# GENERALIZABLE MULTI-RELATIONAL GRAPH REPRESENTATION LEARNING: A MESSAGE INTERVENTION APPROACH

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

With the edges associated with labels and directions, the so-called multi-relational graph possesses powerful expressiveness, which is beneficial to many applications. However, as the heterogeneity brought by the higher cardinality of edges and relations climbs up, more trivial relations are taken into account for the downstream task since they are often highly correlated to the target. As a result, with being forced to fit the non-causal relational patterns on the training set, the downstream model, like graph neural network (GNN), may suffer from poor generalizability on the testing set since the inference is mainly made according to misleading clues. In this paper, under the paradigm of graph convolution, we probe the multi-relational message passing process from the perspective of causality and then propose a Message Intervention method for learning generalizable muLtirElational gRaph representations, coined MILER. In particular, MILER first encodes the vertices and relations into embeddings with relational and directional awareness, then a message diverter is employed to split the original message flow into two flows of interest, *i.e.,* the causal and trivial message flows. Afterward, the message intervention is carried out with the guidance of the backdoor adjustment rule. Extensive experiments on several knowledge graph benchmarks validate the effectiveness as well as the superior generalization ability of MILER.

## 1 INTRODUCTION

Multi-relational graphs (MRGs) are a family of graphs where the edges are associated with labels and directions. MRGs differentiate themselves by the heterogeneity of edges. Numerous research efforts (Schlichtkrull et al., 2018; Bordes et al., 2013; Vashishth et al., 2020; Dettmers et al., 2018) have been made to efficiently integrate the ample heterogeneous knowledge and learn more expressive representations of the graph components, such as vertices and edges. As a concrete type of MRGs, Knowledge Graphs (KGs) have been applied to various downstream applications with the help of multi-relational graph representation learning (MRGRL), such as information retrieval (Shen et al., 2022), question answering (Qiu et al., 2020), and semantic matching (Wang et al., 2022).

In the literature, a majority of works focusing on MRGRL lies in two threads. The first line of works (Bordes et al., 2013; Yang et al., 2015; Trouillon et al., 2016; Dettmers et al., 2018) has paid attention to embedding knowledge graphs by vectorizing the entities and relations, and learning low-dimensional representations under specific optimization criterion (*e.g.,* translation). These methods mostly ignore the structural information that could bring benefits to the representation learning. The other line of works counts on the graph neural networks to capture the structural properties of the multi-relational graph. The main idea of these works is to bring relation awareness into the graph convolution process. For example, Schlichtkrull et al. (2018); Shang et al. (2019) utilized relation-specific filters to distinguish different relation types during convolution. Vashishth et al. (2020); Ye et al. (2019); Chen et al. (2022) attempted to encode vertices together with the relations to learn more comprehensive representations and alleviate the over-parameterization problem.

Nevertheless, with the increase of heterogeneity in multi-relational graphs, the generalization issue is worth pondering. Intuitively, the expressiveness of a multi-relational graph is expected to grow with the diversity of relations because of the enriched knowledge. This is, however, not always

tenable as not all the relations are truly useful for the final task. Parts of the relations could establish a spurious correlation with the downstream task in the training phase, which probably leads to poor generalizability in the inference phase. For example, consider a training query *(Mission Impossible II, language, ?)* and a test query *(Crouching Tiger, Hidden Dragon, language, ?)*, which are two film-related queries based on the knowledge graph. In the training stage, several relational clues can be involved to infer this query, such as *genre*, *country*, and *release year*, where the *country* is supposed to be the rationale for answering this query. Unfortunately, if some trivial relation types, say, *release year*, have a high correlation with the query in the training set, the model would tend to make the prediction based on these relations instead of those that really matter to the answer. Consequently, when it comes to the inference, the model could possibly follow the patterns in the training set and mistakenly answer this query as: the *language* of *Crouching Tiger, Hidden Dragon* is *English*, whereas the movie is a Chinese movie which happens to have the same release year as *Mission Impossible II*.

How to overcome the generalization issue brought by the heterogeneity in MRGs is indeed challenging. Firstly, with the model naturally being forced to fit the training data (Chang et al., 2020), it is non-trivial to explore which relations really account for the task, *i.e.,* should generalize to the test set. Besides, different vertices have different relational contexts, and the crucial relations cannot be determined in a general way. Toward this end, we have investigated the causal story of multi-relational message passing, where the trivial relational message acts as a confounder between the final prediction and the causal relational message. In order to remedy the backdoor path opened by the confounder, we propose a Message Intervention method for learning generalizable muLti-rElational gRaph representations, named MILER. Specifically, we first hire a composition-based encoder to encode the vertices and relations with directional and relational awareness. Then, for each relation, we propose a message diverter to split the original message flow into two flows of interest, *i.e.,* the causal and trivial message flows, with a learnable causal gate. Moreover, instructed by the backdoor adjustment rule, we formulate the optimization objective to estimate the interventional distribution, and carry out the intervention with the trivial classifier (scorer) from a message tank acting on the causal classifier (scorer). To sum up, the contributions of this work are as follows:

- To the best of our knowledge, we are among the first to study the generalization issue brought by the heterogeneity on multi-relational graphs. We actually inspect the multi-relational message passing process with the help of causal inference.
- We propose a generalizable multi-relational graph representation learning approach via message intervention, called MILER.
- Extensive experiments on multiple knowledge graph benchmark tasks validate the effectiveness of MILER. In addition, we demonstrate that MILER can effectively alleviate the generalization issue and deliver human understandable interpretability.

## 2 NOTATIONS AND TASK FORMULATION

**Notations.** We denote a multi-relational graph as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{R}, \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Z}}\}$, where $\mathcal{V}$ and $\mathcal{R}$ are the set of vertices and relations, respectively, $\mathcal{E} = \{(u, r, v)|u, v \in \mathcal{V}, r \in \mathcal{R}\}$ is the set of edges in which each edge $(u, r, v)$ indicates that there is a relation $r$ from vertex $u$ to $v$, and $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Z}}$ denote the initial representations of vertices and relations, which can either be randomly initialized or filled with semantic features. Similar to (Marcheggiani & Titov, 2017; Vashishth et al., 2020), we extend $\mathcal{E}$ and $\mathcal{R}$ with corresponding inverse edges and relations to enable the bidirectional flow of messages as follows:

$$\widetilde{\mathcal{E}} = \mathcal{E} \cup \{(v, r^{-1}, u)|(u, r, v) \in \mathcal{E}\} \cup \{(u, \rho, u)|u \in \mathcal{V}\}, \widetilde{\mathcal{R}} = \mathcal{R} \cup \mathcal{R}_{inv} \cup \rho,$$

where $\widetilde{\mathcal{R}}$ denotes the extended relation set of $\mathcal{G}$, $\mathcal{R}_{inv} = \{r^{-1}|r \in \mathcal{R}\}$ and $\rho$ is the self-loop relation.

**Task Formulation.** Given a multi-relational graph $\mathcal{G}$, the goal is to learn the representations of both vertices and relations toward different downstream tasks (*e.g.,* link prediction and node classification). As the heterogeneity could make the model excessively concentrate on some specific relations due to the high but spurious correlation to the target and thus lead to poor generalizability, the learned representations are also expected to reveal the decisive relational messages to the final prediction, and generalize well to the test set.

# 3 A CAUSAL GLIMPSE OF MULTI-RELATIONAL MESSAGE PASSING

## 3.1 STRUCTURAL CAUSAL MODEL

Generally, from the perspective of spatial-based convolution (Bruna et al., 2014; Hamilton et al., 2017; Gilmer et al., 2017), when performing the message passing on the multi-relational graph, the target vertex will accept messages from neighbor vertices through different relations. However, not all the relational messages contribute to the final prediction. Some relations may be mistakenly involved in the prediction only because they have a high correlation but not actual causation. Incorporating such misleading messages could result in poor generalizability as spurious correlations could be established between the objective and these trivial relational messages. To further probe the lying causality, as shown in Figure 1, we formulate the multi-relational message passing process with the Structural Causal Model (SCM) (Pearl et al., 2016).

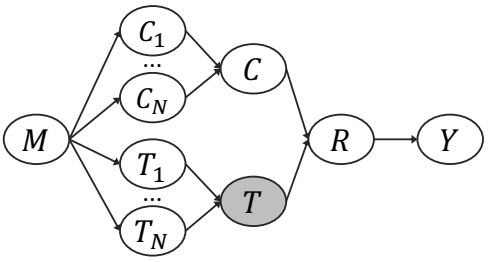



Figure 1: The Structural Causal Model of multi-relational message passing process.

$\boldsymbol{M \rightarrow C_i}$ **and** $\boldsymbol{M \rightarrow T_i}$. We denote variable $M$ as the messages to be accepted by the target vertex, and $C_i$ and $T_i$ ($i = 1, \ldots, N, N = |\widetilde{\mathcal{R}}|$) as the causal message and trivial message *w.r.t.* the $i$-th relation $r_i$. These two types of causal paths severally indicate that the messages can be divided by the relation.

$\boldsymbol{T_i \leftarrow M \rightarrow C_i}$. This link represents that within a specific relation, the message consists of the causal and trivial parts. The causal part is the message that plays a decisive role in the prediction, while the trivial part is the redundant message that could impair the generalizability of the model.

$\boldsymbol{C_i \rightarrow C \rightarrow R \leftarrow T \leftarrow T_i}$. We denote $R$ as the representation of the target vertex after aggregating the message, $C$ as the causal component, and $T$ as the trivial component. This link shows that the aggregated representation is made of causal and trivial components, which mix messages from different relations, respectively.

$\boldsymbol{R \rightarrow Y}$. With variable $Y$ denoting the final prediction, this link tells an acknowledged fact that the model will make the prediction based on the learned representations.

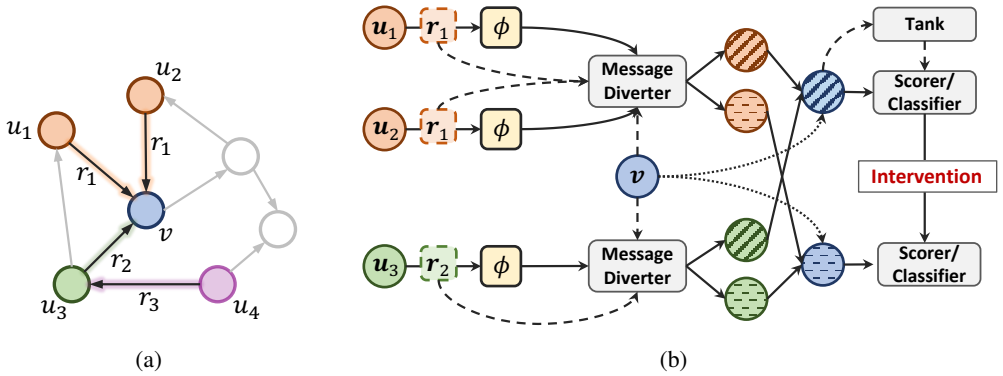
When we look at the outcome of the model, we actually expect the prediction is directly inferred from the causal component (*i.e.,* $C \rightarrow R \rightarrow Y$). Unfortunately, the trivial messages ($T_i$ and $T$) open an undesired backdoor path between $C$ and $Y$. If we want to estimate the causal effect of $C$ on $Y$, we need to find a feasible way to eliminate the causal effect through the backdoor path.

## 3.2 BACKDOOR ADJUSTMENT

In Section 3.1, we noticed that the conditional probability $P(Y|\{C_i\})$ that we naturally estimate is confounded by the confounder $T$. According to the *do-calculus* (Pearl et al., 2016), what we really desire to estimate is the interventional probability $P(Y|do(\{C_i\}))$. Apparently, $P(Y|\{C_i\}) \neq P(Y|do(\{C_i\}))$. Intuitively, if we condition on variable $T$, all the backdoor paths between $C$ and $R$ are blocked. In other words, $C$ and $R$ are *d-separated* by $T$. Formally, as derived in Appendix A.1, the backdoor adjustment is given by:

$$P(Y|do(\{C_i\})) = \sum_t P(Y|\{C_i\}, T = t)\, P(T = t), \quad i = 1, 2, \ldots, |\widetilde{\mathcal{R}}|. \tag{1}$$

In Section 4.1, we will introduce how we implement such adjustment to get rid of the negative impact of the trivial component on the prediction.




(a) (b)

Figure 2: (a) An illustration of the multi-relational graph. (b) The overview of our proposed MILER. The nodes with tilted solid lines denote the trivial component representation, while those with dashed lines denote the causal component representation.

## 4 METHODOLOGY

The overview of the architecture is illustrated in Figure 2. In Section 4.1, we will introduce MILER following single-layer message passing formulation for clarity, and the generalized MILER toward multi-layer convolution will be introduced in Section 4.2.

### 4.1 MILER

**Encoder.** To leverage the heterogeneity in multi-relational graphs, several works have explored the representation of both vertices and relations (Vashishth et al., 2020; Schlichtkrull et al., 2018; Shang et al., 2019). Without loss of generality, we take the composition-based operator as the encoder, which has been proved able to generalize to several representative multi-relational GNN methods (Vashishth et al., 2020).

Specifically, given a vertex $u \in \mathcal{V}$, based on the multi-relational graph $\mathcal{G}$, $u$ can be encoded as:

$$\boldsymbol{e}_u = \boldsymbol{W}_{dir(r)} \phi \left( \boldsymbol{x}_u, \boldsymbol{z}_r \right), \tag{2}$$

where $\boldsymbol{x}_u$ is the initial representation of vertex $u$, $\boldsymbol{z}_r$ is the initial representation of relation $r$, $\phi$ is the composition operator which can be instantiated as, say, subtraction, multiplication and circular-correlation, and $\boldsymbol{W}_{dir(r)}$ is a direction-aware learnable weight which corresponds to independent weights *w.r.t.* different relational directions, *i.e.,*

$$\boldsymbol{W}_{dir(r)} = \begin{cases} \boldsymbol{W}_O, & r \in \mathcal{R} \\ \boldsymbol{W}_I, & r \in \mathcal{R}_{inv} \\ \boldsymbol{W}_L & r = \rho. \end{cases}$$

**Message Diverter.** To enable us to study the causal effect of the causal and trivial messages, we propose a message diverter to split the message flow into causal message flow and trivial message flow under a specific relation.

Specifically, given a target vertex $v$ and its $r$-neighbors $\mathcal{N}_r(v) = \{u | (u, r, v) \in \tilde{\mathcal{E}}\}$, we first aggregate the messages from the neighbors within relation $r$ as:

$$\boldsymbol{g}_v^r = \sum_{u \in \mathcal{N}_r(v)} \frac{1}{\mu_{u,v}^r} \boldsymbol{e}_u, \tag{3}$$

where $\mu_{u,v}^r$ is a normalization constant and we choose $\mu_{u,v}^r = \sqrt{|\mathcal{N}(u)||\mathcal{N}(v)|}$ following (Vashishth et al., 2020). Then, we employ a causal gate to determine how much message of relation $r$ (denoted as $r$-message) should be spared for the causal flow:

$$\alpha_v^r = \sigma \left( \boldsymbol{a}^\mathsf{T} \left[ \boldsymbol{W}_C \boldsymbol{z}_r \parallel \boldsymbol{W}_C \boldsymbol{x}_v \right] \right), \tag{4}$$

where $\boldsymbol{W}_C$ and $\boldsymbol{a}$ are two trainable parameters, $[\cdot \parallel \cdot]$ is the concatenation operation, $\sigma$ is the sigmoid function, and $\alpha_v^r$ is the derived causal coefficient that controls the ratio of message flowing into the causal flow. With the help of the causal gate, we can further obtain the causal and trivial $r$-messages, respectively, as follows:

$$\boldsymbol{m}_v^r = \alpha_v^r \boldsymbol{g}_v^r, \quad \widetilde{\boldsymbol{m}}_v^r = (1 - \alpha_v^r)\boldsymbol{g}_v^r, \tag{5}$$

where $\boldsymbol{m}_v^r$ is vertex $v$'s causal $r$-message and $\widetilde{\boldsymbol{m}}_v^r$ is its trivial $r$-message.

**Message Receiving and Utilizing.** After respectively acquiring the causal and trivial $r$-message, we let the target vertex receive messages from all relations in two flows as below:

$$\boldsymbol{h}_v = g\left(\sum_r \boldsymbol{m}_v^r\right), \quad \widetilde{\boldsymbol{h}}_v = g\left(\sum_r \widetilde{\boldsymbol{m}}_v^r\right), \tag{6}$$

where $g$ is the activation function, and $\boldsymbol{h}_v$ and $\widetilde{\boldsymbol{h}}_v$ are the causal component and trivial component of vertex $v$'s representation, respectively. Besides, we reserve the trivial component into a message tank $\mathcal{T}$ for future use.

To utilize the representations for final prediction, we first use two separate scorers (or classifiers) $f_C$ and $f_T$ to take the causal/trivial component representations as input to make respective predictions. Note that, $f_C$ and $f_T$ vary according to the downstream task. For instance, in the link prediction task, the scorer can be but is not limited to ConvE (Dettmers et al., 2018), DistMult (Yang et al., 2015) or TransE (Bordes et al., 2013), while in the node classification task, the classifier can be an MLP. For brevity, we will take node classification as the example below.

**Optimization.** From Section 3.2, we have realized that the distribution we need to estimate is the interventional distribution $P(Y|do(\{C_i\}))$. Guided by the backdoor adjustment rule in Equation (1), we define our optimization objective as:

$$\max_{\Theta} \ \mathbb{E}_{(m,y),t}\Big[\log P_\Theta(Y|\{C_i\}, T=t)\Big], \quad i = 1, 2, \ldots, |\widetilde{\mathcal{R}}|$$

where $(m, y)$ denotes the tuple of the message to be received of a given vertex and its corresponding label, and $\Theta$ denotes the parameters in MILER. Relevant proofs can be found in Appendix A.2.

Besides, as the trivial messages are not supposed to make a difference to the prediction, we further add a constraint to the above objective to better shield the prediction from the trivial messages, and rewrite it as follows:

$$\begin{aligned} &\max_{\Theta} \ \mathbb{E}_{(m,y),t}\Big[\log P_\Theta(Y|\{C_i\}, T=t)\Big], \\ s.t. \ &\mathbb{E}_{(m,y)}\Big[\mathbb{D}_t\left[P_\Theta(Y|\{C_i\}, T=t)\right]\Big] \leq \epsilon, \quad i = 1, 2, \ldots, |\widetilde{\mathcal{R}}|, \end{aligned} \tag{7}$$

where $\mathbb{D}$ is the variance of a probability distribution, and $\epsilon$ is a small positive constant.

**Intervention and Its Implementation.** Inspired by (Wu et al., 2022; Cadène et al., 2019), to achieve the intervention, we instantiate the distribution $P_\Theta(Y|\{C_i\}, T=t)$ as:

$$P_\Theta(Y|\{C_i\}, T=t) = f_C \odot \sigma(f_T(\widetilde{\boldsymbol{h}}_t)), \quad \widetilde{\boldsymbol{h}}_t \in \mathcal{T}, \ i = 1, 2, \ldots, |\widetilde{\mathcal{R}}|, \tag{8}$$

where $\odot$ is the Hadamard production. Besides, given a vertex with its label $y$, we here employ a sole optimizer to train the parameters of $f_T$ below:

$$\min_{\Theta_T} \mathcal{L}\left(f_T, y\right), \tag{9}$$

where $\Theta_T$ denotes the parameters of scorer/classifier $f_T$, and $\mathcal{L}$ is the loss function. Note that, we let $\Theta_T$ be optimized by and only by Equation (9). The rationale behind the implementation is that once we punish the causal classifier with the trivial classifier that is forced to learn the ground truth, the training procedure would pay more attention to the causal flow instead of the trivial flow, which is beneficial for estimating the interventional probability.

Practically, we can jointly optimize the objectives in Equations (7) and (9). After the optimization, the message diverter is endowed with the ability to distinguish the causal and trivial parts of each $r$-message. Hence, we use the causal message split by the message diverter and its classifier $f_C$ to make the inference.

### 4.2 GENERALIZATION TO MULTI-LAYER CONVOLUTION

In this subsection, we introduce how to further generalize MILER to multi-layer convolution. When we try to capture the messages from higher-order neighbors, the roles that the causal and trivial messages play need to be reassessed. For example, consider the vertex chain $u_4 \rightarrow u_3 \rightarrow v$ in Figure 2(a) and a two-layer convolution operation. When $u_3$ is trying to send the causal and trivial messages accepted from $u_4$ to $v$ in the second layer, both the causal and trivial properties may not hold anymore. This is mainly because of the context change in different layers, *i.e.,* the context has changed from relation $r_3$ to $r_2$-$r_3$. Therefore, assuming a $K$-layer convolution, we generalize the intervention-based multi-relational message passing process as follows:

$$\begin{aligned} \boldsymbol{h}_v^{(k+1)} &= g\left(\sum_r \alpha_v^{r,k} \sum_{u \in \mathcal{N}_r(v)} \frac{1}{\mu_{u,v}^r} \boldsymbol{W}_{dir(r)}^k \phi\left(\boldsymbol{p}_u^k, \boldsymbol{h}_r^k\right)\right), \\ \widetilde{\boldsymbol{h}}_v^{(k+1)} &= g\left(\sum_r \left(1-\alpha_v^{r,k}\right) \sum_{u \in \mathcal{N}_r(v)} \frac{1}{\mu_{u,v}^r} \boldsymbol{W}_{dir(r)}^k \phi\left(\boldsymbol{p}_u^k, \boldsymbol{h}_r^k\right)\right), \end{aligned} \tag{10}$$

$$\alpha_v^{r,k} = \sigma\left(\boldsymbol{a}^{k\mathsf{T}}\left[\boldsymbol{W}_C^k \boldsymbol{h}_r^k \parallel \boldsymbol{W}_C^k \boldsymbol{p}_v^k\right]\right), \tag{11}$$

$$\boldsymbol{h}_r^{(k+1)} = \boldsymbol{W}_{rel}^k \boldsymbol{h}_r^k, \tag{12}$$

$$\boldsymbol{p}_u^k = \begin{cases} \psi^k\left(\left[\boldsymbol{h}_u^k \parallel \widetilde{\boldsymbol{h}}_u^k\right]\right), & \text{if } k > 0 \\ \boldsymbol{h}_u^0, & \text{if } k = 0. \end{cases} \tag{13}$$

where $\boldsymbol{h}_u^0 = \boldsymbol{x}_u$, $\boldsymbol{h}_r^0 = \boldsymbol{z}_r$, $\boldsymbol{W}_{rel}$ is a learnable parameter, and $\psi$ is an MLP to model the non-linear interactions between causal and trivial messages towards higher-order context. After we obtain the causal and trivial component representation $\boldsymbol{h}_v^K$ and $\widetilde{\boldsymbol{h}}_v^K$ from the last layer, we can perform the same optimization and intervention strategy as stated in Section 4.1.

## 5 EXPERIMENTS

### 5.1 SETUPS

**Downstream Tasks for Evaluation.** We evaluate MILER with two representative downstream tasks including link prediction and node classification:

- **Link Prediction.** This task aims to infer missing edges in a multi-relational graph, which correspond to the missing facts in a knowledge graph. We use two widely-adopted knowledge graph benchmarks in our experiments: FB15k-237 (Toutanova & Chen, 2015) and WN18RR (Dettmers et al., 2018). Besides, the metrics adopted for evaluation are Mean Reciprocal Rank (MRR), Mean Rank (MR) and Hits@N. Following (Bordes et al., 2013), the results are reported under the filtered setting.
- **Node Classification.** This task is to predict the labels of nodes within a multi-relational graph based on the graph structure and the node features (or relations). We evaluate the performance on three RDF-format datasets (Ristoski et al., 2016) including AIFB, MUTAG, and BGS, in terms of Accuracy metric.

**Baselines.** For link prediction, we compare MILER against five non-GNN methods (*i.e.,* TransE (Bordes et al., 2013), DistMult(Yang et al., 2015), ComplEx (Trouillon et al., 2016), ConvE (Dettmers et al., 2018) and RotatE (Sun et al., 2019)), as well as four GNN-based methods (*i.e.,* RGCN (Schlichtkrull et al., 2018), SACN (Shang et al., 2019), VR-GCN (Ye et al., 2019) and CompGCN (Vashishth et al., 2020)). For the node classification task, we take four algorithms as competitors including Feat (Paulheim & Fümkranz, 2012), RDF2Vec (Ristoski & Paulheim, 2016), RGCN (Schlichtkrull et al., 2018), and CompGCN (Vashishth et al., 2020).

The dataset description and implementation details can be found in Appendix B.1 and Appendix B.2, respectively.

| | FB15k-237 | | | | | WN18RR | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | MRR | MR | Hits@1 | Hit@3 | Hits@10 | MRR | MR | Hits@1 | Hit@3 | Hits@10 |
| TransE | 0.294 | 357 | - | - | 0.465 | 0.226 | 3384 | - | - | 0.501 |
| DistMult | 0.241 | 254 | 0.155 | 0.263 | 0.419 | 0.43 | 5110 | 0.39 | 0.44 | 0.49 |
| ComplEx | 0.247 | 339 | 0.158 | 0.275 | 0.428 | 0.44 | 5261 | 0.41 | 0.45 | 0.51 |
| ConvE | 0.325 | 244 | 0.237 | 0.356 | 0.501 | 0.43 | 4187 | 0.40 | 0.44 | 0.52 |
| RotatE | 0.336 | **177** | 0.239 | 0.373 | **0.531** | **0.474** | 3340 | 0.426 | **0.491** | **0.571** |
| RGCN | 0.248 | - | 0.153 | 0.258 | 0.414 | - | - | - | - | - |
| VR-GCN | 0.248 | - | 0.159 | 0.272 | 0.432 | - | - | - | - | - |
| SACN | 0.339 | 203 | 0.249 | 0.373 | 0.521 | 0.429 | 3510 | 0.382 | 0.453 | 0.514 |
| CompGCN | 0.351 | 205 | 0.261 | 0.385 | 0.529 | 0.469 | 3273 | 0.436 | 0.482 | 0.534 |
| MILER | **0.353**$^{\dagger}$ | 217 | **0.263**$^{\dagger}$ | **0.387**$^{\dagger}$ | **0.531**$^{\dagger}$ | 0.471 | **3175** | **0.437** | 0.481 | 0.538 |

Table 1: Performance comparisons of link prediction on FB15k-237 and WN18RR datasets. The best scores are in boldface and the second best underlined. ($^{\dagger}$significantly outperform at 0.01 level)

| | AIFB | MUTAG | BGS |
|---|---|---|---|
| Feat | 55.55 | 77.94 | 72.41 |
| RDF2Vec | 88.88 | 67.20 | 87.24 |
| RGCN | 83.33 | 67.65 | 79.31 |
| CompGCN | **88.89** | 83.82 | 79.31 |
| MILER | **88.89** | **85.29** | **89.66** |

Table 2: Performance comparisons of node classification on datasets AIFB, MUTAG and BGS.

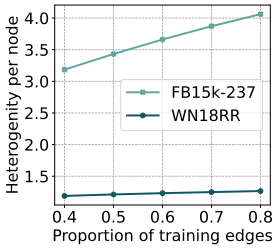



Figure 3: The variation of neighbor heterogeneity against the size of training set.

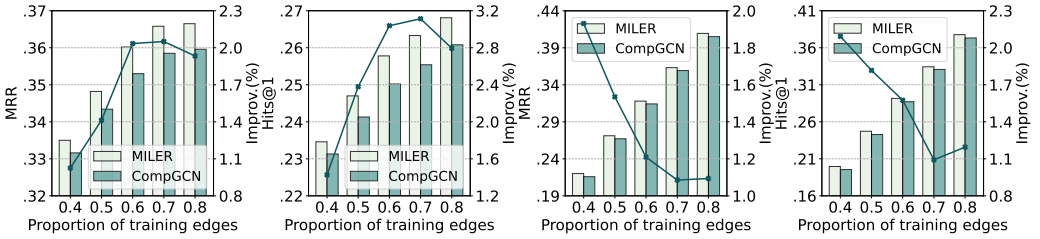
### 5.2 Main Observations

**Overall Performance.** For link prediction, the scores of the baselines (except RotatE, SACN, and CompGCN) are taken from previous papers directly. From Table 1, we can observe that MILER performs the best *w.r.t.* 4 out of 5 metrics on FB15k-237, and the best and 2nd-best *w.r.t.* 2 out of 5 metrics on WN18RR, respectively. Besides, MILER outperforms CompGCN (*i.e.,* the base model of MILER), which demonstrates that intervening in the message passing and picking those crucial relations indeed improves the generalizability of the predictive GNN method. Note that, compared to RotatE, the lower performance of MILER on WN18RR *w.r.t.* a few metrics are probably because of the different compositional operators. That is, MILER uses circular correlation as the operator, while RotatE employs rotation operation. Since the choice of the operators is not the key point for discussion in this work, we will leave this in our future work.

For the node classification task, CompGCN and RGCN are reimplemented, while the results of others are from previous works. As shown in Table 2, MILER achieves the best results on MUTAG and BGS datasets while tying with CompGCN on AIFB dataset.

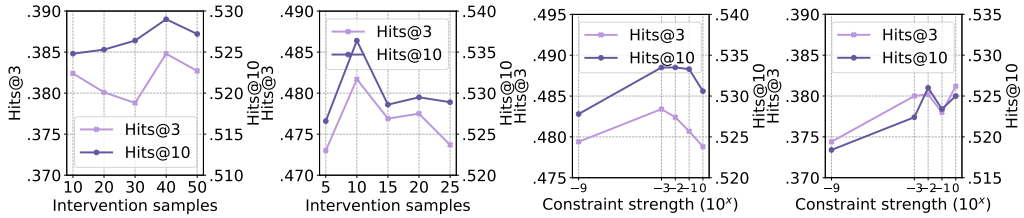
**Generalizability.** We evaluate the link prediction performance of MILER and CompGCN on FB15k-237 and WN18RR datasets by varying the available training set, such that the generalizability improvement of MILER can be verified accordingly. The detailed training set construction can be seen in Appendix B.3. The performance of MILER and CompGCN, as well as the improvement of MILER over CompGCN, are plotted in Figure 4. We can see that, the generalizability of both models is weakened when the training data is limited, while MILER firmly outperforms CompGCN under different settings, which verifies the superiority of MILER in mitigating the overfitting defect.

In addition, we have also noticed two totally different trends that the improvement of MILER over CompGCN increases as the volume of training edges on FB15k-237 dataset climbs up, while the opposite is observed on WN18RR dataset. To explain this difference, in Figure 3, we examine the heterogeneity variations of these two datasets by counting how many neighbor relation types exist for each node on average. As can be seen, for FB15k-237 dataset, when more training data is available, the node's neighborhood becomes more heterogeneous, which may lower the generalization




(a) MRR on FB15k-237 (b) Hits@1 on FB15k-237 (c) MRR on WN18RR (d) Hits@1 on WN18RR

Figure 4: The performance variation of MILER and CompGCN under different proportions of training edges on two datasets. The improvement percentage is shown by the broken line.




(a) FB15k-237 (b) WN18RR (c) FB15k-237 (d) WN18RR

Figure 5: Parameter sensitivity analysis of MILER.

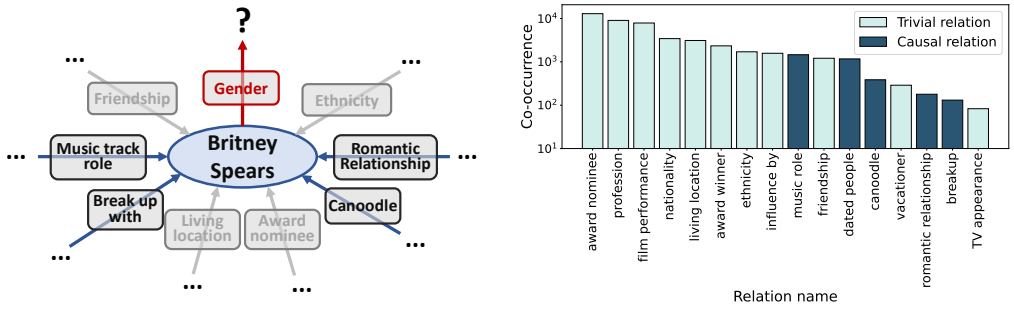
capability of the downstream model and even counteract the overfitting alleviation that has benefited from the increase of training set. Subsequently, the effectiveness of MILER alleviating trivial relations in attacking the generalization issue would be paid more attention to. On the contrary, with the heterogeneous level keeping stable on WN18RR dataset, simply enlarging the training set can probably ease the overfitting issue without too much help from the message passing intervention.

**Parameter Sensitivity.** We first examine how the number of intervention samples from the message tank affects the link prediction performance on two datasets. In Figure 5(a), the performance peaks when the number of intervention samples is around 40, and in Figure 5(b), the peak occurs around 10. Either too small or too large a sample size could result in performance decline. The possible reason is that too small a sample size cannot fully unleash the power of discovering the crucial relations for better generalization, while too large a sample size can impair the model's fitting ability. Besides, too many intervention samples may also pose time and space efficiency issues. Therefore, it is important to find a satisfactory trade-off.

We also investigate the impact of the constraint in Equation (7), where we utilize the method of Lagrange multipliers to leverage such constraint into the optimization. We use a regularization hyperparameter to imply the constraint strength. As shown in Figure 5(c) and Figure 5(d), with the absence of the constraint (the regularization coefficient is tuned to be small), the performance is faced with a huge fall, which validates the effectiveness of the constraint learning. Also, the strength should not be set too large so as to avoid domination over the main optimization objective.

### 5.3 Case Study

In Figure 6, we showcase the prediction of MILER on the FB15k-237 dataset. In particular, MILER scores Britney being a female 0.9937, while CompGCN scores 0.9368. In Figure 6(a), we illustrate several relations that might be helpful for inference. We also highlight those relations that MILER considers as causal relations by the causal gate in the message diverter. Specifically, in this study, if the causal score is greater than 0.5, we see it as a causal relation. As can be seen, MILER has selected the evidence that is intuitively useful for the inference, such as people she has broken up with or had a romantic relationship with, since we can infer Britney's gender through these people's gender. In the meantime, MILER has shielded the inference from the relations such as friendship and living location, which have nothing to do with gender. Moreover, we track the co-occurrence of the gender relation and these evidential relations on the training set in Figure 6(b), which further proves that MILER made the prediction via causation instead of correlation.




(a) Inferring the gender of Britney Spears, an American singer.

(b) The co-occurrence of the target relation and evidential relations on the training set.

Figure 6: An illustration of the case study on FB15k-237 dataset.

## 6 Related Work

**Representation Learning for Multi-Relational Graph.** The multi-relation graph distinguishes itself from universal graphs by its heterogeneity of edges. A group of studies has worked on embedding the components in multi-relational graphs under the paradigm of Graph Convolutional Networks (GCN) (Kipf & Welling, 2017). Marcheggiani & Titov (2017) first proposed a directed GCN to model the syntactic dependency graphs by introducing direction-specific filters. Schlichtkrull et al. (2018) assigned a relation-specific weight to each relation and designed basis and block-diagonal decompositions to address the efficiency issue. Some latest works (Ye et al., 2019; Vashishth et al., 2020; Chen et al., 2022) also involved the relations into representation learning and achieved satisfying performance toward different tasks. In the meantime, the representation learning for multi-relational graph also emerges from another line of works known as knowledge graph embedding (KGE) by respectively regarding the vertices and relations as entities and facts in KG. Recent works on KGE can be categorized into three genres including: translation-based (Bordes et al., 2013), factorization-based (Yang et al., 2015; Trouillon et al., 2016) and neural network-based (Dettmers et al., 2018; Socher et al., 2013) methods. These methods for KGE mostly vectorized the entities and facts, and optimized the representations under a specific criterion. However, the above methods could suffer from the generalization issue with the increase of heterogeneity.

**Causal Inference.** Causal inference (Pearl et al., 2016) is a powerful tool that aims to analyze the causality behind the data. Investigating pure causality can help to better understand both data generation and model inference. Causal inference has proved promising in various communities such as computer vision (Zhang et al., 2020; Yue et al., 2020), natural language processing (Feder et al., 2021), and recommender system (Yang et al., 2021; Zhang et al., 2021). Particularly, a body of research has paid attention to graphs through the lens of causality. Feng et al. (2021) considered the discrepancy of local structure in GNN and estimated the causal effect of a node's local structure for the prediction. Sui et al. (2022) leveraged backdoor adjustment to discover the causal graph structure for graph classification. Lin et al. (2022) proposed a framework to generate post-hoc causal explanations for GNN based on latent causal factors by finding which part of the whole graph causes the final prediction. In this work, we have studied the multi-relational graph neural networks from the perspective of causality toward generalizable representation learning.

## 7 Conclusion

In this paper, under the paradigm of graph convolution, we investigate the multi-relational message passing process from the perspective of causality. Then, we propose a message intervention method for generalizable multi-relational graph representation learning, named MILER, to remedy the generalization issue that exists in multi-relational graphs due to heterogeneity. We first use a composition-based encoder to embed the vertices and relations with relational and directional awareness, and then hire a message diverter to split the relational message into the causal and trivial message flows. Afterward, we achieve the message intervention guided by the backdoor adjustment rule. Through extensive experiments on several knowledge graph benchmarks toward different tasks, we validate both the effectiveness and the generalization ability of our proposed method.

## 8 REPRODUCIBILITY STATEMENT

In this work, we adopted five public datasets, where the description can be found in Appendix B.1. Particularly, in the generalizability experiments, we used several variants of FB15k-237 and WN18RR datasets by adjusting the proportion of training edges. The modification details can be found in Appendix B.3. Besides, we implemented MILER by Deep Graph Libary (DGL). The implementation details are listed in Appendix B.2. Following ICLR policies, we will anonymously release the code to the reviewers and ACs during the discussion stage. For the baselines, the code that we use to reproduce and the corresponding hyperparameters are also introduced in Appendix B.2.

## REFERENCES

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pp. 2787–2795, 2013.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Rémi Cadène, Corentin Dancette, Hedi Ben-younes, Matthieu Cord, and Devi Parikh. Rubi: Reducing unimodal biases for visual question answering. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 839–850, 2019.

Shiyu Chang, Yang Zhang, Mo Yu, and Tommi S. Jaakkola. Invariant rationalization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1448–1458. PMLR, 2020.

Guanzheng Chen, Jinyuan Fang, Zaiqiao Meng, Qiang Zhang, and Shangsong Liang. Multi-relational graph representation learning with bayesian gaussian process network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 5530–5538, 2022.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 1811–1818. AAAI Press, 2018.

Amir Feder, Katherine A Keith, Emaad Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E Roberts, et al. Causal inference in natural language processing: Estimation, prediction, interpretation and beyond. *arXiv preprint arXiv:2109.00725*, 2021.

Fuli Feng, Weiran Huang, Xiangnan He, Xin Xin, Qifan Wang, and Tat-Seng Chua. Should graph convolution trust neighbors? a simple causal inference method. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1208–1218, 2021.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1263–1272. PMLR, 2017.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 1024–1034, 2017.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Wanyu Lin, Hao Lan, Hao Wang, and Baochun Li. Orphicx: A causality-inspired latent variable model for interpreting graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13729–13738, 2022.

Diego Marcheggiani and Ivan Titov. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1506–1515, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1159.

Heiko Paulheim and Johannes Fümkranz. Unsupervised generation of data mining features from linked open data. In *Proceedings of the 2nd international conference on web intelligence, mining and semantics*, pp. 1–12, 2012.

Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. *Causal Inference in Statistics: A Primer*. John Wiley & Sons, 2016.

Yunqi Qiu, Kun Zhang, Yuanzhuo Wang, Xiaolong Jin, Long Bai, Saiping Guan, and Xueqi Cheng. Hierarchical query graph generation for complex question answering over knowledge graph. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pp. 1285–1294. ACM, 2020. doi: 10.1145/3340531.3411888.

Petar Ristoski and Heiko Paulheim. Rdf2vec: Rdf graph embeddings for data mining. In *International Semantic Web Conference*, pp. 498–514. Springer, 2016.

Petar Ristoski, Gerben Klaas Dirk de Vries, and Heiko Paulheim. A collection of benchmark datasets for systematic evaluations of machine learning on the semantic web. In *International semantic web conference*, pp. 186–194. Springer, 2016.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pp. 593–607. Springer, 2018.

Chao Shang, Yun Tang, Jing Huang, Jinbo Bi, Xiaodong He, and Bowen Zhou. End-to-end structure-aware convolutional networks for knowledge base completion. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pp. 3060–3067. AAAI Press, 2019. doi: 10.1609/aaai.v33i01.33013060.

Xinyao Shen, Jiangjie Chen, Jiaze Chen, Chun Zeng, and Yanghua Xiao. Diversified query generation guided by knowledge graph. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pp. 897–907, 2022.

Richard Socher, Danqi Chen, Christopher D. Manning, and Andrew Y. Ng. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pp. 926–934, 2013.

Yongduo Sui, Xiang Wang, Jiancan Wu, Min Lin, Xiangnan He, and Tat-Seng Chua. Causal attention for interpretable and generalizable graph classification. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1696–1705, 2022.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. Rotate: Knowledge graph embedding by relational rotation in complex space. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Kristina Toutanova and Danqi Chen. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, pp. 57–66, Beijing, China, 2015. Association for Computational Linguistics. doi: 10.18653/v1/W15-4007.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pp. 2071–2080. JMLR.org, 2016.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha P. Talukdar. Composition-based multi-relational graph convolutional networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Jie Wang, Zhanqiu Zhang, Zhihao Shi, Jianyu Cai, Shuiwang Ji, and Feng Wu. Duality-induced regularizer for semantic matching knowledge graph embeddings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, Tianjun Xiao, Tong He, George Karypis, Jinyang Li, and Zheng Zhang. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315*, 2019.

Ying-Xin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. Discovering invariant rationales for graph neural networks. *arXiv preprint arXiv:2201.12872*, 2022.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Mengyue Yang, Quanyu Dai, Zhenhua Dong, Xu Chen, Xiuqiang He, and Jun Wang. Top-n recommendation with counterfactual user preference simulation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 2342–2351, 2021.

Rui Ye, Xin Li, Yujie Fang, Hongyu Zang, and Mingzhong Wang. A vectorized relational graph convolutional network for multi-relational network alignment. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pp. 4135–4141. ijcai.org, 2019. doi: 10.24963/ijcai.2019/574.

Zhongqi Yue, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Interventional few-shot learning. *Advances in neural information processing systems*, 33:2734–2746, 2020.

Dong Zhang, Hanwang Zhang, Jinhui Tang, Xian-Sheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. *Advances in Neural Information Processing Systems*, 33:655–666, 2020.

Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. Causal intervention for leveraging popularity bias in recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 11–20, 2021.

# A PROOFS

## A.1 DERIVATION OF BACKDOOR ADJUSTMENT FOR MULTI-RELATION MESSAGE PASSING

According to the *do-calculus* (Pearl et al., 2016), when we try to intervene on a variable in a causal graph, we remove all the edges that point to the variable and obtain the modified causal graph. Similarly, when we are conducting the operation $do(\{C_i\})$ in Figure 1, the original SCM can be modified to Figure 7, where all edges point to $C_i$ are cut off.

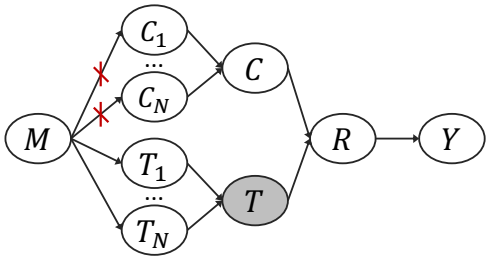



Figure 7: The modified version of the SCM in Figure 1 (in the main text).

Considering the two versions of SCM, the interventional distribution $P(Y|do(\{C_i\}))$ in Equation (1) can be derived by:

$$P(Y|do(\{C_i\})) = \sum_t P(Y|do(\{C_i\}), T=t)P(T=t|do(\{C_i\})) \tag{A.1}$$

$$= \sum_t P(Y|do(\{C_i\}), T=t)P(T=t) \tag{A.2}$$

$$= \sum_t P(Y|\{C_i\}, T=t)P(T=t), \quad (i=1,2,\ldots,|\widetilde{\mathcal{R}}|), \tag{A.3}$$

where Equation (A.1) is derived by Bayesian Rule, Equation (A.2) is because $T$ is independent with $\{C_i\}$ after the removal of the edges and the removal will not affect the prior of $T$, and Equation (A.3) is because causal effect from $C_i$ and $T$ to $Y$ will not change no matter the existence of the removed edges.

Moreover, we also noticed that given the conditional distribution $P(Y|\{C_i\})$:

$$\begin{aligned} P(Y|\{C_i\}) &= \sum_t P(Y, T=t|\{C_i\}) \\ &= \sum_t P(Y|\{C_i\}, T=t)P(T=t|\{C_i\}) \\ &= \sum_t P(Y|\{C_i\}, T=t)P(T=t)\cdot\frac{P(\{C_i\}|T=t)}{P(\{C_i\})}, \end{aligned} \tag{A.4}$$

the interventional distribution $P(Y|do(\{C_i\}))$ is not theoretically equal to $P(Y|\{C_i\})$ because of the extra term.

## A.2 DERIVATION OF THE OPTIMIZATION OBJECTIVE

As shown in Section 3.2, we intend to estimate the interventional distribution $P(Y|do(\{C_i\}))$. According to Maximum Likelihood Estimation (MLE), the optimization objective can be formulated as follows:

$$\max_{\Theta} \ \mathbb{E}_{(m,y)}\Big[\log P_\Theta\left(Y|do\left(\{C_i\}\right)\right)\Big], \quad i=1,2,\ldots,|\widetilde{\mathcal{R}}|, \tag{A.5}$$

where $m$ denotes the relational message that the node is about to receive from neighbors and $y$ is its corresponding label. In MILER, $m$ can be divided into the causal message and trivial message by the proposed message diverter, and thus can be sampled for estimating the interventional distribution.

| | Link Prediction | | Node Classification | | |
|---|---|---|---|---|---|
| | **FB15k-237** | **WN18RR** | **AIFB** | **MUTAG** | **BGS** |
| Vertices | 14,541 | 40,943 | 8,285 | 23,644 | 333,845 |
| Edges | 310,116 | 93,003 | 29,043 | 74,227 | 916,199 |
| Relations | 237 | 11 | 45 | 23 | 103 |
| Labeled | - | - | 176 | 340 | 146 |
| Classes | - | - | 4 | 2 | 2 |

Table 3: Statistics of the datasets used in our work.

Please refer to Section 4.1 for more details. Then we let $\mathcal{O} = \mathbb{E}_{(m,y)}\Big[\log P_\Theta\left(Y|do\left(\{C_i\}\right)\right)\Big]$, and we have:

$$\begin{aligned}\mathcal{O} &= \mathbb{E}_{(m,y)}\Big[\log P_\Theta\left(Y|do\left(\{C_i\}\right)\right)\Big] \\ &= \mathbb{E}_{(m,y)}\Big[\log \sum_t \left(P_\Theta(Y|\{C_i\}, T=t)P(T=t)\right)\Big] \\ &= \mathbb{E}_{(m,y)}\Big[\log \mathbb{E}_t\left[P_\Theta(Y|\{C_i\}, T=t)\right]\Big] \\ &\geq \mathbb{E}_{(m,y),t}\Big[\log P_\Theta(Y|\{C_i\}, T=t)\Big].\end{aligned} \tag{A.6}$$

Thus:

$$\max_\Theta\ \mathcal{O} \Longleftrightarrow \max_\Theta\ \mathbb{E}_{(m,y),t}\Big[\log P_\Theta(Y|\{C_i\}, T=t)\Big], \quad i = 1, 2, \ldots, |\widetilde{\mathcal{R}}|. \tag{A.7}$$

# B More Experimental Details

## B.1 Data Description

In this work, we use the following two datasets for link prediction:

- **FB15k-237** (Toutanova & Chen, 2015) is the subset of the knowledge graph dataset FB15k (Bordes et al., 2013). FB15k-237 removes the reverse triples from FB15k to avoid unreasonable inference on these triples.
- **WN18RR** (Dettmers et al., 2018) is the subset of WN18 (Bordes et al., 2013). A similar modification is conducted to cure the flaws of the complete set.

As for the node classification task, we use three RDF-format datasets (Ristoski et al., 2016) as follows:

- **AIFB** describes the AIFB research institute about the staff, research groups and publications. The goal is to predict the affiliation of people in the dataset.
- **MUTAG** is a dataset that was originally published for the DL-Learner toolkit[1]. It gives information on hundreds of complex molecules that are potentially carcinogenic. The goal is to classify whether the molecule is mutagenic or not.
- **BGS** describes the geological measurements in Great Britain. The goal is to predict the lithogenesis property of named rock units.

The statistics of the above datasets are given in Table 3.

## B.2 Implementation Details

We implemented MILER for both link prediction and node classification tasks using Deep Graph Library[2] (DGL) 0.8.1 (Wang et al., 2019). We trained the model on Ubuntu 16.04.7 LTS Linux

[1] `http://www.dl-learner.org`

[2] `https://www.dgl.ai`

| Hyperparameter | Link Prediction | Node Classification |
|---|---|---|
| # layers | {1, 2} | {1, 2, 3} |
| learning rate | {0.01, 0.005, 0.001} | {0.01, 0.005, 0.001} |
| batch size | {256, 512, 1024} | - |
| dropout | {0, 0.1, 0.3} | {0, 0.1, 0.3, 0.5, 0.7} |
| intervention sample size | {10, 20, 40, 50} | {20, 50, 100, 200} |
| L2 regularization | 0 | {0, 0.01, 0.0001} |
| layer size | 200 | {32, 64} |
| regularization for optimizing Eq.(9) & constraint in Eq.(7) | {0.001, 0.01, 0.1, 1} | {0.01, 0.1, 1} |

Table 4: Hyperparameter candidates on two tasks.

Machine with 160 cores, 1510G of RAM, and NVIDIA A100 GPU with 40GB of GPU memory. For both tasks, we took circular correlation as the composition operator. We used randomly initialized embeddings as the input features of both vertices and relations. We split all the standard datasets into train, validation, and test sets following (Schlichtkrull et al., 2018). We utilized Adam optimizer (Kingma & Ba, 2015) for optimization, and reported the average results over 5 runs using different initial parameters. Specifically, in the link prediction, we set the input size as 100, and used ConvE (Dettmers et al., 2018) as the scorer. Then, we performed a hyperparameter search on the validation set following Table 4. In the node classification, we set the input size as 32, and used a single-layer MLP as the classifier. A similar hyperparameter search was conducted based on Table 4.

Regarding the baselines, we reproduced RotatE[3], SACN[4], and CompGCN[5] in the link prediction, and RGCN[6] and CompGCN in the node classification by ourselves, while using the results originally reported for the remainder. For RotatE and SACN, we followed the officially recommended hyperparameters, while for RGCN and CompGCN, we tuned the shared hyperparameters according to Table 4.

### B.3 Dataset Modification for Generalizability Experiments

In the experiments, we used the variants of the standard FB15k-237 and WN18RR datasets with different proportions of training edges to validate the generalization ability. Here we introduce how we construct these variant datasets. Specifically, given a training proportion, we first calculate how many edges need to be removed from the training set compared with the standard split. Then we take out these edges from the rear of the standard training set, orderly divide them in half, and dispatch the two counterparts into the validation and test set, respectively. Therefore, the modification is completely deterministic and reproducible.

[3] https://github.com/DeepGraphLearning/KnowledgeGraphEmbedding

[4] https://github.com/JD-AI-Research-Silicon-Valley/SACN

[5] https://github.com/dmlc/dgl/tree/master/examples/pytorch/compGCN

[6] https://github.com/dmlc/dgl/tree/master/examples/pytorch/rgcn-hetero