Propagation on Multi-relational Graphs for Node Regression

Eda Bayram(✉)

École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
eda.bayram@epfl.ch

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 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 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 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 several node regression scenarios in different settings.

Keywords: Multi-relational data · Label propagation · Node regression

1 Introduction

Various disciplines are now able to capture different level of interactions between entities of their interest, which promotes multiple types of relationships within data. Examples include social networks [8,17], biological networks [3,9], transportation networks [1,4], 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 node-level and graph-level categorical features [6]. Most of these are working on simple graphs and there has been little interest in the 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. Given multi-relational structure of data and partially observed continuous features belonging to the data entities, we aim at completing missing features. It is possible to encode intrinsic structure of the data by a graph accommodating multiple types of directed edges between graph's nodes that represent the data

R. M. Benito et al. (Eds.): COMPLEX NETWORKS 2021, SCI 1072, pp. 155–167, 2022.
https://doi.org/10.1007/978-3-030-93409-5_14

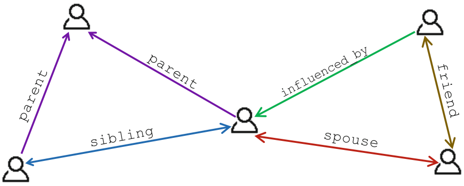

Fig. 1. A fragment of a multi-relational and directed social network

entities. Accordingly, we establish the main research question we address: *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 missing ones across a multi-relational and directed graph. We take inspiration from the well-known label propagation algorithm [20] 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 the proposed model, which can be estimated over the observed set of node features. 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. At each, our approach enhances the results considerably by integrating multi-relational structure of data into the regression framework.

Comparison to Existing Schemes. The node regression problem has been studied on simple graphs for signal inpainting [7,14] and node representation learning [10,12,13,18]. Many of these approaches implicitly employ a smoothness prior which promotes similar representations at the neighboring nodes of the graph [19]. The smoothness prior exploited in node representation learning studies broadly prescribes minimizing the Euclidean distance between features at the 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 that might mislead regression on a multi-relational and directed graph. First, it treats all neighbors of a node equally while reasoning about the node's state even though the neighbors connected via different types of relations might play different roles 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 a different affinity rule or a different level of importance depending on the node regression task. Second, some relation types are inherently symmetric, and some others are asymmetric[1]. Euclidean distance minimization broadly assumes that values at neighboring nodes are as close as possible, which may not always be the case for asymmetric 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 a breakdown of propagation algorithm on simple graphs from the Bayesian perspective, (ii) we introduce a relational local generative model, which permits neighborhood aggregation operation on a multi-relational, directed neighborhood, (iii) 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.

2 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, we denote $x_i \in \mathbb{R}$ as the continuous node feature[2] held 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 $x_i = x_j + \epsilon$ 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 first-hop neighbors:

$$\underset{x_i}{\operatorname{argmax}} \quad \mathrm{p}(x_i | \{x_j : (i,j) \in \mathcal{E}\}) = \underset{x_i}{\operatorname{argmax}} \quad \frac{\mathrm{p}(\{x_j : (i,j) \in \mathcal{E}\} | x_i) \mathrm{p}(x_i)}{\mathrm{p}(\{x_j : (i,j) \in \mathcal{E}\})}, \tag{1}$$

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

$$\underset{x_i}{\operatorname{argmax}} \prod_{(i,j) \in \mathcal{E}} \mathrm{p}(x_j | x_i) \quad = \underset{x_i}{\operatorname{argmin}} - \sum_{(i,j) \in \mathcal{E}} \log(\mathrm{p}(x_j | x_i), \tag{2}$$

[1] In directed graphs, symmetric relationships emerge from bi-directed edges where the edge direction is valid in both directions such as `sibling` whereas in asymmetric relationships, the edge direction is valid in only one direction such as `parent`, `child`. See the edge directions in Fig. 1.

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

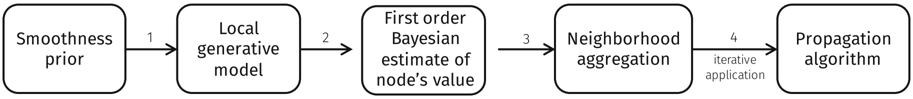

Fig. 2. Overview of the pipeline for the development of a propagation algorithm

and rewrite it as minimization of negative log-likelihood. Next, we plug in the local generative model and obtain the following problem:

$$\underset{x_i}{\operatorname{argmin}} \sum_{(i,j)\in\mathcal{E}} \frac{\|x_j - x_i\|_2^2}{\sigma_{ij}^2}. \tag{3}$$

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 (3). Its solution is simply found by setting the gradient of the objective to zero:

$$\hat{x}_i = \frac{\sum_{(i,j)\in\mathcal{E}} \omega_{ij} x_j}{\sum_{(i,j)\in\mathcal{E}} \omega_{ij}}, \tag{4}$$

where $\omega_{ij} = 1/\sigma_{ij}^2$. As seen, it is a linear combination of the neighbors' features. The first-order Bayesian estimation in the conditions considered above clarifies the neighborhood aggregation operation accomplished in one iteration of a propagation algorithm [20]. Estimating the node states across the whole graph iteratively, a propagation algorithm expands the scope of the approximation beyond the first-order until a stopping criterion is satisfied. Hence, we summarize the pipeline for developing a propagation algorithm as given in Fig. 2.

3 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, the function returns the reverse as $\mathtt{p}^{-1}$.

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

$$x_i = \begin{cases} \eta_{\mathtt{p}} x_j + \tau_{\mathtt{p}} + \epsilon, & \forall \mathtt{r}(i,j) = \mathtt{p} \text{ where } \epsilon \sim \mathcal{N}(0, \sigma_{\mathtt{p}}^2) \\ \dfrac{x_j}{\eta_{\mathtt{p}}} - \dfrac{\tau_{\mathtt{p}}}{\eta_{\mathtt{p}}} + \epsilon, & \forall \mathtt{r}(i,j) = \mathtt{p}^{-1} \text{ where } \epsilon \sim \mathcal{N}(0, \dfrac{\sigma_{\mathtt{p}}^2}{\eta_{\mathtt{p}}^2}). \end{cases} \tag{5}$$

This model builds a linear relationship between neighboring nodes by introducing relation-dependent scaling parameter η and a shift parameter τ. The latter case in (5) indicates the generative model yielded by the reverse relation, where the direction of the edge is reversed. 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.

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 as in (1), which casts the problem as maximizing the likelihood of node's first-hop neighbors. Once the likelihood of relational neighbors is expressed through the model in (5), the estimation can be found by minimizing the negative log-likelihood as in (2). Consequently, we obtain the following objective:

$$\underset{x_i}{\operatorname{argmin}} \sum_{\mathtt{p}\in\mathcal{P}} \left(\sum_{\mathtt{r}(i,j)=\mathtt{p}} \frac{\omega_{\mathtt{p}}}{2}\Big(x_i - \eta_{\mathtt{p}} x_j - \tau_{\mathtt{p}}\Big)^2 + \sum_{\mathtt{r}(i,j)=\mathtt{p}^{-1}} \frac{\omega_{\mathtt{p}}\eta_{\mathtt{p}}^2}{2}\Big(x_i - \frac{x_j}{\eta_{\mathtt{p}}} + \frac{\tau_{\mathtt{p}}}{\eta_{\mathtt{p}}}\Big)^2 \right), \qquad (6)$$

where we apply a change of parameter $\omega_p = 1/\sigma_{\mathtt{p}}^2$.

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

$$\hat{x}_i = \frac{\sum_{\mathtt{p}\in\mathcal{P}} \left(\sum_{\mathtt{r}(i,j)=\mathtt{p}} \omega_{\mathtt{p}}\Big(\eta_{\mathtt{p}} x_j + \tau_{\mathtt{p}}\Big) + \sum_{\mathtt{r}(i,j)=\mathtt{p}^{-1}} \omega_{\mathtt{p}}\eta_{\mathtt{p}}^2 \Big(\frac{x_j}{\eta_{\mathtt{p}}} - \frac{\tau_{\mathtt{p}}}{\eta_{\mathtt{p}}}\Big)\right)}{\sum_{\mathtt{p}\in\mathcal{P}} \left(\sum_{\mathtt{r}(i,j)=\mathtt{p}} \omega_{\mathtt{p}} + \sum_{\mathtt{r}(i,j)=\mathtt{p}^{-1}} \omega_{\mathtt{p}}\eta_{\mathtt{p}}^2 \right)}. \qquad (7)$$

3.1 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:

$$\underset{\tau_{\mathtt{p}},\eta_{\mathtt{p}}\,\omega_{\mathtt{p}}}{\operatorname{argmax}} \quad \mathrm{p}\Big(\big\{(x_i, x_j)\; \forall i, j \in \mathcal{V} \,|\, \mathtt{r}(i,j) = \mathtt{p}\big\} \;\big|\; \tau_{\mathtt{p}}, \eta_{\mathtt{p}}\, \omega_{\mathtt{p}}\Big). \qquad (8)$$

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:

$$\underset{\tau_{\mathtt{p}},\eta_{\mathtt{p}}\,\omega_{\mathtt{p}}}{\operatorname{argmax}} \quad \prod_{\mathtt{r}(i,j)=\mathtt{p}} \mathrm{p}\Big((x_i, x_j) \;\big|\; \tau_{\mathtt{p}}, \eta_{\mathtt{p}}\, \omega_{\mathtt{p}}\Big). \qquad (9)$$

We proceed with the minimization of negative log-likelihood to solve (9). The reader might recognize that its solution is equivalent to the parameters of a linear regression model [15]. This is simply because we introduce linear generative models (5) for the relationships existing on the graph. Therefore, the parameters can be found as follows ($\mu = \text{mean}(\mathbf{x})$ is the mean of node values):

$$\eta_{\mathtt{p}} = \frac{\sum_{\mathtt{r}(i,j)=\mathtt{p}}(x_i-\mu)(x_j-\mu)}{\sum_{\mathtt{r}(i,j)=\mathtt{p}}(x_j-\mu)^2}, \quad \begin{aligned} &\tau_{\mathtt{p}} = \text{mean}\Big(\big\{(x_i-\eta_{\mathtt{p}}x_j)\,\forall i,j\in\mathcal{V}\,|\,\mathtt{r}(i,j)=\mathtt{p}\big\}\Big),\\ &\omega_{\mathtt{p}} = 1/\text{mean}\Big(\big\{(x_i-\eta_{\mathtt{p}}x_j-\tau_{\mathtt{p}})^2\,\forall i,j\in\mathcal{V}\,|\,\mathtt{r}(i,j)=\mathtt{p}\big\}\Big).\end{aligned} \tag{10}$$

Local Generative Model and Local Operation. We summarize the local generative model, the associated loss and the first order estimate in simple and multi-relational cases in Table 1. In the multi-relational case, the neighborhood aggregation 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: $f(x) = x$ in simple case—no actual transformation applied, and $f_{\mathtt{p}}(x) = \eta_{\mathtt{p}}x + \tau_{\mathtt{p}}$ in relational case for type $\mathtt{p}$. 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 $\eta_{\mathtt{p}^{-1}} = 1/\eta_{\mathtt{p}}$, $\tau_{\mathtt{p}^{-1}} = \tau_{\mathtt{p}}/\eta_{\mathtt{p}}$, $\omega_{\mathtt{p}^{-1}} = \eta_{\mathtt{p}}^2\omega_{\mathtt{p}}$. Subsequent to the transformations, the estimation is computed by a weighted average that is controlled by the parameter ω. It is worth to notice that ω is set as the inverse of the error variance of the relational local generative model (5). Therefore, the estimate can be interpreted as the outcome of an aggregation with precision that ranks the relational information.

Table 1. Local generative model and operation

	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	$x_i = \eta_{\mathtt{p}}x_j + \tau_{\mathtt{p}} + \epsilon$ $\forall\mathtt{r}(i,j) = \mathtt{p}$ $\epsilon \sim \mathcal{N}(0, 1/\omega_{\mathtt{p}})$	$\sum_{\mathtt{p}\in\mathcal{P}\cup\mathcal{P}^{-1}} \sum_{\mathtt{r}(i,j)=\mathtt{p}} \omega_{\mathtt{p}}(x_i - \eta_{\mathtt{p}}x_j - \tau_{\mathtt{p}})^2$	$\frac{\sum_{\mathtt{p}\in\mathcal{P}\cup\mathcal{P}^{-1}} \sum_{\mathtt{r}(i,j)=\mathtt{p}} \omega_{\mathtt{p}} f_{\mathtt{p}}(x_j)}{\sum_{\mathtt{p}\in\mathcal{P}\cup\mathcal{P}^{-1}} \sum_{\mathtt{r}(i,j)=\mathtt{p}} \omega_{\mathtt{p}}}$

3.2 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. Similarly, we propose a propagation algorithm that relies on the first-order relational Bayesian estimate that is introduced in (6). The algorithm operates iteratively where the relational neighborhood aggregation (7) 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 denote matrix $\mathbf{A}_\mathrm{p}$ for encoding the adjacency pattern of relation type $\mathtt{p}$. It is $(N \times N)$ asymmetric matrix storing incoming edges on its rows and outgoing edges on its columns. One can compile aggregations in (7) simultaneously over the entire graph using a matrix notation. Then, the relational local operations at iteration-k can be expressed as

$$\mathbf{x}^{(k)} = \Bigg(\sum_{\mathrm{p}\in\mathcal{P}} \Big(\omega_\mathrm{p} \big(\eta_\mathrm{p} \mathbf{A}_\mathrm{p} \mathbf{x}^{(k-1)} + \tau_\mathrm{p} \mathbf{A}_\mathrm{p} \mathbf{1} \big) + \omega_\mathrm{p} \eta_\mathrm{p} \big(\mathbf{A}_\mathrm{p}^\top \mathbf{x}^{(k-1)} - \tau_\mathrm{p} \mathbf{A}_\mathrm{p}^\top \mathbf{1} \big) \Big) \Bigg) \odot \Bigg(\sum_{\mathrm{p}\in\mathcal{P}} \Big(\omega_\mathrm{p} \mathbf{A}_\mathrm{p} \mathbf{1} + \omega_\mathrm{p} \eta_\mathrm{p}^2 \mathbf{A}_\mathrm{p}^\top \mathbf{1} \Big) \Bigg)^{-1}, \quad (11)$$

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 (7) in vector format. Thus, it can be seen as the normalization factor over the neighborhood aggregation. For the purpose of simplification, we re-write (11) as

$$\mathbf{x}^{(k)} = (\mathbf{T}\mathbf{x}^{(k-1)} + \mathbf{S}\mathbf{1}) \odot (\mathbf{H}\mathbf{1})^{-1}, \quad (12)$$

by introducing the auxiliary matrices

$$\mathbf{T} = \sum_{\mathrm{p}\in\mathcal{P}} \eta_\mathrm{p}\omega_\mathrm{p}(\mathbf{A}_\mathrm{p} + \mathbf{A}_\mathrm{p}^\top),\ \mathbf{S} = \sum_{\mathrm{p}\in\mathcal{P}} \tau_\mathrm{p}\omega_\mathrm{p}(\mathbf{A}_\mathrm{p} - \eta_\mathrm{p}\mathbf{A}_\mathrm{p}^\top),\ \mathbf{H} = \sum_{\mathrm{p}\in\mathcal{P}} \omega_p(\mathbf{A}_\mathrm{p} + \eta_\mathrm{p}^2\mathbf{A}_\mathrm{p}^\top). \quad (13)$$

Algorithm. Given the iterations above, we now formalize the Multi-relational Propagation algorithm (MrP). 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 $\mathcal{U} \subseteq \mathcal{V}$. To manage propagation of continuous values at the labeled set of nodes towards the unlabeled, we introduce an indicator vector $\mathbf{u} \in \mathbb{R}^N$, which encodes the labeled nodes. 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 with the values over $\mathcal{U}$, and zero-padded at the unlabeled nodes, i.e., $\mathbf{x}_i^{(0)} = 0$ if $i \in \mathcal{V} \backslash \mathcal{U}$.

Similar to the label propagation [20], our algorithm fundamentally consists of aggregation and normalization steps. In order to encompass multi-relational transformations during aggregation, we formulate an iteration of MrP by the steps of aggregation, shift and normalization. In addition, similar to the Pagerank algorithm [5], we employ a damping factor $\xi \in [0, 1]$ to update a node's state by combining its value from the previous iteration. We provide a pseudocode for MrP in Algorithm 1. The propagation parameters for each relation type, $\{\tau_\mathrm{p}, \eta_\mathrm{p}, \omega_\mathrm{p}\}$ are estimated over the labeled set of nodes $\mathcal{U}$, as described in Sect. 3.1 and given to the algorithm as input together with the adjacency matrices encoding the multi-relational, directed graph. Steps 1–4 in Algorithm 1 are essentially responsible for the multi-relational neighborhood aggregation. At Step-5, 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 node's state. This adjusts 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 and ensures that the normalization complies with valid collected neighborhood information. Hence, at Step 6, we update it as well. Finally, at Step 7, we clamp labeled set of nodes[3] by leaving their values unchanged, simply because they store the governing information for completing the missing features. The algorithm terminates when all the nodes are propagated and the difference between two consecutive iterations is under a certain threshold. Accordingly, the number of iterations is related to the choice of hyperparameter ξ and the stopping criterion. MrP is implemented using PyTorch-scatter package[4], which efficiently computes neighborhood aggregation on a sparse relational structure. Thus, the aggregation steps require $2|\mathcal{E}|$ operations, then, normalization and update steps require $|\mathcal{V}|$ operations at each iteration. Therefore, MrP scales linearly with the number of edges in the graph, similar to the standard label propagation algorithm (LP). We finally note that by setting $\tau_\mathtt{p} = 0$, $\eta_\mathtt{p} = 1$, $\omega_\mathtt{p} = 1 \, \forall \mathtt{p} \in \mathcal{P}$, MrP drops down to LP[5] as if we operate on a simple graph regardless of the relation types and directions.

```
Algorithm 1: MrP
Input: $\mathcal{U}, \{x_i | i \in \mathcal{U}\}, \{\mathbf{A}_\mathtt{p}, \tau_\mathtt{p}, \eta_\mathtt{p}, \omega_\mathtt{p}\}_\mathcal{P}$
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: $\mathbf{z} = \mathbf{z} \odot \mathbf{r}^\dagger$        //$\dagger$ is for element-wise pseudo-inverse
    Step 5. Update values:

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

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

[3] Clamping step also exists in label propagation algorithm [20], which provides re-injection of true labels at each iteration throughout the propagation instead of overwriting the labeled nodes with the aggregated neighborhood information.

[4] Source code is available at https://github.com/bayrameda/MrAP. A special case of MrP is studied to propagate heterogeneous node features in [2] for numerical attribute completion in knowledge graphs.

[5] The label propagation algorithm [20] was originally designed for completing categorical features across a simple, weighted graph. We render it to propagate continuous features and to be applicable for the node regression by the default parameter set of MrP.

4 Experiments

We now present a proof of concept of the proposed multi-relational propagation method in several different node regression scenarios in different settings. We first test MrP in estimating weather measurements on a multi-relational and directed graph that connects the weather stations. Then, 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. 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. We calculate them over the estimation error on the unlabeled set of nodes. In the experiments, we leave the parameter η in MrP 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 estimate the parameter τ and ω for each relation type based on the observed set of node values as described in Sect. 3.1.

4.1 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[6]. In particular, we use yearly averages of weather measurements in our experiments.

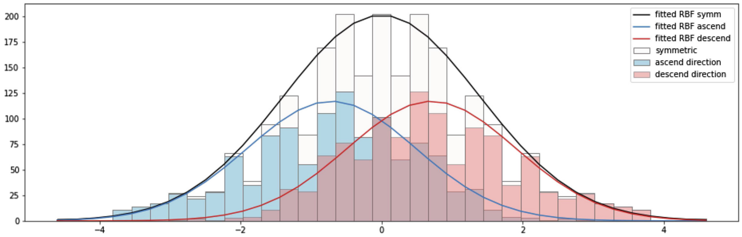

Fig. 3. Distribution of change in temperature measurements between weather stations that are related via altitude proximity in ascend and descend direction.

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

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

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

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 them based on geographical proximity by inserting an edge between a pair of weather stations if the Euclidean distance between their GPS coordinates is below a threshold. The geographical proximity leads to a symmetric relationship where we acquire 372 edges. Second, we relate them based on the altitude proximity in a similar logic yet we anticipate an asymmetric relationship where the direction of an edge indicates an altitude ascend from one station to another. For both relation types, we adjust the threshold for building connections so that there is not any disconnected node. Consequently, we acquire 1144 edges for the altitude relationship. 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.

Predicting Temperature and Snowfall on Directed Altitude Graph. We first conduct experiments on a simple scenario where we target predicting temperature and snowfall measurements using altitude relations. We compare the proposed method to the standard label propagation algorithm, LP, which overlooks asymmetric relational reasoning. Thus, we aim at evaluating the directional transformation utility of MrP during the neighborhood aggregation, which is mainly gained by the parameter τ and η. We visualize the distribution of measurement changes on the altitude edges in Fig. 3. Here, the parameter τ directly corresponds to the mean measurement difference computed along the directed altitude edges since $\eta = 1$. We fit radial basis function (RBF) to the distribution since the residual error in local generative model (5) is assumed to be normal. Then, the parameter ω is simply associated with the inverse of its variance. We see that the temperature differences in the ascend direction, *i.e.*, $\big\{(x_i - x_j)\forall \mathtt{r}(i,j) = \mathtt{altitude\_ascend}\big\}$, has a mean in the negative region. This signifies an expected decrease in temperature values along altitude ascend.

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

Predicting Precipitation on Directed, Multi-relational Graph. We test MrP in 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.

4.2 Predicting People's Date of Birth in a Social Network

We also conduct experiment on a small subset of a relational database called Freebase [16]. 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.*, $|\mathcal{P}| = 8$. Here, the task is to predict people's date of birth while it is only known for a subset of people. A fragment of the multi-relational graph is illustrated in Fig. 1, where two asymmetric relationships exist: `influenced_by` and `parent`. Table 4 summarizes the statistics for each. For symmetric relationships, the date of birth difference is counted along both directions of an edge, which sets the mean to zero. Also, the distributions over certain relation types are visualized in Fig. 4.

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

Table 4. Statistics for each relationship, columns respectively: number of edges, mean and variance of 'date of birth' difference over the associated relation type.

Relation type	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 5. Date of birth prediction performances

	Relation type	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

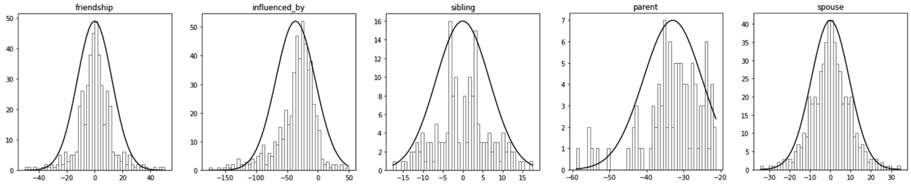

Fig. 4. Distribution of 'date of birth' difference (year) over different relationships. An RBF is fitted to each.

compare 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 relational neighborhood aggregation over different types of edges. 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.

5 Conclusion

In this study, we proposed MrP, a propagation algorithm working on multi-relational and directed graphs for regression of continuous node features and we show its superior performance on multi-relational data compared to standard propagation algorithm. 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 [11].

References

1. Aleta, A., Moreno, Y.: Multilayer networks in a nutshell. Annu. Rev. Condens. Matter Phys. **10**, 45–62 (2019)
2. Bayram, E., García-Durán, A., West, R.: Node attribute completion in knowledge graphs with multi-relational propagation. In: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3590–3594. IEEE (2021)
3. Bentley, B., et al.: The multilayer connectome of caenorhabditis elegans. PLoS Comput. Biol. **12**(12), e1005283 (2016)
4. Boccaletti, S., et al.: The structure and dynamics of multilayer networks. Phys. Rep. **544**(1), 1–122 (2014)
5. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst. **30**(1–7), 107–117 (1998)
6. Chami, I., Abu-El-Haija, S., Perozzi, B., Ré, C., Murphy, K.: Machine learning on graphs: a model and comprehensive taxonomy. CoRR abs/2005.03675 (2020). https://arxiv.org/abs/2005.03675
7. Chen, S., et al.: Signal inpainting on graphs via total variation minimization. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8267–8271. IEEE (2014)

8. Cozzo, E., de Arruda, G.F., Rodrigues, F.A., Moreno, Y.: Multilayer networks: metrics and spectral properties. In: Garas, A. (ed.) Interconnected Networks. UCS, pp. 17–35. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-23947-7_2
9. De, M.D.: Multilayer modeling and analysis of human brain networks. GigaScience **6**(5), 1–8 (2017)
10. Deng, J., et al.: Edge-aware graph attention network for ratio of edge-user estimation in mobile networks. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 9988–9995. IEEE (2021)
11. Dwivedi, V.P., Joshi, C.K., Laurent, T., Bengio, Y., Bresson, X.: Benchmarking graph neural networks. arXiv preprint arXiv:2003.00982 (2020)
12. Ivanov, S., Prokhorenkova, L.: Boost then convolve: gradient boosting meets graph neural networks. In: International Conference on Learning Representations (2021)
13. Opolka, F.L., Solomon, A., Cangea, C., Veličković, P., Liò, P., Hjelm, R.D.: Spatio-temporal deep graph infomax. arXiv preprint arXiv:1904.06316 (2019)
14. Perraudin, N., Vandergheynst, P.: Stationary signal processing on graphs. IEEE Trans. Signal Process. **65**(13), 3462–3477 (2017)
15. Rencher, A.C., Christensen, W.: Methods of multivariate analysis, chap. 3, pp. 47–90. Wiley (2012)
16. Toutanova, K., Chen, D.: 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 (2015)
17. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications, vol. 8. Cambridge University Press, Cambridge (1994)
18. Wu, Y., Tang, Y., Yang, X., Zhang, W., Zhang, G.: Graph convolutional regression networks for quantitative precipitation estimation. IEEE Geosci. Remote. Sens. Lett. **18**, 1124–1128 (2021)
19. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. In: Advances in Neural Information Processing Systems, pp. 321–328 (2004)
20. Zhu, X., Ghahramani, Z.: Learning from labeled and unlabeled data with label propagation. Technical report, Technical Report CMU-CALD-02-107, Carnegie Mellon University (2002)