Positive Unlabeled Link Prediction via Transfer Learning for Gene Network Reconstruction (Discussion Paper)

Paolo Mignone, Gianvito Pio and Michelangelo Ceci
paolo.mignone@uniba.it, gianvito.pio@uniba.it, michelangelo.ceci@uniba.it

Department of Computer Science - University of Bari Aldo Moro
Via Orabona, 4 - 70125 Bari (Italy)

Abstract. Transfer learning can be employed to leverage knowledge from a source domain in order to better solve tasks in a target domain, where the available data is exiguous. While most of the previous papers work in the supervised setting, we study the more challenging case of positive-unlabeled transfer learning, where few positive labeled instances are available for both the source and the target domains. Specifically, we focus on the link prediction task on network data, where we consider known existing links as positive labeled data and all the possible remaining links as unlabeled data. The transfer learning method described in this paper exploits the unlabeled data and the knowledge of a source network in order to improve the reconstruction of a target network. Experiments, conducted in the biological field, showed the effectiveness of the proposed approach with respect to the considered baselines, when exploiting the *Mus Musculus* gene network (source) to improve the reconstruction of the *Homo Sapiens Sapiens* gene network (target).

1 Introduction

The link prediction task aims at estimating the probability of the existence of an interaction between two entities on the basis of the available set of known interactions, which belong to the same data distribution, the same context and described according to the same features. However, in many real cases, the identical data distribution assumption does not hold, especially if data are organized in heterogeneous information networks [12]. For example, in the study of biological networks, collecting training data is very expensive and it is necessary to build link prediction models on the basis of data regarding different (even if related) contexts. At this regard, *transfer learning* strategies can be adopted to leverage knowledge from a source domain to improve the performance of a task solved on a target domain, for which we have few labeled data (see Figure 1). In the literature, we can find several applications where transfer learning approaches have been proved to be beneficial. For example, in the classification of Web documents, transfer learning approaches can be exploited to classify newly

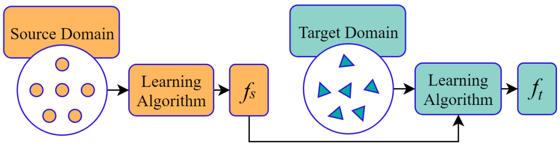

Fig. 1. Knowledge on a source task exploited to solve the target task.

created Web sites which follow a different data distribution [3] (e.g., the content is related to new subtopics). Another example is the work proposed in [9], where the authors exploit transfer learning approaches to adapt a WiFi localization model trained in one time period (source domain) to a new time period (target domain), where the available data possibly follow a different data distribution.

Focusing on the link prediction task, in the literature we can find several works in the biological field, since biological entities and their relationships can be naturally represented as a network. In the specific field of genomics, the working mechanisms of several organisms are usually modeled through gene interaction networks, where nodes represent genes and edges represent regulation activities. Since gene expression data are easy to obtain, several methods have been proposed in the literature that exploit this kind of data [7]. These approaches analyze the expression level of the genes under different conditions. Therefore, most machine learning approaches aiming to solve the link prediction task generally analyze gene expression data, with the final goal of reconstructing the whole network structure (Gene Network Reconstruction - GNR).

While existing methods generally work effectively on sufficiently large training data, a transfer learning approach could favor the GNR of specific organisms which are not well studied, by exploiting the knowledge acquired about different, related organisms. The main contribution of our work [8] is to evaluate the possible benefits that transfer learning techniques can provide to the task of link prediction. In particular, we exploit the available information about a source network for the reconstruction of a target network with poor available data. Moreover, we study the more challenging setting, i.e., the Positive-Unlabeled (PU) setting, where few positive labeled examples are available for both the domains, and no negative example is available [2, 11]. PU learning setting holds in many real contexts (e.g., text categorization, bioinformatics) where it is very expensive or unfeasible to obtain negative examples for the studied concept.

In order to evaluate the performance of the proposed method, we performed experiments in the biological domain. In particular, our experiments focused on the reconstruction of the human (Homo Sapiens Sapiens) gene network guided by the gene network of another, related organism, i.e., the mouse (Mus Musculus).

2 The proposed method

In this section, we describe our transfer learning approach to solve link prediction tasks in network data. Before describing it in details, we introduce some useful notions and formally define the link prediction task for a single domain. Let:

– V be the set of nodes of the network;
– $x = \langle v', v'' \rangle \in (V \times V)$ be a (possible) link between v' and v'', where $v' \neq v''$;
– $e(v) = [e_1(v), e_2(v), \ldots, e_n(v)]$ be the vector of features related to the node v, where $e_i(v) \in \mathbb{R}$, $\forall i \in \{1, 2, \ldots, n\}$;
– $e(x) = [e_1(v'), e_2(v'), \ldots, e_n(v'), e_1(v''), e_2(v''), \ldots, e_n(v'')]$ be the vector of features related to the link $x = \langle v', v'' \rangle$;
– $sim(a, b) \in [0, 1]$ be a similarity function between the vectors a and b;
– $l(x)$ be a function that returns 1 if the link x is a known existing link, and 0 if its existence is unknown;
– $L = \{x \mid x \in (V \times V) \wedge l(x) = 1\}$ be the set of labeled links;
– $U = (V \times V) \setminus L$ be the set of unlabeled links;
– $D = \{\widetilde{X}, P(X)\}$ be the domain described by the feature space $\widetilde{X} = \mathbb{R}^{2n}$, with a specific marginal data distribution $P(X)$, where $X = L \cup U$;
– $w(x)$ $(0 \leq w(x) \leq 1)$ be a computed weight for the link $x \in U$, which estimates the degree of certainty of its existence;
– $f(x)$ be an ideal function which returns 1 if x exists, and 0 otherwise.
The task we intend to solve is then defined as follows:
Given: a set of training examples $\{\langle e(x), w(x) \rangle\}_x$, each of which described by a feature vector and a weight.
Find: a function $f' : \mathbb{R}^{2n} \rightarrow [0, 1]$ which takes as input a vector of features $e(x)$ and returns the probability that x exists. Thus, $f'(e(x)) \approx \mathcal{P}(f(x) = 1)$ or, in other terms, f' approximates the probability distribution over the values of f.

Our method works with two different domains: the source domain $D_s = \{\widetilde{X_s}, P(X_s)\}$, and the target domain $D_t = \{\widetilde{X_t}, P(X_t)\}$, which are described according to the same feature space, i.e., $\widetilde{X_s} = \widetilde{X_t}$, while the marginal data distributions is generally different, i.e., $P(X_s) \neq P(X_t)$.

Given the two sets of labeled examples L_s and L_t, regarding the source and the target domain respectively, the method consists of three stages, that are summarized in Figure 2 and detailed in the following subsections.

Stage I - Clustering. The first stage of our method consists in the identification of a clustering model for the positive examples of each domain (i.e., on L_s and L_t). The application of a clustering method is motivated by the necessity to distinguish among possible multiple viewpoints of the underlying concept of positive interactions. Moreover, a summarization in terms of clusters' centroids becomes useful also from a computational viewpoint, since in the subsequent stages we can compare centroids instead of single instances. We adopt the classical *k-means* algorithm, since it is well established in the literature. However, any other prototype-based clustering algorithm, possibly able to catch specific peculiarities of the data at hand, could be plugged into our method.

Stage II - Instance Weighting. Although an unlabeled link could be either a positive or a negative example, we consider all the unlabeled examples as positive examples and compute a weight representing the degree of certainty in $[0, 1]$ of being a positive example: a value close to 0 (respectively, to 1) means that the

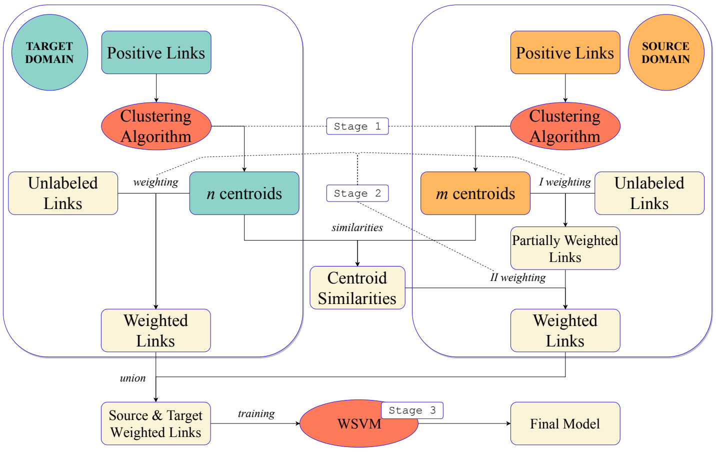

Fig. 2. A graphical overview of the proposed transfer learning approach.

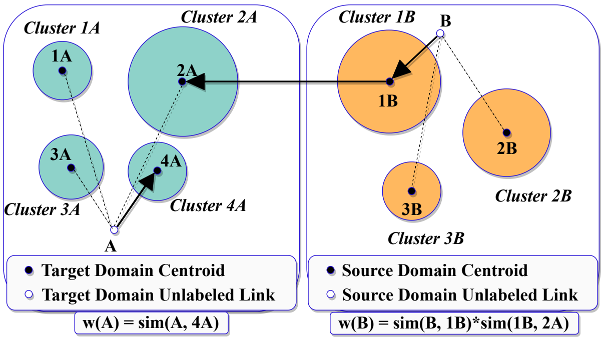

Fig. 3. Weighting process. The instance A is weighted according to the similarity to its most similar centroid (4A), while the instance B is weighted according to the similarity to its most similar centroid (1B), and to the similarity between 1B and 2A.

example is likely to be a negative (respectively, positive) example. The weight associated to the unlabeled instances of both the source and the target domains are computed according to their similarities with respect to the centroids obtained in the first stage. In particular, we identify a different weighting function for the source and target domains, in order to smooth the contribution provided by instances coming from the source domain.

Specifically, an unlabeled link x belonging to the target network (i.e., $x \in V_t \times V_t$) is weighted according to its similarity with respect to the centroid of its closest cluster, among the clusters identified from the target network. Formally:

$$w(x) = max_{c_t \in C_t}(sim(e(x), c_t)), \tag{1}$$

where C_t are the clusters identified from positive examples of the target network.

On the other hand, an unlabeled link x' belonging to the source network (i.e., $x' \in V_s \times V_s$) is weighted by considering two similarity values: *i)* the similarity with respect to the centroid of its closest cluster, computed among the clusters identified from the source network, and *ii)* the similarity between such a centroid and the closest centroid identified on the target network. Formally, let $c' = argmax_{c_s \in C_s}(sim(e(x'), c_s))$ be the closest centroid with respect to x' among the possible centroids C_s identified in the source network. Then:

$$w(x') = sim(e(x'), c') \cdot max_{c'' \in C_t}(sim(c', c'')). \tag{2}$$

As a similarity function, we exploit the Euclidean distance, after applying a min-max normalization (in the range $[0, 1]$) to all the features of the feature vectors. Formally, $sim(e(x'), e(x'')) = 1 - \sqrt{\sum_{k=1}^{n} \left(e_k(x') - e_k(x'')\right)^2}$.
An overview of the weighting strategy can be graphically observed in Figure 3.

Stage III - Training the classifier. In the third stage, we train a probabilistic classifier, based on linear Weighted Support Vector Machines (WSVM) [13] with Platt scaling [1], from the weighted unlabeled instances coming from both the source and the target networks. We selected an SVM-based classifier mainly because *i)* it has a (relatively) good computational efficiency, especially in the prediction phase, and *ii)* it already proved to be effective (with Platt scaling) in the semi-supervised setting [4]. At the end of the training phase, the WSVM classifier produces a model in the form of an hyperplane function h. Then, by exploiting the Platt scaling, for each unlabeled link x, we compute the probability of being a positive example as $f'(e(x)) = \frac{1}{1+e^{-h(e(x))}}$, where $h(e(x))$ is the score obtained by the learned WSVM. Finally, we rank all the predicted links in descending ordering with respect to the their probability of being positive. A pseudo-code representation of the proposed method is shown in Algorithm 1.

3 Experiments

Our experiments have been performed in the biological field. In particular, the specific task we intend to solve is the reconstruction of the human (*Homo Sapiens Sapiens - HSS*) gene network. As a source task, we will exploit the gene network of another, related organism, i.e., the mouse (*Mus Musculus - MM*).

3.1 Dataset

The considered dataset consists of gene expression data related to 6 organs (lung, liver, skin, brain, bone marrow, heart), obtained by the samples available at Gene Expression Omnibus (GEO), a public functional genomics repository. On overall, 161 and 174 samples were considered for MM and HSS, respectively, involving a total of $45,101$ and $54,675$ genes, respectively, each of which described by 6 features (one for each organ). Accordingly, the interactions among genes were described by 12-dimensional feature vectors, obtained by concatenating the feature vectors associated to the involved genes.

The set of validated gene interactions (verified positive examples) was extracted from the public repository BioGRID (https://thebiogrid.org).

Algorithm 1: PU Link Prediction via Transfer Learning

```
Data:
    ·$L_s = \{x_s \mid x_s \in (V_s \times V_s) \wedge l(x_s) = 1\}$: positive links of the source network
    ·$U_s = (V_s \times V_s) \setminus L_s$: unlabeled links of the source network
    ·$L_t = \{x_t \mid x_t \in (V_t \times V_t) \wedge l(x_t) = 1\}$: positive links of the target network
    ·$U_t = (V_t \times V_t) \setminus L_t$: unlabeled links of the target network
    ·$e(x)$: feature vector of the link $x$
    ·$sim(e(x'), e(x'')) = 1 - \sqrt{\sum_{k=1}^{n} (e_k(x') - e_k(x''))^2}$
    ·$k_1, k_2$: number of positive clusters for the source and the target network, respectively
Result:
    ·$ranked\_links$: predicted links ordered according to their likelihood
begin
    $C_s \leftarrow kmeans(L_s, k_1)$; $C_t \leftarrow kmeans(L_t, k_2)$;
    $source\_training\_set \leftarrow \emptyset$; $target\_training\_set \leftarrow \emptyset$; $ranked\_links \leftarrow \emptyset$;
    foreach $x_t \in U_t$ do
        $w(x_t) \leftarrow max_{(c_t \in C_t)}(sim(e(x_t), c_t))$;
        $target\_training\_set \leftarrow target\_training\_set \cup \{\langle e(x_t), w(x_t)\rangle\}$;
    foreach $x_s \in U_s$ do
        $source\_centroid \leftarrow argmax_{(c_s \in C_s)}(sim(e(x_s), c_s))$;
        $partial\_weight \leftarrow sim(e(x_s), source\_centroid)$;
        $centroid\_sim \leftarrow max_{(c_t \in C_t)}(sim(source\_centroid, c_t))$;
        $w(x_s) \leftarrow partial\_weight \cdot centroid\_sim$;
        $source\_training\_set \leftarrow source\_training\_set \cup \{\langle e(x_s), w(x_s)\rangle\}$;
    $training\_set \leftarrow source\_training\_set \cup target\_training\_set$;
    $h(\cdot) \leftarrow WSVM(training\_set)$;
    $f'(\cdot) = \frac{1}{1+e^{-h(\cdot)}}$;
    foreach $x \in U_t$ do
        $ranked\_links \leftarrow ranked\_links \cup \{\langle x, f'(e(x))\rangle\}$;
    $ranked\_links \leftarrow sort\_by\_score(ranked\_links)$;
    return $ranked\_links$;
```

3.2 Experimental Setting

Since our method exploits the k-means clustering algorithm, we performed the experiments with different values for k_1 (i.e., the number of clusters for the MM organism) and k_2 (i.e., the number of clusters for the HSS organism), in order to evaluate the possible effect of such parameters on the results. In particular, we considered the following parameter values: $k_1 \in \{2, 3\}, k_2 \in \{2, 3\}$. We remind that we work in the PU learning setting (i.e., the dataset does not contain any negative example). Therefore, inspired by [10], we evaluated the results in terms of the average recall@k, in a 10 fold CV setting. In particular, in order to quantitatively compare the obtained results, we draw the recall@k curve, by varying the value of k (i.e, the number of top-ranked interactions to consider as positive), and compute the area under the curve.

We compared our method, indicated as **transfer**, with two approaches:

- no_transfer, which corresponds to the WSVM with Platt scaling learned only from the target network (i.e., from the HSS network). This baseline allows us to evaluate the contribution of the source domain.

- union, which is the WSVM with Platt scaling learned from a single dataset consisting of the union of the instances coming from both MM and HSS. This baseline allows us to evaluate the effect of our weighing strategy.

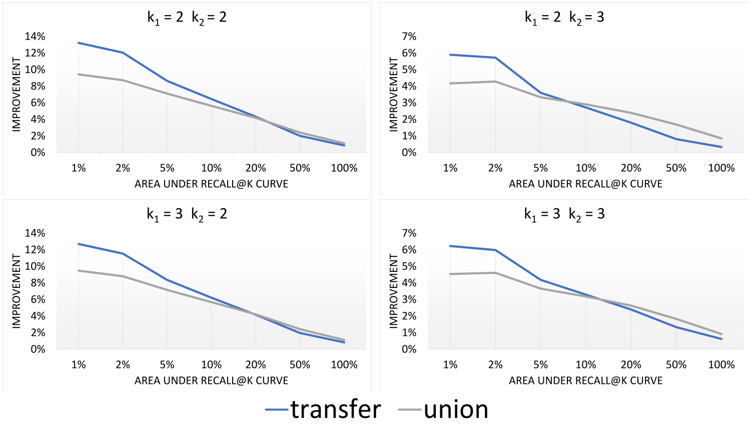

Fig. 4. Improvement over no_transfer, with different values of k_1 and k_2.

Since we are interested in observing the contribution provided by our approach with respect to the non-transfer approach, results will be evaluated in terms of improvement with respect to the **no_transfer** baseline.

3.3 Results

In Figure 4, we show the results obtained with different values of k_1 and k_2. In particular, we considered different percentages of the recall@k curve and measured the area under each sub-curve. This evaluation is motivated by the fact that biologists usually focus their in-lab studies on the analysis of the top-ranked predicted interactions. Therefore, a better result in the first part of the recall@k curve (i.e., at 1%, 2%) appears to be more relevant for the real biological application. The graphs show that both the **union** baseline and the proposed method (**transfer**) are able to obtain a better result with respect to the variant without any transfer of knowledge (**no_transfer**). This confirms that, in this case, the external source of knowledge (the MM gene network) can be exploited to improve the reconstruction of the target network (the HSS gene network).

By comparing the **union** baseline with our method, we can observe that the proposed weighting strategy was effective in assigning the right contribution to each unlabeled instance (coming either from the source or from the target network) in the learning phase. This is even more evident in the first part of the recall@k curve, where our method was able to retrieve about 120 additional true interactions at the top 1% of the ranking with respect to the baseline approaches.

Finally, by analyzing the results with respect to the values of k_1 and k_2, we can conclude that the highest improvement over the baseline approaches has been obtained with $k_1 = 2$ and $k_2 = 2$. This means that clustering can affect the results, and that even higher improvements could be obtained by adopting smarter clustering strategies that can, for example, catch and exploit the distribution, in terms of density, of the examples in the feature space.

4 Conclusion and Future Work

We proposed a transfer learning method to solve the link prediction task in the PU learning setting. By resorting to a clustering-based strategy, our method is able to exploit unlabeled data as well as labeled and unlabeled data of a different, related domain, identifying a different weight for each training instance. Focusing on biological networks, we evaluated the performance of the proposed method in the reconstruction of the Human gene network, supported by the mouse gene network. Results show that our method is able to improve the accuracy of the reconstruction, with respect to baseline approaches. As future work, we plan to implement a distributed version of the proposed method and to adopt some ensemble-based approaches [5, 6] to exploit multiple clusters in the prediction.

Acknowledgments

We would like to acknowledge the European project MAESTRA - Learning from Massive, Incompletely annotated, and Structured Data (ICT-2013-612944).

References

1. J. c. Platt. Probabilistic outputs for support vector machine and comparisons to regularized likelihood methods. In *Advances in large margin classifiers*, 1999.
2. M. Ceci, G. Pio, V. Kuzmanovski, and S. Džeroski. Semi-supervised multi-view learning for gene network reconstruction. *PLOS ONE*, 10(12):1–27, 12 2015.
3. W. Dai, Q. Yang, G. Xue, and Y. Yu. Boosting for transfer learning. In *Proc. of ICML*, pages 193–200, 2007.
4. C. Elkan and K. Noto. Learning classifiers from only positive and unlabeled data. In *Proc. of ACM SIGKDD*, pages 213–220, 2008.
5. J. Levatic, M. Ceci, D. Kocev, and S. Dzeroski. Self-training for multi-target regression with tree ensembles. *Knowl.-Based Syst.*, 123:41–60, 2017.
6. J. Levatic, D. Kocev, M. Ceci, and S. Dzeroski. Semi-supervised trees for multi-target regression. *Inf. Sci.*, 450:109–127, 2018.
7. D. Marbach and J. C. et al. Wisdom of crowds for robust gene network inference. *Nature Methods*, 2016.
8. P. Mignone and G. Pio. Positive unlabeled link prediction via transfer learning for gene network reconstruction. In *ISMIS 2018*, pages 13–23, 2018.
9. S. J. Pan, V. W. Zheng, Q. Yang, and D. H. Hu. Transfer learning for wifi-based indoor localization. *Workshop on Transfer Learning for Complex Task AAAI*, 2008.
10. G. Pio, M. Ceci, D. Malerba, and D. D'Elia. ComiRNet:a web-based system for the analysis of miRNA-gene regulatory networks. *BMC Bioinform*, 16(S-9):S7, 2015.
11. G. Pio, D. Malerba, D. D'Elia, and M. Ceci. Integrating microRNA target predictions for the discovery of gene regulatory networks: a semi-supervised ensemble learning approach. *BMC Bioinform*, 15(S-1):S4, 2014.
12. F. Serafino, G. Pio, and M. Ceci. Ensemble learning for multi-type classification in heterogeneous networks. *IEEE Trans. Knowl. Data Eng.*, 30(12):2326–2339, 2018.
13. X. Yang, Q. Song, and Y. Wand. A weighted support vector machine for data classification. In *Int J Pattern Recogn, Vol. 21*, 2007.