Propagation on Multi-relational Graphs for Node Regression

Eda Bayram
École Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland

Abstract

Recent years have witnessed a rise in real-world data captured with rich structural information that can be conveniently depicted by multi-relational graphs. While the inference of continuous node features across a simple graph is rather under-studied by the current relational learning research, we go one step further and focus on the node regression problem on multi-relational graphs. We take inspiration from the well-known label propagation algorithm aiming at completing categorical features across a simple, weighted graph and propose a novel propagation framework for completing missing continuous features at the nodes of a multi-relational and directed graph. Our multi-relational propagation algorithm is composed of iterative neighborhood aggregations which originate from a relational local generative model. Our findings show the benefit of exploiting the multi-relational structure of the data in node regression task.

Introduction

Various disciplines are now able to capture different level of interactions between the entities of their interest, which promotes multiple types of relationships within data. Examples include social networks (Cozzo et al. 2016; Wasserman and Faust 1994), biological networks (Bentley et al. 2016; De 2017), transportation networks (Boccaletti et al. 2014; Aleta and Moreno 2019) and etc. Multi-relational graphs are convenient for representing such complex network-structured data. Recent years have witnessed a strong line of relational learning studies focusing on the inference of the node-level and graph-level categorical features (Chami et al. 2020). Most of these are working on simple graphs and there has been little interest in regression of continuous node features across the graph. In particular, node regression on multi-relational graphs still remains unexplored.

In this study, we present a multi-relational node regression framework. In particular, we address the following problem. Given the multi-relational structure of the data and partially observed continuous features belonging to the data entities, we aim at completing the missing features. It is possible to encode the intrinsic structure of the data by a graph accommodating multiple types of directed edges where each data entity is represented by a node. Accordingly, we establish the main research question we address as *How can we achieve node-value imputation on a multi-relational and directed graph?*.

For this purpose, we propose an algorithm which propagates observed set of node features towards the missing ones across a multi-relational and directed graph. We take inspiration from the well-known label propagation algorithm (Zhu and Ghahramani 2002) aiming at completing categorical features across a simple, weighted graph. We see that simple neighborhood aggregations operated on a given relational structure hold the basis for many iterative graph algorithms including the label propagation. Thus, we first break down the propagation framework by the neighborhood aggregations derived through a simple local generative model. Later, we extend this by incorporating a multi-relational neighborhood and suggest a relational local generative model. Then, we build our algorithm, which we call multi-relational propagation (MRP), by iterative neighborhood aggregation steps originating from this new model. We provide the derivation of the parameters of relational local generative model, which can be estimated over the observed set of node features and assigned as the parameters of the proposed propagation algorithm, MRP. Our method can be considered as a sophisticated version of the standard propagation algorithm by enabling regression of continuous node features over a multi-relational and directed graph. We compare our multi-relational propagation method against the standard propagation in several node regression scenarios. In each case, our approach enhances the results considerably by integrating the multi-relational structure of the data into the regression framework.

Related Work. Node regression problem has been studied on simple graphs for signal inpainting (Chen et al. 2014; Perraudin and Vandergheynst 2017) and node representation learning (Opolka et al. 2019; Wu et al. 2020; Deng et al. 2021; Ivanov and Prokhorenkova 2021). Many of these approaches implicitly employ a smoothness prior which promotes similar representations at the neighboring nodes of the graph (Zhou et al. 2004). The smoothness prior exploited in node representation learning studies broadly prescribes minimizing the Euclidean distance between features at the

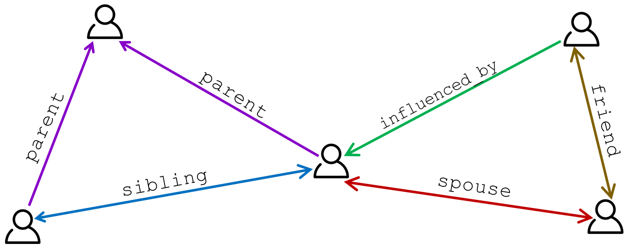

Figure 1: A fragment of a multi-relational and directed social network

connected nodes. Throughout the paper, we refer to such prior as ℓ_2 sense smoothness. Despite its practicality, ℓ_2 sense smoothness prior suffers from several major limitations which might mislead regression on a multi-relational and directed graph. First, it treats all neighbors of a node equally during the reasoning about the node's state, although neighbors connected via different types of relations might play a different role in the inference task. For instance, Fig. 1 illustrates multiple types of relationships that might arise between people. Here, each relation type presumably relies on different affinity rules or different levels of importance depending on the node regression task. It is also worth to notice that some relation types are inherently symmetric such as `sibling`, whereas some others are asymmetric such as `parent`. This is also indicated by the direction of the graph edges, see Figure 1. Euclidean distance minimization broadly assumes the values at the neighboring nodes are as close as possible, which may not always be the case. Therefore, the inference approach applied on simple graphs is insufficient for handling the asymmetry emerging from the directed relationships. We thus depart from the straightforward ℓ_2 sense of smoothness and augment the prior with a relational local generative model.

Contributions. In this study, (i) we provide the breakdown of propagation algorithm on simple graphs from Bayesian perspective, (ii) we introduce a relational local generative model, which permits neighborhood aggregation operation on a multi-relational, directed neighborhood, (iii) we provide a comparison of local operations applied on simple graphs and multi-relational graphs, (iv) we propose a novel propagation framework MRP, which properly handles propagating observed continuous node features across a multi-relational directed graph and complete missing ones.

Propagation on Simple Graphs

We denote a simple, undirected graph by $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with set of nodes $\mathcal{V}$ and set of edges $\mathcal{E}$. Also, let us denote $x_i \in \mathbb{R}$ as the continuous node feature[1] hold by node-i.

Local generative model. We recall the smoothness prior prescribing the neighboring node representations to be as close as possible in terms of ℓ_2-norm. Consequently, we write a simple local generative model which relates two neighboring nodes as follows:

$$x_i = x_j + \epsilon, \tag{1}$$

where $(i, j) \in \mathcal{E}$ and $\epsilon \sim \mathcal{N}(0, \sigma_{ij}^2)$.

First order Bayesian estimate of node's value. The local generative model can be used to obtain an approximation of the node's state in terms of its local neighborhood. This can be achieved by maximizing the expectation of the node's feature given that of its 1st-hop neighbors.

$$\underset{x_i}{\operatorname{argmax}} \quad \mathrm{p}(x_i | \{x_j : (i, j) \in \mathcal{E}\}) \tag{2}$$

Applying Bayes' rule, we obtain

$$\underset{x_i}{\operatorname{argmax}} \quad \frac{\mathrm{p}(\{x_j : (i, j) \in \mathcal{E}\} | x_i)\mathrm{p}(x_i)}{\mathrm{p}(\{x_j : (i, j) \in \mathcal{E}\})}. \tag{3}$$

Here, we make two assumptions. First, we assume that the prior distribution on the node features, $\mathrm{p}(x_i)$ for $i \in \mathcal{V}$, is uniform. Second, we only consider the partial correlations between the central node—whose state is to be estimated—and its 1st-hop neighbors while we neglect any partial correlation among the neighbor set—conditionally independence assumption. Accordingly, we reformulate the problem as

$$\underset{x_i}{\operatorname{argmax}} \prod_{(i,j)\in\mathcal{E}} \mathrm{p}(x_j | x_i) \quad = \underset{x_i}{\operatorname{argmin}} - \sum_{(i,j)\in\mathcal{E}} \log(\mathrm{p}(x_j | x_i)), \tag{4}$$

and rewrite it as minimizing the negative log-likelihood. Next, we plug in the local generative model (1):

$$\underset{x_i}{\operatorname{argmin}} \quad \sum_{(i,j)\in\mathcal{E}} \frac{\|x_j - x_i\|_2^2}{\sigma_{ij}^2}. \tag{5}$$

Neighborhood aggregation. The first order Bayesian estimate boils down to minimizing the Euclidean distance between node's feature to that of its neighbors, *i.e.*, suggesting a least squares problem in (5). Then, a first order Bayesian estimate is simply found by setting the gradient of the objective to zero:

$$\hat{x}_i = \frac{\sum_{(i,j)\in\mathcal{E}} \omega_{ij} x_j}{\sum_{(i,j)\in\mathcal{E}} \omega_{ij}}, \tag{6}$$

where $\omega_{ij} = 1/\sigma_{ij}^2$. Here, we draw attention to the fact that the obtained estimate is a linear combination of the neighbors' features.

The first order Bayesian estimate (6) fits the neighborhood aggregation operation accomplished in one iteration of a propagation algorithm (Zhu and Ghahramani 2002). This implies that propagating the estimated node states across the whole graph in an iterative manner, we expand the scope of the approximation until reaching a convergence. Therefore, we summarize the pipeline for developing a propagation algorithm as given in Fig. 2.

[1]Generalization to vectorial node representations is possible in principle, yet omitted here for the sake of simplicity.

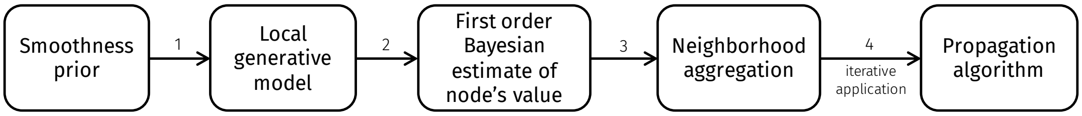

Figure 2: Overview of the pipeline for the development of a propagation algorithm

Multi-relational Model

We now introduce a multi-relational and directed graph as $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{P}$ is the set of relation types, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{P} \times \mathcal{V}$ is the set of multi-relational edges. The function $\mathtt{r}(i, j)$ returns the relation type $\mathtt{p} \in \mathcal{P}$ that is pointed from node j to node i. If such a relation exists between them, yet pointed from the node i to the node j, then the function returns the reverse as $\mathtt{p}^{-1}$.

Relational local generative model. It is required to diversify the simple local generative model in (1) by the set of relationships existing on a multi-relational graph. To this end, we propose the following generative model for the node's state given its multi-relational and directed neighbors:

$$x_i = \begin{cases} \eta_{\mathtt{p}} x_j + \tau_{\mathtt{p}} + \epsilon, & \forall \mathtt{r}(i,j) = \mathtt{p} \text{ where } \epsilon \sim \mathcal{N}(0, \sigma_{\mathtt{p}}^2) \\ \dfrac{x_j}{\eta_{\mathtt{p}}} - \dfrac{\tau_{\mathtt{p}}}{\eta_{\mathtt{p}}} + \epsilon, & \forall \mathtt{r}(i,j) = \mathtt{p}^{-1} \text{ where } \epsilon \sim \mathcal{N}(0, \dfrac{\sigma_{\mathtt{p}}^2}{\eta_{\mathtt{p}}^2}), \end{cases} \tag{7}$$

Equation (7) builds a linear relationship between the neighboring nodes by introducing relation-dependent scaling parameter η and a shift parameter τ. The latter case in (7) indicates the generative model yielded by the reverse relation, where the direction of the edge is reversed with respect to the former, thus, it is simply the reverse of the equation in the former case. Such a linear model conforms both symmetric and asymmetric relationships. This is because it can capture any bias over a certain relation through parameter τ or any change in scale through parameter η. We note that the default set for these parameters are suggested as $\tau = 0, \eta = 1$, which boils down to the simple local generative model (1).

First-order Relational Bayesian Estimate. We now estimate the node's state by its first-hop neighbors connected via multiple types relationships. We repeat the same assumptions, which casts the problem as maximizing the likelihood of node's 1st hop neighbors. To begin with, one can express the likelihood of a relational neighbor as follows:

$$\mathrm{p}(x_j|x_i) = \begin{cases} \sqrt{\frac{\omega_{\mathtt{p}}}{2\pi}} \exp\Big(-\dfrac{\omega_{\mathtt{p}}}{2}\Big(x_i - \eta_{\mathtt{p}} x_j - \tau_{\mathtt{p}}\Big)^2\Big), & \forall \mathtt{r}(i,j) = \mathtt{p} \\ \sqrt{\frac{\omega_{\mathtt{p}}\eta_{\mathtt{p}}^2}{2\pi}} \exp\Big(-\dfrac{\omega_{\mathtt{p}}\eta_{\mathtt{p}}^2}{2}\Big(x_i - \dfrac{x_j}{\eta_{\mathtt{p}}} + \dfrac{\tau_{\mathtt{p}}}{\eta_{\mathtt{p}}}\Big)^2\Big), & \forall \mathtt{r}(i,j) = \mathtt{p}^{-1}, \end{cases} \tag{8}$$

where we apply a change of parameter $\omega_p = 1/\sigma_{\mathtt{p}}^2$. Next, the estimation can be found by minimizing the negative log-likelihood as in (4). Once, the likelihoods (8) are substituted, we obtain the following objective:

$$\underset{x_i}{\operatorname{argmin}} \sum_{\mathtt{p} \in \mathcal{P}} \Bigg(\sum_{\mathtt{r}(i,j)=\mathtt{p}} \frac{\omega_{\mathtt{p}}}{2}\Big(x_i - \eta_{\mathtt{p}} x_j - \tau_{\mathtt{p}}\Big)^2 + \sum_{\mathtt{r}(i,j)=\mathtt{p}^{-1}} \frac{\omega_{\mathtt{p}}\eta_{\mathtt{p}}^2}{2}\Big(x_i - \frac{x_j}{\eta_{\mathtt{p}}} + \frac{\tau_{\mathtt{p}}}{\eta_{\mathtt{p}}}\Big)^2 \Bigg). \tag{9}$$

Relational Neighborhood Aggregation. For an arbitrary node $i \in \mathcal{V}$, we denote the loss to be minimized as $\mathcal{L}_i$. Such a loss leads to a least squares problem whose solution satisfies $\dfrac{\partial \mathcal{L}_i}{\partial x_i}(\hat{x}_i) = 0$. Accordingly, the estimate can be found as

$$\hat{x}_i = \frac{\sum_{\mathtt{p}\in\mathcal{P}} \Big(\sum_{\mathtt{r}(i,j)=\mathtt{p}} \omega_{\mathtt{p}}\Big(\eta_{\mathtt{p}} x_j + \tau_{\mathtt{p}}\Big) + \sum_{\mathtt{r}(i,j)=\mathtt{p}^{-1}} \omega_{\mathtt{p}}\eta_{\mathtt{p}}^2\Big(\frac{x_j}{\eta_{\mathtt{p}}} - \frac{\tau_{\mathtt{p}}}{\eta_{\mathtt{p}}}\Big)\Big)}{\sum_{\mathtt{p}\in\mathcal{P}} \Big(\sum_{\mathtt{r}(i,j)=\mathtt{p}} \omega_{\mathtt{p}} + \sum_{\mathtt{r}(i,j)=\mathtt{p}^{-1}} \omega_{\mathtt{p}}\eta_{\mathtt{p}}^2\Big)}. \tag{10}$$

Estimation of Relational Parameters

The parameters of the local generative model associated with relation type $\mathtt{p} \in \mathcal{P}$ are introduced as $\{\tau_{\mathtt{p}}, \eta_{\mathtt{p}}\, \omega_{\mathtt{p}}\}$. These parameters can be estimated over the set of node pairs connected to each other by relation $\mathtt{p}$, *i.e.*, $\big\{(x_i, x_j)\ \forall i, j \in \mathcal{V} | \mathtt{r}(i,j) = \mathtt{p}\big\}$. For this purpose, we carry out the maximum likelihood estimation over the parameters:

$$\underset{\tau_{\mathtt{p}}, \eta_{\mathtt{p}}\, \omega_{\mathtt{p}}}{\operatorname{argmax}} \quad \mathrm{p}\Big(\big\{(x_i, x_j)\ \forall i, j \in \mathcal{V} \,|\, \mathtt{r}(i,j) = \mathtt{p}\big\} \;\big|\; \tau_{\mathtt{p}}, \eta_{\mathtt{p}}\, \omega_{\mathtt{p}}\Big) \tag{11}$$

Then, we conduct an approximation over the node pairs that are connected by a given relation type while neglecting any conditional dependency that might exist among these node pairs. Hence, we can write the likelihood on each node pair in a product as follows:

$$\underset{\tau_{\mathtt{p}}, \eta_{\mathtt{p}}\, \omega_{\mathtt{p}}}{\operatorname{argmax}} \prod_{\mathtt{r}(i,j)=\mathtt{p}} \mathrm{p}\Big((x_i, x_j) \;\big|\; \tau_{\mathtt{p}}, \eta_{\mathtt{p}}\, \omega_{\mathtt{p}}\Big), \tag{12}$$

while we express the likelihood of a pair of values (x_i, x_j) belonging to the nodes connected by relation type p using the model (7) as

$$\mathrm{p}\Big((x_i, x_j) \,|\, \mathtt{r}(i,j) = \mathtt{p} \,\big|\, \tau_{\mathtt{p}}, \eta_{\mathtt{p}}\, \omega_{\mathtt{p}}\Big) = \sqrt{\tfrac{\omega_{\mathtt{p}}}{2\pi}} \exp\Big(-\frac{\omega_{\mathtt{p}}}{2}\Big(x_i - \eta_{\mathtt{p}} x_j - \tau_{\mathtt{p}}\Big)^2\Big). \tag{13}$$

Accordingly, we proceed with the minimization of negative log-likelihood to solve the problem in (12). The reader might recognize that the solution of this problem is equivalent to the parameters of a linear regression model (Rencher and Christensen 2012). This is simply because we introduce linear generative models (7) for the relationships existing on the graph. Therefore, the parameters of the generative model can be found as follows:

$$\eta_{\mathtt{p}} = \frac{\sum_{\mathtt{r}(i,j)=\mathtt{p}}(x_i - \mu)(x_j - \mu)}{\sum_{\mathtt{r}(i,j)=\mathtt{p}}(x_j - \mu)^2}, \tag{14}$$

where $\mu = \mathrm{mean}(\mathbf{x})$ is the mean of the node values. Then,

$$\tau_{\mathtt{p}} = \mathrm{mean}\Big(\big\{(x_i - \eta_{\mathtt{p}} x_j)\, \forall i,j \in \mathcal{V} \mid \mathtt{r}(i,j) = \mathtt{p}\big\}\Big), \tag{15}$$

$$\omega_{\mathtt{p}} = 1/\mathrm{mean}\Big(\big\{(x_i - \eta_{\mathtt{p}} x_j - \tau_{\mathtt{p}})^2\, \forall i,j \in \mathcal{V} \mid \mathtt{r}(i,j) = \mathtt{p}\big\}\Big). \tag{16}$$

Local Generative Model and Local Operation. We now recap the proposed multi-relational regression approach in comparison to regression on simple graphs. For this purpose, we summarize the local generative model, the loss associated with the estimation and the corresponding first order estimate for both cases in Table 1. In the table, we frame the first order relational Bayesian estimate, which is expressed in (10), in a neighborhood aggregation. Unlike in the simple case, it is not directly a weighted average of the neighbors but the neighbors are subject to a transformation with respect to the type and the direction of their relation to the central node. The relational transformation is controlled by the parameters η and τ. For this reason, in Table 1 we use the following functions as shortcuts for the transformations applied on the neighbors in simple and multi-relational case:

$$\begin{aligned} f(x) &= x, \\ f_{\mathtt{p}}(x) &= \eta_{\mathtt{p}} x + \tau_{\mathtt{p}}. \end{aligned}$$

In addition, $\mathcal{P}^{-1} = \{\mathtt{p}^{-1}, \forall \mathtt{p} \in \mathcal{P}\}$ denotes the set relation types where the edge direction is reversed. For the reversed relationships, the set of parameters can be simply set as follows:

$$\eta_{\mathtt{p}^{-1}} = \frac{1}{\eta_{\mathtt{p}}}, \qquad \tau_{\mathtt{p}^{-1}} = -\frac{\tau_{\mathtt{p}}}{\eta_{\mathtt{p}}}, \qquad \omega_{\mathtt{p}^{-1}} = \eta_{\mathtt{p}}^2 \omega_{\mathtt{p}}. \tag{17}$$

Following the transformations, the estimation is computed by a weighted average of those, that is controlled by the parameter ω. It is worth to notice that this parameter is equivalent to the inverse of error variance of the relational local generative model (7). Therefore, the estimate can be interpreted as the outcome of an aggregation with precision that ranks the relational information.

Multi-relational Propagation Algorithm

In Fig. 2, the propagation algorithm is depicted as an iterative neighborhood aggregation method where each iteration computes the solution of a first order Bayesian estimation problem on the graph. In a similar manner, here, we propose a propagation algorithm that relies on the first order relational Bayesian estimate that is introduced in (9). The algorithm operates iteratively where the relational neighborhood aggregation (10) is accomplished at each node of the graph simultaneously. Thus, we denote a vector $\mathbf{x}^{(k)} \in \mathbb{R}^N$ composing the values at iteration-k over the set of nodes for $|\mathcal{V}| = N$. Next, we express the iterations in matrix-vector multiplication format.

Iterations in matrix notation. We first introduce matrix $\mathbf{A}_{\mathtt{p}}$ for encoding the adjacency pattern of relation type p. Therefore, it is $(N \times N)$ asymmetric matrix storing the incoming edges on its rows and outgoing edges on its columns. Accordingly, one can compile the aggregations (10) accomplished simultaneously over the entire graph using a matrix notation. Then, the relational local operations at iteration-k can be expressed as follows:

$$\begin{aligned} \mathbf{x}^{(k)} = \Bigg(\sum_{\mathtt{p} \in \mathcal{P}} \Big(& \omega_{\mathtt{p}} \big(\eta_{\mathtt{p}} \mathbf{A}_{\mathtt{p}} \mathbf{x}^{(k-1)} + \tau_{\mathtt{p}} \mathbf{A}_{\mathtt{p}} \mathbf{1} \big) \\ & + \omega_{\mathtt{p}} \eta_{\mathtt{p}} \big(\mathbf{A}_{\mathtt{p}}^{\top} \mathbf{x}^{(k-1)} - \tau_{\mathtt{p}} \mathbf{A}_{\mathtt{p}}^{\top} \mathbf{1} \big) \Big) \Bigg) \\ & \odot \left(\sum_{\mathtt{p} \in \mathcal{P}} \Big(\omega_{\mathtt{p}} \mathbf{A}_{\mathtt{p}} \mathbf{1} + \omega_{\mathtt{p}} \eta_{\mathtt{p}}^2 \mathbf{A}_{\mathtt{p}}^{\top} \mathbf{1} \Big) \right)^{-1}, \end{aligned} \tag{18}$$

where $\mathbf{1}$ is the vector of ones, $\odot$ stands for element-wise multiplication. In addition, the inversion on the latter sum term is applied element-wise. This part, in particular, arranges the denominator in Equation (10) in vector format. Thus, it can be seen as the normalization factor over the neighborhood aggregation. For the purpose of simplification, we re-write (18) as

$$\mathbf{x}^{(k)} = (\mathbf{T}\mathbf{x}^{(k-1)} + \mathbf{S}\mathbf{1}) \odot (\mathbf{H}\mathbf{1})^{-1}, \tag{19}$$

by introducing the auxiliary matrices

$$\mathbf{T} = \sum_{\mathtt{p} \in \mathcal{P}} \eta_{\mathtt{p}} \omega_{\mathtt{p}} (\mathbf{A}_{\mathtt{p}} + \mathbf{A}_{\mathtt{p}}^{\top}), \tag{20}$$

$$\mathbf{S} = \sum_{\mathtt{p} \in \mathcal{P}} \tau_{\mathtt{p}} \omega_{\mathtt{p}} (\mathbf{A}_{\mathtt{p}} - \eta_{\mathtt{p}} \mathbf{A}_{\mathtt{p}}^{\top}), \tag{21}$$

$$\mathbf{H} = \sum_{\mathtt{p} \in \mathcal{P}} \omega_{p} (\mathbf{A}_{\mathtt{p}} + \eta_{\mathtt{p}}^2 \mathbf{A}_{\mathtt{p}}^{\top}). \tag{22}$$

Algorithm. Given the iterations above, we can now formalize the proposed algorithm that we call as Multi-relational Propagation (MRP). As we described initially, MRP targets a node-level completion task where the multi-relational graph $\mathcal{G}$ is a priori given and the nodes are partially labeled at $\mathcal{U} \subseteq \mathcal{V}$. In order to properly handle the propagation of continuous values at the labeled set of nodes towards the unlabeled ones, we introduce an indicator vector $\mathbf{u} \in \mathbb{R}^N$, which encodes the labeled nodes. Thus, it is initialized as $\mathbf{u}_i^{(0)} = 1$, if $i \in \mathcal{U}$, else 0. Then, the vector $\mathbf{x}$ stores the node values throughout the iterations. It is initialized by the values over $\mathcal{U}$, and, it is zero-padded at the unlabeled nodes, i.e., $\mathbf{x}_i^{(0)} = 0$ if $i \in \mathcal{V} \setminus \mathcal{U}$.

Similar to the label propagation algorithm (Zhu and Ghahramani 2002), our algorithm fundamentally consists of aggregation and normalization steps. In order to encompass

Table 1: Local Generative Model and Operation in Simple and Multi-relational Graphs

	Local Generative Model	Loss	Local Operation
Simple Weighted Graph	$x_i = x_j + \epsilon$ $\forall (i,j) \in \mathcal{E}$ $\epsilon \sim \mathcal{N}(0, 1/\omega_{ij})$	$\sum_{(i,j)\in\mathcal{E}} \omega_{ij}(x_i - x_j)^2$	$\frac{\sum_{(i,j)\in\mathcal{E}} \omega_{ij} f(x_j)}{\sum_{(i,j)\in\mathcal{E}} \omega_{ij}}$
Multi-relational Directed Graph	$x_i = \eta_{\mathrm{p}} x_j + \tau_{\mathrm{p}} + \epsilon$ $\forall \mathrm{r}(i,j) = \mathrm{p}$ $\epsilon \sim \mathcal{N}(0, 1/\omega_{\mathrm{p}})$	$\sum_{\mathrm{p}\in\mathcal{P}\cup\mathcal{P}^{-1}} \sum_{\mathrm{r}(i,j)=\mathrm{p}} \omega_{\mathrm{p}}(x_i - \eta_{\mathrm{p}} x_j - \tau_{\mathrm{p}})^2$	$\frac{\sum_{\mathrm{p}\in\mathcal{P}\cup\mathcal{P}^{-1}} \sum_{\mathrm{r}(i,j)=\mathrm{p}} \omega_{\mathrm{p}} f_{\mathrm{p}}(x_j)}{\sum_{\mathrm{p}\in\mathcal{P}\cup\mathcal{P}^{-1}} \sum_{\mathrm{r}(i,j)=\mathrm{p}} \omega_{\mathrm{p}}}$

the multi-relational transformation procedure during the aggregation, we formulate an iteration of MRP by the steps of aggregation, shift and normalization respectively. In addition, similar to the Page-rank algorithm (Brin and Page 1998), we employ a damping factor $\xi \in [0,1]$ in order to update the node's state by combining its value from the previous iteration.

Algorithm 1: MRP

Input: $\mathcal{U}, \{x_i | i \in \mathcal{U}\}, \{\mathbf{A}_{\mathrm{p}}, \tau_{\mathrm{p}}, \eta_{\mathrm{p}}, \omega_{\mathrm{p}}\}_{\mathcal{P}}$
Output: $\{x_i | i \in \mathcal{V} \setminus \mathcal{U}\}$
Initialization: $\mathbf{u}^0, \mathbf{x}^0, \mathbf{T}, \mathbf{S}, \mathbf{H}$
for $k = 1, 2, \cdots$ **do**

Step 1. Aggregate: $\mathbf{z} = \mathbf{T}\mathbf{x}^{(k-1)}$
Step 2. Shift: $\mathbf{z} = \mathbf{z} + \mathbf{S}\mathbf{u}^{(k-1)}$
Step 3. Aggregate the normalization factors: $\mathbf{r} = \mathbf{H}\mathbf{u}^{(k-1)}$
Step 4. Normalize: $\mathbf{z} = \mathbf{z} \odot \mathbf{r}^{\dagger}$
//† is for element-wise pseudo-inverse
Step 5. Update values:

$$\mathbf{x}_i^{(k)} = \begin{cases} \mathbf{x}_i^{(k-1)}, & \text{if } \mathbf{r}_i = 0 \\ \mathbf{z}_i, & \text{if } \mathbf{r}_i > 0, \mathbf{u}_i^{(k-1)} = 0 \\ (1-\xi)\mathbf{x}_i^{(k-1)} + \xi \mathbf{z}_i, & \text{e.w.} \end{cases}$$

Step 6. Update propagated nodes:
$\mathbf{u}^{(k)} = \mathbf{u}^{(k-1)}, \quad \mathbf{u}_i^{(k)} = 1$ if $\mathbf{r}_i > 0$
Step 7. Clamp the known values:
$\mathbf{x}_i^{(k)} = x_i, \forall i \in \mathcal{U}$

break if $\mathrm{all}(\mathbf{u}^{(k)})$ & $\mathrm{all}(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)} < \varepsilon)$
$x_i = \mathbf{x}_i^{(k)}, \forall i \in \mathcal{X} \setminus \mathcal{U}$.

We provide a pseudocode for MRP in Algorithm 1. Here, we reserve that the propagation parameters for each relation type, $\{\tau_{\mathrm{p}}, \eta_{\mathrm{p}}, \omega_{\mathrm{p}}\}$ are estimated in advance over the known set of nodes $\mathcal{U}$, as described in Section . Then, we provide them to the algorithm as input together with the adjacency matrices encoding the multi-relational, directed graph. Steps 1-4 in MRP are essentially responsible for the multi-relational neighborhood aggregation. Then at Step-5, the nodes' states are updated based on the collected information from the neighbors. If valid information collected from neighbors and the node is labeled then we employ the damping ratio, ξ, to update the node's state. This adjusts the amount of trade-off between the neighborhood aggregation and the previous state of the node. We distinguish whether an arbitrary node is currently labeled or not by the indicator vector, $\mathbf{u}^{(k)}$, which keeps track of propagated nodes throughout the iterations. Hence, in Step 6, we update it as well. Finally, in Step 7, we clamp the values at the known set of nodes, which means we leave their states unchanged, simply because they store the governing information for completing the missing ones. The algorithm terminates when all the nodes are propagated and the difference between two consecutive iterations is under a certain threshold.

We finally note that setting $\tau_{\mathrm{p}} = 0,\ \eta_{\mathrm{p}} = 1,\ \omega_{\mathrm{p}} = 1\ \forall \mathrm{p} \in \mathcal{P}$ manually, MRP drops down to a standard label propagation algorithm (LP)[2] as if we operate on a simple graph regardless of the relation types and directions.

Experiments

We now present a proof of the proposed multi-relational propagation method for node regression task on two applications. First, we test MRP in estimating weather measurements on a multi-relational and directed graph that connects the weather stations. Second, we evaluate the performance in predicting people's date of birth, where people are connected to each other on a social network composing different types relationships.

In the experiments, the damping factor is set as $\xi = 0.5$, then the threshold for terminating the iterations is set as 0.001 of the range of given values. Then, as evaluation metrics, we use root mean square error (RMSE), mean absolute percentage error (MAPE) and a normalized RMSE (nRMSE) with respect to the range of groundtruth values. The evaluation metrics are calculated over the estimation error on the unlabeled set of nodes. In the experiments, η parameter in MRP is left as default by 1 since we do not empirically observe a scale change over the relation types given by the datasets we work on. Then, we realize the estimation of parameters τ and ω for the relation types based on the observed set of node values as described in Section .

[2]The label propagation algorithm (Zhu and Ghahramani 2002) was originally designed for completing categorical features across a simple, weighted graph. By leaving the parameters of MRP as default, we actually revise it to propagate continuous features and apply for the node regression task.

The proposed multi-relational propagation algorithm[3] is previously adapted for node attribute completion in knowledge graphs in (Bayram, García-Durán, and West 2021).

Multi-relational Estimation of Weather Measurements

We test our method on a meteorological dataset provided by MeteoSwiss, which compiles various types of weather measurements on 86 weather stations between years 1981-2010 [4]. In particular, we use yearly averages of weather measurements in our experiments.

Construction of multi-relational directed graph. To begin with, we prepare a multi-relational graph representation $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ of the weather stations, *i.e.*, $|\mathcal{V}| = 86$, where we relate them based on two types of relationships, *i.e.*, $|\mathcal{P}| = 2$. First, we connect weather stations based on geographical proximity. Thus, we insert an edge between a pair of stations if the Euclidean distance between their GPS coordinates is below a threshold, on which we acquire 372 edges. The geographical proximity leads to a symmetric (bi-directed) relationship. Second, we relate the weather stations based on the altitude proximity in a similar logic. However, this time we anticipate an asymmetric relationship where the direction of an edge indicates an altitude ascend between weather stations. For both of the relation types, we adjust the threshold for building connections such that there is not any disconnected node. Consequently, altitude relations end up with 1144 edges. In the experiments, we randomly sample labeled set of nodes, $\mathcal{U}$, from the entire node set, $\mathcal{V}$, with a ratio of 80%. Then, we repeat the experiment in this setting for 50 times in Monte Carlo fashion. The evaluation metrics are then averaged over the series of simulations.

Table 2: Temperature and Snowfall Prediction Performances

		RMSE	MAPE	nRMSE
Temperature	LP	1.120	0.155	0.050
	MRP	1.040	0.147	0.045
Snowfall	LP	194.49	0.405	0.112
	MRP	180.10	0.357	0.105

Predicting Temperature and Snowfall on Directed Altitude Graph. We first conduct experiments on a simple scenario where we target predicting temperature and snowfall measurements by MRP, which permits reasoning over the directed altitude relations. Hence, we compare the proposed method to the standard label propagation algorithm, LP, which overlooks asymmetric relational reasoning. In this regard, we aim at evaluating the importance of the directed transformation during the neighborhood aggregation that is mainly gained by the shift parameter, τ. In fact, this parameter directly corresponds to the mean of differences computed along the direction of the altitude edges—since $\eta = 1$. Then, the parameter ω is simply associated with the inverse of the variance of the differences. This can be visualized by fitted RBFs on the distribution of the measurement changes on the edges, which is shown in Fig. 3. Here, we see that the temperature differences in the ascend direction, *i.e.*, $\left\{(x_i - x_j) \,\forall \mathtt{r}(i,j) = \mathtt{altitude\_ascend}\right\}$, has a mean in the negative region. This can be interpreted as an expected decrease in temperature values along altitude ascend. On the contrary, the mean of snowfall differences along the ascend direction has a positive value, which signifies a increase in snowfall as altitude rises.

As seen in Table 2, even in the case of single relation type—altitude proximity, incorporating the directionality in the graph MRP manages to record an enhancement in predictions over the regression realized by the label propagation, LP.

Predicting Precipitation on Directed, Multi-relational Graph. We now test our method in a another scenario where we integrate both altitude and geographical proximity relations to predict precipitation measurements on the weather stations. The prediction performance is compared to the regression by LP, that is accomplished over the altitude relations and GPS relations separately. Since MRP handles both of the relation types and the direction of the edges simultaneously, it achieves a better performance than LP, as seen in Table 3.

Table 3: Precipitation Prediction Performances

	RMSE	MAPE	nRMSE
LP-altitude	381.86	0.261	0.174
LP-gps	374.38	0.242	0.168
MRP	347.98	0.238	0.157

Relationship	edges	mean	variance
award_nomination	454	0	320.23
friendship	221	0	155.82
influenced_by	528	-36.25	1019.77
sibling	83	0	45.16
parent	98	-32.90	62.90
spouse	262	0	87.60
dated	231	0	90.95
awards_won	183	0	257.45

Table 4: Statistics for each type of relation. Columns respectively: number of edges, mean and variance of the date of birth difference belonging to the associated relation type.

Predicting People's Date of Birth in a Social Network

We also conduct experiment on a small subset of a relational database called Freebase (Toutanova and Chen 2015). For this purpose, we work on a graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P})$ composing 830 people, *i.e.*, $|\mathcal{V}| = 830$, connected via 8 different types of relationship, *i.e.*, $|\mathcal{P}| = 8$. Table 4 summarizes the statistics for each of them. Here, the task is to predict people's date of birth while it is only known for a subset of people. A

[3]Source code is available at https://github.com/bayrameda/MrAP/

[4]https://github.com/bayrameda/MaskLearning/tree/master/MeteoSwiss

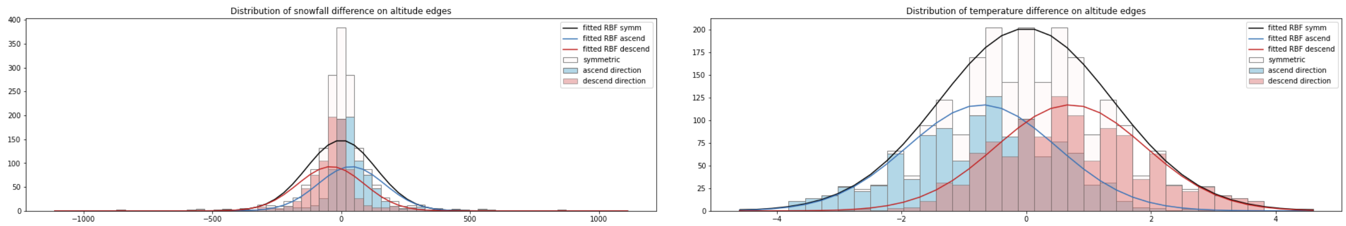

Figure 3: Distribution of change in temperature and snowfall(cm) measurements between the weather stations that are related via altitude proximity. Differences are shown along the ascend and descend direction separately, then, symmetric distribution shows the changes regardless of the direction. Also, a radial basis function (RBF) is fitted to each histogram.

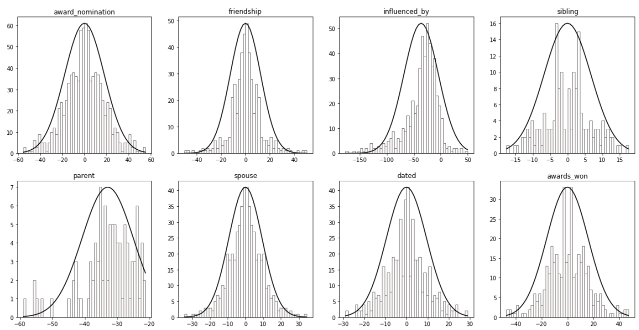

Figure 4: Distribution of difference (year) in date of births over different types of relations between people.

Table 5: Date of Birth Prediction Performances

		RMSE	**MAPE**	**nRMSE**
LP	`award_nomination`	32.43	0.011	0.115
	`friendship`	31.92	0.011	0.113
	`influenced_by`	30.29	0.012	0.108
	`sibling`	32.69	0.012	0.116
	`parent`	33.62	0.013	0.119
	`spouse`	31.45	0.011	0.112
	`dated`	31.70	0.011	0.113
	`awards_won`	33.04	0.012	0.117
	union	24.22	0.008	0.086
MRP		15.62	0.005	0.055

fragment of the multi-relational graph is also illustrated in Fig. 1, where it can be seen that there are basically two types of asymmetric relations: `influenced_by` and `parent`. Thus, the direction of the edges are specifically significant for those. Such asymmetry is also shown by visualizing the distribution of the difference in date of births, which is given over each type of relationship in Figure 4. We note that here we try to fit a radial basis function to the histogram of the differences since the residual term in the local generative model (7) is assumed to be normally distributed.

In the experiments, we randomly select the set of people whose date of birth is initially known, $\mathcal{U}$, with a ratio of 50% in $\mathcal{V}$. We again report the evaluation metrics that are averaged over a series of experiments repeated for 50 times. We compare the performance of MRP to the regression of date of birth values obtained with label propagation LP. We run LP over the edges of each relation type separately and also at the union of those. The results are given in Table 5. Based on the results, we can say that the most successful relation types for predicting the date of birth seems to be `influenced_by` and `spouse` using LP. Nonetheless, when LP operates on the union of the edges provided by different type of relationships, it performs better than any single type. Moreover, MRP is able to surpass this record by enabling a smart neighborhood aggregation over different types of relations. Once again, we argue that its success is due to the fact that it regards asymmetric relationships, here encountered as `influenced_by` and `parent`. In addition, it assigns different level of importance to the predictions collected through different type of relationships based on the uncertainty estimated over the observed data.

Conclusion

In this study, we proposed MRP, a sophisticated version of label propagation algorithm working on multi-relational and directed graphs for regression of continuous node features and we show its superior performance compared to standard propagation algorithm. Although we here target imputing continuous values at the nodes of a multi-relational and directed graph, it is possible to generalize the proposed approach for node embedding learning and then for the node classification tasks. The augmentation of the computational graph of the propagation algorithm using multiple types of directed relationships provided by the domain knowledge permits anisotropic operations on graph, which is claimed to be promising for future directions in graph representation learning (Dwivedi et al. 2020).

References

[Aleta and Moreno 2019] Aleta, A., and Moreno, Y. 2019. Multilayer networks in a nutshell. *Annual Review of Condensed Matter Physics* 10:45–62.

[Bayram, García-Durán, and West 2021] Bayram, E.; García-Durán, A.; and West, R. 2021. Node attribute completion in knowledge graphs with multi-relational propagation. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3590–3594. IEEE.

[Bentley et al. 2016] Bentley, B.; Branicky, R.; Barnes, C. L.; Chew, Y. L.; Yemini, E.; Bullmore, E. T.; Vértes, P. E.;

and Schafer, W. R. 2016. The multilayer connectome of caenorhabditis elegans. *PLoS computational biology* 12(12):e1005283.

[Boccaletti et al. 2014] Boccaletti, S.; Bianconi, G.; Criado, R.; Del Genio, C. I.; Gómez-Gardenes, J.; Romance, M.; Sendina-Nadal, I.; Wang, Z.; and Zanin, M. 2014. The structure and dynamics of multilayer networks. *Physics Reports* 544(1):1–122.

[Brin and Page 1998] Brin, S., and Page, L. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems* 30(1-7):107–117.

[Chami et al. 2020] Chami, I.; Abu-El-Haija, S.; Perozzi, B.; Ré, C.; and Murphy, K. 2020. Machine learning on graphs: A model and comprehensive taxonomy. *CoRR* abs/2005.03675.

[Chen et al. 2014] Chen, S.; Sandryhaila, A.; Lederman, G.; Wang, Z.; Moura, J. M.; Rizzo, P.; Bielak, J.; Garrett, J. H.; and Kovačevic, J. 2014. Signal inpainting on graphs via total variation minimization. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 8267–8271. IEEE.

[Cozzo et al. 2016] Cozzo, E.; de Arruda, G. F.; Rodrigues, F. A.; and Moreno, Y. 2016. Multilayer networks: metrics and spectral properties. In *Interconnected Networks*. Springer. 17–35.

[De 2017] De, M. D. 2017. Multilayer modeling and analysis of human brain networks. *GigaScience* 6(5):1–8.

[Deng et al. 2021] Deng, J.; Wan, S.; Wang, X.; Tu, E.; Huang, X.; Yang, J.; and Gong, C. 2021. Edge-aware graph attention network for ratio of edge-user estimation in mobile networks. In *2020 25th International Conference on Pattern Recognition (ICPR)*, 9988–9995. IEEE.

[Dwivedi et al. 2020] Dwivedi, V. P.; Joshi, C. K.; Laurent, T.; Bengio, Y.; and Bresson, X. 2020. Benchmarking graph neural networks. *arXiv preprint arXiv:2003.00982*.

[Ivanov and Prokhorenkova 2021] Ivanov, S., and Prokhorenkova, L. 2021. Boost then convolve: Gradient boosting meets graph neural networks. *arXiv preprint arXiv:2101.08543*.

[Opolka et al. 2019] Opolka, F. L.; Solomon, A.; Cangea, C.; Veličković, P.; Liò, P.; and Hjelm, R. D. 2019. Spatio-temporal deep graph infomax. *arXiv preprint arXiv:1904.06316*.

[Perraudin and Vandergheynst 2017] Perraudin, N., and Vandergheynst, P. 2017. Stationary signal processing on graphs. *IEEE Transactions on Signal Processing* 65(13):3462–3477.

[Rencher and Christensen 2012] Rencher, A. C., and Christensen, W. 2012. *Methods of Multivariate Analysis*. John Wiley & Sons. chapter 3, 47–90.

[Toutanova and Chen 2015] Toutanova, K., and Chen, D. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, 57–66.

[Wasserman and Faust 1994] Wasserman, S., and Faust, K. 1994. *Social network analysis: Methods and applications*, volume 8. Cambridge university press.

[Wu et al. 2020] Wu, Y.; Tang, Y.; Yang, X.; Zhang, W.; and Zhang, G. 2020. Graph convolutional regression networks for quantitative precipitation estimation. *IEEE Geoscience and Remote Sensing Letters*.

[Zhou et al. 2004] Zhou, D.; Bousquet, O.; Lal, T. N.; Weston, J.; and Schölkopf, B. 2004. Learning with local and global consistency. In *Advances in neural information processing systems*, 321–328.

[Zhu and Ghahramani 2002] Zhu, X., and Ghahramani, Z. 2002. Learning from labeled and unlabeled data with label propagation. *Tech. Rep., Technical Report CMU-CALD-02–107, Carnegie Mellon University*.