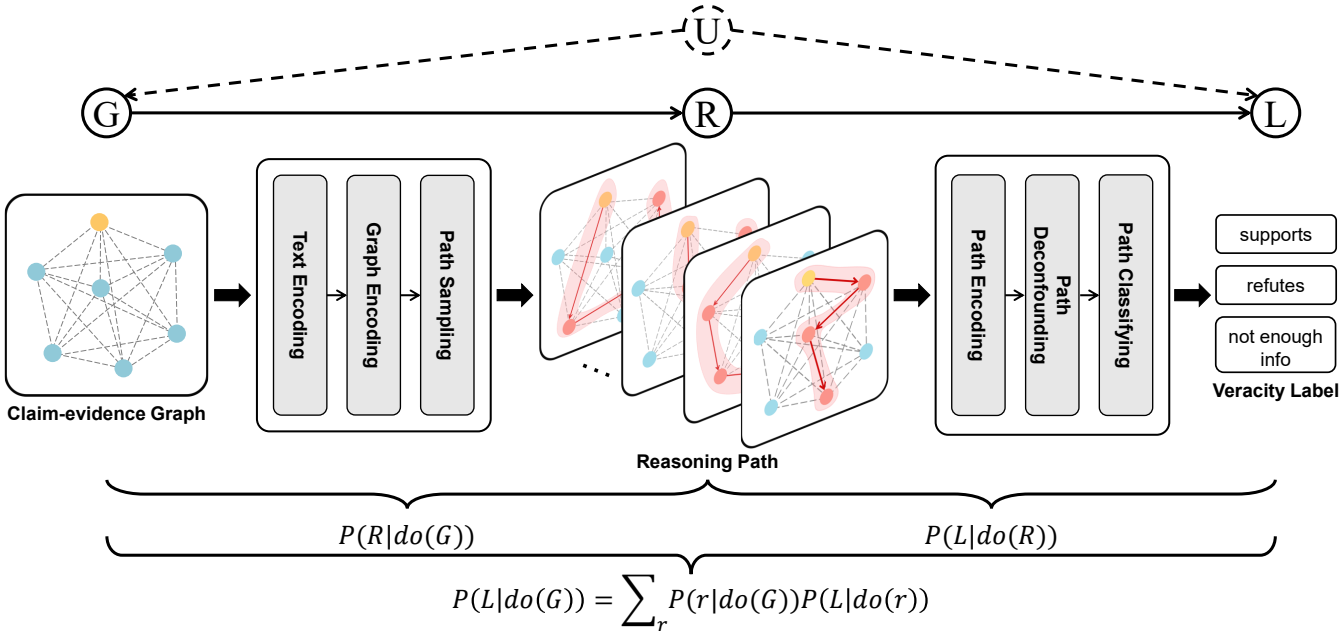

Figure 3: The causal view of the proposed method.

node in graph G represents a sentence. We define multi-hop fact verification as a graph classification problem, with labels including SUPPORTS, REFUTES, and NOT ENOUGH INFO. Specifically, we define the graph $G = \{v_0, ...v_n\}$. G can be divided into two subgraphs $G_{claim} = \{v_0\}$ and $G_{evidence} = \{v_1, ...v_n\}$. Since there is no more fine-grained prior information, we set graph G to be fully connected.

Estimation of $P(r|do(G))$

In this subsection, we will introduce how to use the walk-based method to estimate the $P(r|do(G))$ in the Equation (1). According to the Equation (2), the estimation of $P(r|do(G))$ is equivalent to the estimation of $P(r|G)$. To estimate the probability $P(r|G)$, we first obtain the node representation using the **Text Encoding** and **Graph Encoding**. We then compute the transition probability matrix using the node representation. Finally, we sample a path r with probability $P_{walk}(r)$ through the **Path Sampling**.

Text Encoding Following the previous work (Zhou et al. 2019; Liu et al. 2020), we employ BERT (Devlin et al. 2019) to obtain the semantic representation of the text.

The claim is directly fed into BERT to obtain the claim representation $\boldsymbol{x}^c$ while the evidence is concatenated with the claim as an evidence-claim pair (e_i, c) into BERT to obtain the evidence representation $\boldsymbol{x}_i^e$.

$$\begin{aligned} \boldsymbol{x}_0^c &= \text{BERT}(c) \\ \boldsymbol{x}_i^e &= \text{BERT}(e_i, c) \end{aligned} \tag{4}$$

For simplification, the superscripts of $\boldsymbol{x}_0^c$ and $\boldsymbol{x}_i^e$ are omitted in the following paper.

Graph Encoding To capture the structural information of the graph, we use graph convolution to update the node representation.

$$\boldsymbol{X}^{layer} = \text{GConv}(\boldsymbol{X}, \boldsymbol{A}) \tag{5}$$

$\boldsymbol{X} = \{\boldsymbol{x}_0, \boldsymbol{x}_1, ..., \boldsymbol{x}_n\} \in \mathbb{R}^{|V| \times F}$ is the initialized nodes representation from BERT. $\boldsymbol{A} \in \mathbb{R}^{|V| \times |V|}$ is the adjacency matrix. $layer$ represents the number of stacked layers of the graph convolution. $\boldsymbol{X}^{layer} \in \mathbb{R}^{|V| \times d}$ represents the updated nodes representation. The following operations in this paper are based on the updated node representation, i.e. $\boldsymbol{x}_i = \boldsymbol{X}^{layer}[i, :]$.

Path Sampling To sample a path and estimate the probability, we apply the idea of a random walk (Li, Zhu, and Zhang 2016; Christopoulou, Miwa, and Ananiadou 2018). The path on the graph represents the inference process for the claim.

The core parameter of the random walk is the transition probability. We calculate the weights of the adjacency matrix first and then use the softmax function to calculate the transition probability. In a random walk starting from v_0, the weight of edge (v_i, v_j) is calculated as follows:

$$a_{ij} = \text{MLP}(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_0) \tag{6}$$

We assume that the random walk follows a first-order Markov chain. The probability of jumping from node v_i to v_j is:

$$P(i \to j) = \text{softmax}(a_{ij}) = \frac{exp(a_{ij})}{\sum_{k \in N(i)} exp(a_{ik})} \tag{7}$$

where $N(i)$ is the neighbor set of node v_i.

Then we random sample an inference path $r = \{v_0, ..., v_m\}$. Its length is $m+1$. The probability of this path

can be calculated by:

$$P_{walk}(r) = \prod_{k=0}^{m-1} P(k \to k+1) \tag{8}$$

According to the assumption of a first-order Markov chain, we consider the conditional probability $P(r|G)$ and the path sampling probability $P_{walk}(r)$ to be equivalent. And then according to the Equation (2), we have

$$P(r|do(G)) = P(r|G) = P_{walk}(r) \tag{9}$$

Estimation of $P(L|do(r))$

In this subsection, we describe how to use Normalized Weighted Geometric Mean approximation to estimate $P(L|do(r))$. According to the Equation (3), we need to estimate $\sum_g P(g)P(L|r,g)$. We first obtain the path representation $\boldsymbol{x}_r$ and graph representation $\boldsymbol{x}_g$ by utilizing the **Path Encoding**. In **Path Deconfounding**, to reduce the computational cost, we use NWGM to absorb the sampling of g into the model, as shown in Equation (15). Since it is not possible to exhaust the values of g, we use a fixed dictionary $\boldsymbol{D}_g$ to store the compressed representation space of graph g. Finally, in **Path Classifying**, we use a classifier to classify the path representation after causal intervention.

Path Encoding We encode the nodes on the path r with a recurrent neural network to get the path representation $\boldsymbol{x}_r$,

$$\boldsymbol{x}_r = \text{LSTM}([\boldsymbol{x}_0, ..., \boldsymbol{x}_m], \boldsymbol{h}_0, \boldsymbol{c}_0) \tag{10}$$

where $\boldsymbol{h}_0$ and $\boldsymbol{c}_0$ are the initial state of the LSTM network. We set $\boldsymbol{h}_0 = \boldsymbol{c}_0 = \boldsymbol{x}_g$. The $\boldsymbol{x}_g$ is the graph representation:

$$\boldsymbol{x}_g = \text{Attention}(\boldsymbol{x}_0, [\boldsymbol{x}_1, ..., \boldsymbol{x}_n]) \tag{11}$$

where n is the number of evidence sentences.

Here we use the attention mechanism to learn graph representation:

$$\begin{aligned} w_i &= \text{MLP}(\boldsymbol{x}_0, \boldsymbol{x}_i) \\ \alpha_i &= \text{softmax}(w_i) = \frac{exp(w_i)}{\sum_{k=1}^{n} exp(w_k)} \\ \boldsymbol{x}_g &= \sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i \end{aligned} \tag{12}$$

Path Deconfounding To estimate $P(L|do(r))$, given the path r's representation $\boldsymbol{x}_r$ and graph g's representation $\boldsymbol{x}_g$, Equation (3) is implemented as:

$$\sum_{g \in G} P(\boldsymbol{x}_g) P(l|\boldsymbol{x}_r, \boldsymbol{x}_g) = \mathbb{E}_g[P(l|\boldsymbol{x}_r, \boldsymbol{x}_g)] \tag{13}$$

where G are the value spaces of g. $P(l|\boldsymbol{x}_r, \boldsymbol{x}_g)$ is the prediction results of classifier $f_{classifier}$:

$$\begin{aligned} P(l|\boldsymbol{x}_r, \boldsymbol{x}_g) &= f_{classifier}(\boldsymbol{x}_r, \boldsymbol{x}_g) \\ &= \text{softmax}(h(\boldsymbol{x}_r, \boldsymbol{x}_g)) \end{aligned} \tag{14}$$

where h is a feature fusion function.

Note that $\mathbb{E}_g$ cannot be calculated analytically, we use Normalized Weighted Geometric Mean (NWGM) (Xu et al. 2015) to make an approximation of expectation:

$$\begin{aligned} \mathbb{E}_g[P(l|\boldsymbol{x}_r, \boldsymbol{x}_g)] &= \mathbb{E}_g[\text{softmax}(h(\boldsymbol{x}_r, \boldsymbol{x}_g))] \\ &\approx \text{softmax}(\mathbb{E}_g[h(\boldsymbol{x}_r, \boldsymbol{x}_g)]) \end{aligned} \tag{15}$$

Following recent works (Tian et al. 2022a; Chen et al. 2023), we model $h(\boldsymbol{x}_r, \boldsymbol{x}_g) = \boldsymbol{W}_r \boldsymbol{x}_r + \alpha \boldsymbol{W}_g \boldsymbol{x}_g$, where $\boldsymbol{W}_r$ and $\boldsymbol{W}_g$ are learnable weight parameters, α is weight parameter for intervention. In this case, $\mathbb{E}_g[h(\boldsymbol{x}_r, \boldsymbol{x}_g)] = \boldsymbol{W}_r \boldsymbol{x}_r + \alpha \boldsymbol{W}_g \cdot \mathbb{E}_g[\boldsymbol{x}_g]$.

Since the values of g are inexhaustible, we propose to approximate it by designing a fixed dictionary $\boldsymbol{D}_g = [\boldsymbol{z}_1, ..., \boldsymbol{z}_N] \in \mathbb{R}^{N \times k \times d}$. N is the number of categories. k is the number of samples for each category. d is the feature dimension of graph representation. We couple the dictionary size and the category of labels so that $\boldsymbol{D}_g$ can be modeled as the confounder for each category. We use the graph representations from the training dataset to initialize the dictionary $\boldsymbol{D}_g$. Specifically, for each category l_i, we sample k cluster centers using the K-Means algorithm, i.e. $\boldsymbol{z}_i = [\boldsymbol{z}_i^1, ..., \boldsymbol{z}_i^k] \in \mathbb{R}^{k \times d}$, where $\boldsymbol{z}_i^k$ is a graph representation and it can be calculated by Equation (11).

To compute $\mathbb{E}_g[\boldsymbol{x}_g]$, we use a dot-product attention mechanism:

$$\begin{aligned} \boldsymbol{z}_i^{'} &= \text{softmax}(\boldsymbol{Q}^T \boldsymbol{K}) \boldsymbol{z}_i \\ \boldsymbol{D}_g^{'} &= [\boldsymbol{z}_1^{'}, ..., \boldsymbol{z}_N^{'}] \\ \mathbb{E}_g[\boldsymbol{x}_g] &= \frac{1}{N} P(l|\boldsymbol{x}_r) \boldsymbol{D}_g^{'} \end{aligned} \tag{16}$$

where $\boldsymbol{Q} = \boldsymbol{W}_q \boldsymbol{x}_r$, $\boldsymbol{K} = \boldsymbol{W}_k \boldsymbol{z}_i^T$ ($\boldsymbol{W}_q \in \mathbb{R}^{d \times d}$ and $\boldsymbol{W}_k \in \mathbb{R}^{d \times d}$ are learnable mapping matrices). $\boldsymbol{z}_i^{'} \in \mathbb{R}^d$ is the graph representation of category l_i computed by the dot-product attention. $\boldsymbol{D}_g^{'} \in \mathbb{R}^{N \times d}$ is the confounder dictionary after attention. $\boldsymbol{x}_r \in \mathbb{R}^d$ is the path r's representation.

Path Classifying $P(l|\boldsymbol{x}_r)$ in Equation (16) is the prediction results of classifier $f_{classifier}$ given only r input:

$$\boldsymbol{l}_r = P(l|\boldsymbol{x}_r) = f_{classifier}(\boldsymbol{x}_r) \tag{17}$$

where $\boldsymbol{l}_r \in \mathbb{R}^N$ is the probability distribution of the classification results.

In summary, with the Normalized Weighted Geometric Mean, we can deduce that

$$P(L|do(r)) = \text{softmax}(\boldsymbol{W}_r \boldsymbol{x}_r + \alpha \boldsymbol{W}_g \cdot \mathbb{E}[\boldsymbol{x}_g]) \tag{18}$$

Training and Inference

In the training stage, since we cannot determine which path is the true inference path, we use the beam search to sample several paths with the highest probability. Specifically, we sample a path set $R_{beam} = \{r_1, ..., r_w\}$, w is the beam search width.

According to Equations (1)(9)(18), we can implement

front-door adjustment as:

$$\begin{aligned}\boldsymbol{l}_{causal} &= P(L|do(G)) \\ &= \sum_r P_{walk}(r) \sum_g P(\boldsymbol{x}_g) P(L|\boldsymbol{x}_r, \boldsymbol{x}_g) \\ &= \mathbb{E}_{r \in R_{beam}}[\text{softmax}(\boldsymbol{W}_r \boldsymbol{x}_r + \alpha \boldsymbol{W}_g \cdot \mathbb{E}[\boldsymbol{x}_g])]\end{aligned} \tag{19}$$

where $\boldsymbol{l}_{causal} \in \mathbb{R}^N$ is the probability distribution of the classification results.

Finally, we use the cross-entropy loss for training:

$$\mathcal{L}_{causal} = -\boldsymbol{l}_{gold}^T log(\boldsymbol{l}_{causal}) \tag{20}$$

where $\boldsymbol{l}_{gold} \in \mathbb{R}^N$ is the ground-truth label.

We use the supervision of ground-truth labels to further enhance the learning of reasoning path:

$$\begin{aligned}\boldsymbol{l}_{pred} &= \sum_{r \in R_{beam}} P_{walk}(r) \boldsymbol{l}_r \\ \mathcal{L}_{walk} &= -\boldsymbol{l}_{gold}^T log(\boldsymbol{l}_{pred})\end{aligned} \tag{21}$$

The total training loss is:

$$\mathcal{L}_{total} = \mathcal{L}_{walk} + \mathcal{L}_{causal} \tag{22}$$

Experiments

Datasets

We evaluate the model performance on the FEVER dataset and PolitiHop dataset and their variants. For all datasets, label classification accuracy is adopted as the evaluation metric. For training, all models are trained on the original training set of FEVER and PolitiHop. For testing, the developed set of FEVER and the test set of PolitiHop are adopted, denoted as **FEVER** (Thorne et al. 2018) and **PolitiHop** (Ostrowski et al. 2021) respectively. We also include the adversarial versions of both datasets, denoted as **Adversarial FEVER** (Thorne et al. 2018) and **Adversarial PolitiHop** (Ostrowski et al. 2021) respectively. To further investigate the effectiveness of the proposed method on debiasing multi-hop fact verification, we also extend **PolitiHop** with the help of GPT-4 (OpenAI 2023).

Hard PolitiHop: We use GPT-4 to generate some misleading evidence sentences for **PolitiHop**. Specifically, we use GPT-4 to generate modified claims that are similar but parallel to the original claim. GPT-4 then generates evidence for the modified claims that have the opposite label. For example, if the original label is REFUTES, the relation between the generated new evidence and the modified claim should be SUPPORTS. We need to make sure that the Original claim and the new evidence are neither REFUTED nor SUPPORTED. Finally, the new evidence replaces the non-evidence sentences in the original claim.

Symmetric PolitiHop: Previous work (Schuster et al. 2019) has shown that symmetric datasets can evaluate the debiasing ability of models. We combine the new samples generated by GPT-4 and the original samples into the **Symmetric PolitiHop**. The generated evidence does not change the label of the original claim, so we use the setting of shared evidence (**ShareEvi**) to increase the difficulty of the dataset. That is, both the new claim and the original claim use the evidence set merged by the new evidence and the original evidence. Here we only consider 'SUPPORTS' and 'REFUTES' samples.

(Adversarial) FEVER-MH: In FEVER, 83.2% of the claims require one piece of evidence. Therefore, to evaluate the multi-hop reasoning ability of the model, we extract the samples that have more than 2 pieces of evidence to construct a variant **FEVER-MH**. We use the same method to construct dataset **Adversarial FEVER-MH**.

Baselines

We compare our proposed method with several baselines including multi-hop fact verification methods and causal-inference-based debiasing methods. For multi-hop fact verification methods, **GEAR** (Zhou et al. 2019), **KGAT** (Liu et al. 2020) and **Transformer-XH** (Zhao et al. 2020b) are adopted. For causal-inference-based debiasing methods for fact verification, **CICR** (Tian et al. 2022b) and **CLEVER** (Xu et al. 2023) are adopted. We also include a baseline for all methods, namely **BERT-Concat**, where the claim and all evidence are concatenated into a sequence to be fed into the BERT model. To investigate the effectiveness of previous causal inference-based methods for multi-hop fact verification, we also replace the encoders of **CICR** and **CLEVER** with the same graph neural networks of Causal walk: **CICR-graph** and **CLEVER-graph**.

Implementation Details

We utilize BERT$_{base}$ (Devlin et al. 2019) in all of the BERT fine-tuning baselines and our Causal Walk framework. The learning rate is 1e-5. All models are trained for 10 epochs with a batch size of 4. We update the parameters using Adam optimizer. BERT-Concat, CICR, and CLEVER have a maximum input length of 512, and the other models have a maximum input length of 128. The maximum number n of evidence per sample is 20. The beam width w is 3 and the path sampling length m is 5. The number of samples k for each category in the confounder dictionary is 5. The intervention weight parameter α is 0.1. Following (Xu et al. 2023), we train models on **FEVER** data and its variants without using 'NOT ENOUGH INFO' samples.

Results

Table 1 shows the comparison results between our proposed model and other baseline models. It can be observed that our model achieves the best performance on the various variant datasets of PolitiHop and FEVER. On the PolitiHop, each sample contains an average of 4 pieces of evidence and 24 non-evidence sentences, which shows that Causal Walk has better multi-hop reasoning ability. In addition, the performance of Causal Walk is more consistent on the PolitiHop, Adversarial PolitiHop, and Hard PolitiHop. This proves that our model is not confused by misleading evidence and is more robust to data bias. On the FEVER and its variants, the performance of GEAR, KGAT, and Causal Walk on the multi-hop dataset (FEVER-MH) is significantly better than other models, which proves that

Models	PolitiHop	Adversarial PolitiHop	Hard PolitiHop	FEVER	FEVER-MH	Adversarial FEVER	Adversarial FEVER-MH
BERT-Concat	76.00	74.50	71.50	82.14	86.32	59.08	62.12
GEAR	75.50	75.00	73.50	86.58	87.04	57.81	63.36
KGAT	77.00	74.50	74.00	86.74	89.90	59.34	64.85
Transformer-XH	75.50	77.00	72.34	83.11	86.58	59.39	64.36
CICR	76.00	74.50	75.00	79.25	83.37	61.88	64.11
CLEVER	76.00	76.00	73.00	78.68	82.50	59.85	64.36
CICR-graph	78.00	77.50	76.50	87.38	91.53	59.21	65.84
CLEVER-graph	78.00	76.50	75.50	86.24	89.99	59.47	64.60
Causal Walk	**80.00**	**79.00**	**79.00**	**90.19**	**92.88**	**62.13**	**67.08**

Table 1: Experimental results for PolitiHop dataset, FEVER dataset, and their variants. The best results are in bold.

Models	PolitiHop	Adversarial PolitiHop	Hard PolitiHop	FEVER	FEVER-MH	Adversarial FEVER	Adversarial FEVER-MH
Causal Walk	**80.00**	**79.00**	**79.00**	**90.19**	**92.88**	**62.13**	**67.08**
w/o intervention	77.00	78.00	77.00	87.86	90.69	60.86	65.35
w/ evidence label	78.00	77.00	77.00	89.48	91.87	59.72	64.36

Table 2: Experimental results for ablation study. The best results are in bold.

Models	PolitiHop -adv	Symmetric	Symmetric -ShareEvi
GEAR	**89.47**	51.17	50.88
CICR-graph	87.72	51.75	50.58
CLEVER-graph	86.55	52.05	53.22
Causal Walk	88.30	**57.02**	**54.09**

Table 3: Experimental results on Symmetric PolitiHop dataset. The best results are in bold.

the graph-based model has strong multi-hop reasoning ability. The performance of CICR, CLEVER, and Causal Walk on Adversarial FEVER data set is significantly better than other models, which proves that the causal-based model has strong robustness. On the multi-hop Adversarial FEVER-MH dataset, our model still has the best performance, proving that our method has all the advantages of both graph-based and causal-based models.

Table 3 shows the performance of the model on the symmetric dataset. It can be observed that our model achieves the best performance on both Symmetric and Symmetric-ShareEvi.

Ablation Study

To investigate the effectiveness of our proposed causal intervention approach, we evaluate the performance of the model after removing the causal intervention part (w/o intervention), that is, directly using the result $\boldsymbol{l}_{pred}$ of Equation (21) as the final classification result. As shown in Table 2, the performance decreases consistently on both the original dataset and the adversarial dataset, which illustrates the effectiveness of the causal intervention.

The evidence set contains ground-truth evidence and non-evidence sentences. Both PolitiHop and FEVER provide the evidence labels. During training, we use these evidence labels to supervise the transition probability matrix in Equation (7) (w/ evidence label). Counterintuitively, the performance of the model decreases after using the evidence labels. The performance decreases less on PolitiHop and FEVER, and more on the Adversarial dataset. We believe that this is because the proportion of evidence sentences and non-evidence sentences in the training data is very small, so the transition probability matrix will converge to a very sparse state, which will lead to serious error accumulation in path sampling on the out-of-distribution data set. From another point of view, in the front-door adjustment, we need to sample the paths multiple times and calculate the weighted sum. For the diversity of path sampling, we want the transition probability matrix to be dense rather than sparse.

Conclusion

In this paper, we propose Causal Walk to debias multi-hop fact verification based on front-door adjustment. The reasoning path of the claim-evidence graph is introduced as the mediator between input and output to perform the front-door adjustment. Specifically, the causal effect between input and output is decomposed into two parts, the causal effect between input and mediator and the causal effect between mediator and output. The former part is estimated by combining the idea of random walk and graph neural network while the latter part is estimated by introducing NWGM approximation into recurrent neural network. What's more, we also extend existing datasets with the help of large language models to further test the proposed method. The experimental results for both existing and new datasets show the effectiveness of the proposed method.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments. This work is funded by the National Natural Science Foundation of China (62176053). This work is supported by the Big Data Computing Center of Southeast University.

References

Chen, Z.; Hu, L.; Li, W.; Shao, Y.; and Nie, L. 2023. Causal Intervention and Counterfactual Reasoning for Multi-modal Fake News Detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 627–638. Toronto, Canada: Association for Computational Linguistics.

Chen, Z.; Hui, S. C.; Zhuang, F.; Liao, L.; Li, F.; Jia, M.; and Li, J. 2022. EvidenceNet: Evidence Fusion Network for Fact Verification. In *Proceedings of the ACM Web Conference 2022*, WWW '22, 2636–2645. New York, NY, USA: Association for Computing Machinery. ISBN 9781450390965.

Christopoulou, F.; Miwa, M.; and Ananiadou, S. 2018. A Walk-based Model on Entity Graphs for Relation Extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 81–88. Melbourne, Australia: Association for Computational Linguistics.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Fajcik, M.; Motlicek, P.; and Smrz, P. 2023. Claim-Dissector: An Interpretable Fact-Checking System with Joint Re-ranking and Veracity Prediction. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Findings of the Association for Computational Linguistics: ACL 2023*, 10184–10205. Toronto, Canada: Association for Computational Linguistics.

Guo, W.; Gong, Q.; Rao, Y.; and Lai, H. 2023. Counterfactual Multihop QA: A Cause-Effect Approach for Reducing Disconnected Reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 4214–4226. Toronto, Canada: Association for Computational Linguistics.

Hanselowski, A.; Zhang, H.; Li, Z.; Sorokin, D.; Schiller, B.; Schulz, C.; and Gurevych, I. 2018. UKP-Athene: Multi-Sentence Textual Entailment for Claim Verification. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, 103–108. Brussels, Belgium: Association for Computational Linguistics.

Karimi Mahabadi, R.; Belinkov, Y.; and Henderson, J. 2020. End-to-End Bias Mitigation by Modelling Biases in Corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8706–8716. Online: Association for Computational Linguistics.

Lee, M.; Won, S.; Kim, J.; Lee, H.; Park, C.; and Jung, K. 2021. CrossAug: A Contrastive Data Augmentation Method for Debiasing Fact Verification Models. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, 3181–3185. New York, NY, USA: Association for Computing Machinery. ISBN 9781450384469.

Li, J.; Zhu, J.; and Zhang, B. 2016. Discriminative Deep Random Walk for Network Classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1004–1013. Berlin, Germany: Association for Computational Linguistics.

Liu, B.; Wang, D.; Yang, X.; Zhou, Y.; Yao, R.; Shao, Z.; and Zhao, J. 2022. Show, Deconfound and Tell: Image Captioning With Causal Inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 18041–18050.

Liu, Z.; Xiong, C.; Sun, M.; and Liu, Z. 2020. Fine-grained Fact Verification with Kernel Graph Attention Network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7342–7351. Online: Association for Computational Linguistics.

Nie, Y.; Chen, H.; and Bansal, M. 2019. Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 6859–6866.

Niu, Y.; Tang, K.; Zhang, H.; Lu, Z.; Hua, X.-S.; and Wen, J.-R. 2021. Counterfactual vqa: A cause-effect look at language bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12700–12710.

OpenAI. 2023. GPT-4 Technical Report. https://openai.com/research/gpt-4. Accessed: 2023-07-28, arXiv:2303.08774.

Ostrowski, W.; Arora, A.; Atanasova, P.; and Augenstein, I. 2021. Multi-Hop Fact Checking of Political Claims. In Zhou, Z.-H., ed., *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, 3892–3898. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Pan, L.; Wu, X.; Lu, X.; Luu, A. T.; Wang, W. Y.; Kan, M.-Y.; and Nakov, P. 2023. Fact-Checking Complex Claims with Program-Guided Reasoning. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 6981–7004. Toronto, Canada: Association for Computational Linguistics.

Schuster, T.; Shah, D.; Yeo, Y. J. S.; Roberto Filizzola Ortiz, D.; Santus, E.; and Barzilay, R. 2019. Towards Debiasing Fact Verification Models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3419–3425. Hong Kong, China: Association for Computational Linguistics.

Si, J.; Zhu, Y.; and Zhou, D. 2023. Exploring faithful rationale for multi-hop fact verification via salience-aware graph

learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 13573–13581.

Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; and Mittal, A. 2018. FEVER: a Large-scale Dataset for Fact Extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 809–819. New Orleans, Louisiana: Association for Computational Linguistics.

Tian, B.; Cao, Y.; Zhang, Y.; and Xing, C. 2022a. Debiasing NLU Models via Causal Intervention and Counterfactual Reasoning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10): 11376–11384.

Tian, B.; Cao, Y.; Zhang, Y.; and Xing, C. 2022b. Debiasing NLU models via causal intervention and counterfactual reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 11376–11384.

Trivedi, H.; Balasubramanian, N.; Khot, T.; and Sabharwal, A. 2020. Is Multihop QA in DiRe Condition? Measuring and Reducing Disconnected Reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 8846–8863. Online: Association for Computational Linguistics.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Wang, S.; Zhou, J.; Sun, C.; Ye, J.; Gui, T.; Zhang, Q.; and Huang, X. 2022. Causal Intervention Improves Implicit Sentiment Analysis. In *Proceedings of the 29th International Conference on Computational Linguistics*, 6966–6977. Gyeongju, Republic of Korea: International Committee on Computational Linguistics.

Wei, J.; and Zou, K. 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 6382–6388. Hong Kong, China: Association for Computational Linguistics.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; and Bengio, Y. 2015. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *International Conference on Machine Learning,International Conference on Machine Learning*.

Xu, W.; Liu, Q.; Wu, S.; and Wang, L. 2023. Counterfactual Debiasing for Fact Verification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 6777–6789. Toronto, Canada: Association for Computational Linguistics.

Yang, X.; Zhang, H.; and Cai, J. 2021. Deconfounded image captioning: A causal retrospect. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yang, X.; Zhang, H.; Qi, G.; and Cai, J. 2021. Causal attention for vision-language tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9847–9857.

Zeng, F.; and Gao, W. 2023. Prompt to be Consistent is Better than Self-Consistent? Few-Shot and Zero-Shot Fact Verification with Pre-trained Language Models. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Findings of the Association for Computational Linguistics: ACL 2023*, 4555–4569. Toronto, Canada: Association for Computational Linguistics.

Zhao, C.; Xiong, C.; Rosset, C.; Song, X.; Bennett, P.; and Tiwary, S. 2020a. Transformer-XH: Multi-Evidence Reasoning with eXtra Hop Attention. In *International Conference on Learning Representations*.

Zhao, C.; Xiong, C.; Rosset, C.; Song, X.; Bennett, P.; and Tiwary, S. 2020b. Transformer-XH: Multi-evidence Reasoning with Extra Hop Attention. In *The Eighth International Conference on Learning Representations (ICLR 2020)*.

Zhong, W.; Xu, J.; Tang, D.; Xu, Z.; Duan, N.; Zhou, M.; Wang, J.; and Yin, J. 2020. Reasoning Over Semantic-Level Graph for Fact Checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6170–6180. Online: Association for Computational Linguistics.

Zhou, J.; Han, X.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; and Sun, M. 2019. GEAR: Graph-based Evidence Aggregating and Reasoning for Fact Verification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 892–901. Florence, Italy: Association for Computational Linguistics.

Zhu, J.; Wu, S.; Zhang, X.; Hou, Y.; and Feng, Z. 2023. Causal Intervention for Mitigating Name Bias in Machine Reading Comprehension. In *Findings of the Association for Computational Linguistics: ACL 2023*, 12837–12852. Toronto, Canada: Association for Computational Linguistics.