WCCI 2012 IEEE World Congress on Computational Intelligence
June, 10-15, 2012 - Brisbane, Australia IJCNN

Time Series Based Link Prediction

Paulo Ricardo da Silva Soares
Center of Informatics
Federal University of Pernambuco
Recife, Brazil prss@cin.ufpe.br

Ricardo Bastos Cavalcante Prudêncio
Center of Informatics
Federal University of Pernambuco
Recife, Brazil rbcp@cin.ufpe.br

***Abstract*—Link prediction is a task in Social Network Analysis that consists of predicting connections that are most likely to appear considering previous observed links in a social network. The majority of works in this area only performs the task by exploring the state of the network at a specific moment to make the prediction of new links, without considering the behavior of links as time goes by. In this light, we investigate if temporal information can bring any performance gain to the link prediction task. A traditional approach for link prediction uses a chosen topological similarity metric on non-connected pairs of nodes of the network at present time to obtain a score that is going to be used by an unsupervised or a supervised method for link prediction. Our approach initially consists of building time series for each pair of non-connected nodes by computing their similarity scores at different past times. Then, we deploy a forecasting model on these time series and use their forecasts as the final scores of the pairs. Our preliminary results using two link prediction methods (unsupervised and supervised) on co-authorship networks revealed satisfactory results when temporal information was considered.**

I. Introduction

With the advance of the Internet, people and organizations could interact and collaborate more, providing the basis for the emergence of social networks in virtual environments. A social network is a set of individuals joined together by some kind of relationship, such as friendship [3] in which two people are connected if they are friends. Since this kind of network is generally complex and highly dynamic, it is really important to understand its behavior along time [24].

Social Network Analysis (SNA) is a broad field of research that tries to cope with the dynamics of that kind of structure [25]. Different tasks can be associated to SNA. In this paper, our specific aim is to investigate the dynamics of links in a social network, that is, we want to predict what relationships are most likely to be formed based on previous states of the network. That is a well-known task treated by SNA called link prediction [2].

Several approaches can deal with the link prediction problem [27], [8], [23]. The most widespread one is based on topological patterns. Basically, it consists of applying topological similarity metrics to non-connected pairs of nodes in the network at a time t in order to predict if a link will occur at a time t' ($t' > t$). Such metrics provide scores to each pair of nodes which are then used to perform the prediction task either by an unsupervised or a supervised technique.

In the unsupervised technique, the pairs of non-connected nodes are ranked by their similarity scores and the top ranked ones are predicted to be connected. In the supervised strategy, in turn, the link prediction is treated as a classification problem. In this approach, pairs of nodes are assigned to positive class if they are connected, or negative class otherwise. The similarity scores related by a chosen set of topological metrics are adopted in this approach as features which are used by a supervised classifier to perform the link prediction task.

Although traditional approaches for link prediction have shown good results, they fail in exploring the network evolution as such, since they only statically analyze the network at the current moment, i.e., topological metrics are computed using all network data up to the present time without considering when links were created in the network. Our work tries to deal with this limitation by exploring how topological metrics evolve in the network over time. In order to accomplish this goal, each chosen similarity metric is applied to all non-connected pairs of nodes in the network in different past times. Then, a time series is built for each pair, recording the values provided by the metric. Finally, a forecasting model is applied to the series in order to predict their next values, which are going to be used as input to unsupervised and supervised link prediction methods. Although some works [11], [5], [21] tried to use time-aware techniques for link prediction, to the best of our knowledge, this work is the first one to investigate the prediction of links by using time series forecasting over similarity metrics.

In order to verify the viability of the proposed approach, we performed experiments on two co-authorship networks extracted from two sections of the physics e-Print arXiv[1]. In these experiments, we adopted similarity metrics from the state-of-the-art of link prediction, such as preferential attachment, common neighbors, Adamic/Adar, among others [4], [19], [1], [22]. Furthermore, we adopted both unsupervised [14], [18], [16] and supervised [15], [2], [6] methods to compare our results with the ones achieved by the traditional approach. In general, the experiments showed that our approach performs better than the traditional one for both unsupervised and supervised strategy.

Section II, briefly presents the link prediction problem. Section III in turn describes the proposed approach and presents the similarity metrics and forecasting models used. Section IV describes the social network data adopted, the experimental process and discusses the results. Finally, Section V concludes

[1] http://www.arxiv.org

the paper by presenting some considerations and future work.

II. Link Prediction

Link prediction consists of using previously observed network states to find hidden connections or predict links that are most likely to appear in the future [8]. There are several approaches to treat this problem. The most popular ones are based on node features (content and/or semantics), probabilistic models (relational learning) and topological/structural patterns.

In the node-wise based approach, similarity measures are adopted to associate nodes based on their content or semantics [27]. Nodes are represented as a vector of features they have, and similarity metrics are then applied to pairs of nodes aiming to find how close they are.

The probabilistic approach attempts to find a model that best represents the network. The idea is to build a probabilistic model defined by a set of parameters θ, estimated using the observed social network. Then the existence of a connection between a given pair of nodes x and y is determined by the conditional probability P($e^{<x,y>}|\theta$) [27]. Examples of models in this approach are Relational Markov Networks, Relational Bayesian Networks and Relational Dependency Networks.

The approach based on topological patterns of the network [27], [18], [10], [14] consists of extracting scores from non-connected nodes of the network by means of topological metrics (see section III-A). These metrics provide a degree of similarity between two nodes by exploring structural patterns of the network in analysis [14]. Those scores are then used as the basis for building models that can perform the prediction.

The topological based approach is the most widespread one. It also presents good performance and is easy to implement [27], [18], [10], [14]. Besides, topological metrics are focused on the network structure, which is what we aim to investigate considering the time dimension. That is why our proposed approach is based on such metrics.

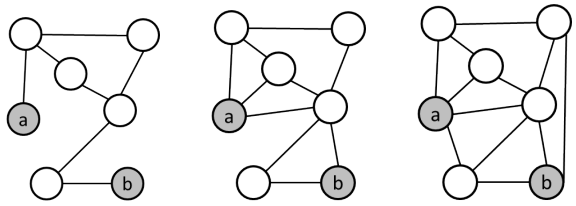
Traditionally, there are two strategies to deal with the prediction after the similarity scores are calculated: *unsupervised* and *supervised* approaches. The first one consists of ranking the non-linked pairs of nodes, positioning the ones with highest scores in the top of the list. Then the top L pairs of nodes are predicted to be linked. This method is very simple to implement and does not require a labeled training set to perform the prediction. However, it shows some limitations such as the need to define the threshold L and the difficulty in combining information provided by more than one metric.

The supervised learning strategy treats the link prediction task as a classification problem, in which pairs of nodes that are actually linked are assigned to class 1 (positive class), whereas the non-connected ones are assigned to class 0 (negative class). Unlike the unsupervised method, the supervised strategy requires a labeled training set to train the classifier that is going to be used. Finally, the prediction task can be performed by deploying a set of pairs of nodes in the trained classifier.

III. Proposed Approach

The link prediction problem has been traditionally defined as: "Given a snapshot of a social network at time *t*, we seek to accurately predict the edges that will be added to the network during the interval from time *t* to a given future time *t'*" [14]. Almost all previous work following this definition perform the link prediction by analyzing the complete network structure at the current time, without considering that the existing links were created in different moments in the past. Since they use a static structure of the network to predict new connections, these approaches do not consider changes in the network behavior over time and, consequently, they are not able to model its evolution as such.

Consider the example illustrated in Fig. 1. It can be seen that the nodes a and b are more active than the other ones in the network through time, i.e., new connections tend to be formed from them. This behavior can only be captured by a strategy that takes into account temporal information. The traditional approach, in contrast, is limited to explore a punctual state of the network, which does not bring any knowledge about the activeness of the nodes over time.



(a) t_0 (b) t_1 (c) t_2

Fig. 1: Example of network states at different times.

Some previous work tried to overcome this limitation. Berlingerio et al. [5] proposed to discover association rules that best explain the appearance of new connections and nodes. Huang et al. [11] fitted the occurrence of links into time series, using an autoregressive model to project their future values in order to measure the probability of new connections. Potgieter et al. [21] considered modeling topological metrics over time and applying some metrics aiming to calculate representative scores to pairs of non-connected nodes. These methods showed that temporal information improves the link prediction task, which stimulates us for further investigation.

The approach proposed in this work performs the prediction of new links by exploring the evolution of topological metrics. We addressed such evolution as a time series problem. Additionally, we used a set of well-known statistical forecasting models to estimate future values. Finally, we applied either unsupervised or supervised methods to link prediction.

The basic idea is to build time series for each non-connected pairs of nodes of the network using the similarity scores provided by a topological metric. A forecasting model is then used in order to predict the next value of the series. Such value is going to be the final score of the pair of nodes to be used by the link prediction methods.

First of all, the network G observed at time t must be split into several time-sliced snapshots, that is, states of the network at different times in the past. Afterwards, a *window of prediction* must be defined. It represents how further in the future we want to make the prediction. Then, consecutive snapshots are grouped in small sets which we call *frames*. Those contain as many snapshots as the length of the window of prediction. These frames compose what we denominate *Framed Time-Sliced Network Structure* ($\mathcal{S}$).

Let G_t be the graph representation of a network at time t. Let $[G_1, G_2, ..., G_T]$ be the frame formed by the union of the graphs from time 1 to T. Let n be the number of periods (frames) in the series. And let w be the window of prediction. Formally, $\mathcal{S}$ can be defined as:

$$\begin{aligned}\mathcal{S} = \{&[G_1, G_2, ..., G_w],\\ &[G_{w+1}, G_{w+2}, ..., G_{2w}], ...,\\ &[G_{(n-1)w+1}, G_{(n-1)w+2}, ..., G_{nw}]\}\end{aligned} \tag{1}$$

For instance, suppose that we observed a network from 2002 to 2007 and our aim is to predict links that will appear up to 2 years later. In this example, the forecast horizon (window of prediction) is 2 years by definition. In our proposal, we aim to model how the network evolves every 2 years in order to predict what will happen in the forecast horizon. Hence, we adopt a 2-years frame and $\mathcal{S}$ would be defined as follows: $\{[G_{2002}, G_{2003}], [G_{2004}, G_{2005}], [G_{2006}, G_{2007}]\}$. Furthermore, the frame we want to predict is, therefore, $[G_{2008}, G_{2009}]$.

Once the network structure is split, we follow the steps bellow:

1) Choose a similarity metric (e.g. preferential attachment, common neighbors,...);
2) For each pair of non-connected nodes:
   a) Create a time series by applying the chosen metric to the pair of nodes on each frame of $\mathcal{S}$;
   b) The score must be null for a frame in which any of the nodes in the pair does not exist;

   Once the time series was built, its forecast is computed by using a forecasting model:
3) Choose a forecasting model (e.g. moving average, simple exponential smoothing,...):
4) For each time series describing a non-connected pair of nodes:
   a) Set the score of the pair of nodes to be the one-step ahead prediction given by the chosen forecasting model when applied to the series.

After the scores were calculated, either unsupervised or supervised methods can be used to predict new links.

A. Similarity Metrics

In this section, we present the topological metrics used to support the prediction process. First, we introduce the notation that will help to understand the metrics. Let $\Gamma(x)$ be the set of neighbors of a node x in a graph; let $||A||$ be the cardinality of the set A. In our work, all the analyzed graphs are undirected and then $||\Gamma(x)||$ is the degree of the node x.

- *Preferential Attachment (PA)*
  The PA measure assumes that the probability of a future link between two nodes is proportional to their degrees. Barabasi et al. [4] and Newman [19] verified that for a pair of nodes in a co-authorship network, such probability is correlated to the product of the number of collaborators they have. Hence, it is defined as:
  $$PA(x, y) = ||\Gamma(x)|| \times ||\Gamma(y)|| \tag{2}$$
- *Common Neighbors (CN)*
  This metric states that the bigger the number of neighbors two nodes share, the higher is their probability to form a link in the future [19]. Formally, the metric is defined as:
  $$CN(x, y) = ||\Gamma(x) \cap \Gamma(y)|| \tag{3}$$
- *Adamic Adar (AA)*
  It refines the CN metric by increasing the importance of nodes which possess less connections [1]. Its formal definition is:
  $$AA(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{log(||\Gamma(z)||)} \tag{4}$$
- *Jaccard's Coefficient (JC)*
  It measures the probability of two nodes be linked by calculating the ratio between the number of neighbors they share and the total number of distinct neighbors they have [22]. It is defined as:
  $$JC(x, y) = \frac{||\Gamma(x) \cap \Gamma(y)||}{||\Gamma(x) \cup \Gamma(y)||} \tag{5}$$

B. Forecasting Models

In this section we present the forecasting models adopted in our work. Since the behavior of the time series in analysis is not known in advance, we used forecasting models with different underlying assumptions in order to verify the ones that best describe the network evolution. Complex models are potentially more efficient for large time series. However, the series we are working with are short (data is relatively recent), thus we chose simple models because they are easy to calculate, computationally efficient and have good performance [17].

We introduce here some useful notations. Let $Z_t (t = 1, ...T)$ be a time series with T observations. Let $\hat{Z}_t$ be the time series forecast at time t.

- *Moving Average (MA)*
  This method makes the prediction of a series by taking the mean of its n most recent observed values [26]. In our work, we defined n = 2 for simplicity. The moving average forecast at time t can be defined as:
  $$\hat{Z}_t = \frac{Z_{t-1} + Z_{t-2} + ... + Z_{t-n}}{n} \tag{6}$$
- *Average (Av)*
  This method is similar to moving average but it uses all

past values of the series. The forecast in this case is given by:

$$\hat{Z}_t = \frac{Z_{t-1} + Z_{t-2} + ... + Z_{t-T}}{T} \tag{7}$$

- *Random Walk (RW)*
  It considers as forecast the last observed value of the series. It can be seen as a moving average with n set to 1. The forecast in this method is defined as:

$$\hat{Z}_t = Z_{t-1} \tag{8}$$

- *Linear Regression (LR)*
  This model fits the series data to a straight line. The level $\hat{a}_t$ of the series and the trend $\hat{b_t}$ (which estimates the slope of the series [12]) are defined by minimizing the sum of the squared errors between the observed values of the series and expected values estimated by the model. The forecast is defined as:

$$\hat{Z}_{t+h} = \hat{a}_t + h\hat{b}_t \tag{9}$$

- *Simple Exponential Smoothing (SES)*
  This model forecasts a series by taking a weighted mean of its past values. In this model, the most recent past values would be more relevant to produce forecasts [26]. Given a weight α, its forecast is defined as:

$$\hat{Z}_t = \alpha Z_{t-1} + (1-\alpha)\hat{Z}_{t-1} \tag{10}$$

  As in the LR model, the value of the constant α was defined by minimizing the sum of squared errors produced by the model.

- *Linear Exponential Smoothing (LES)*
  If a series has some kind of tendency, the forecasts produced by methods such as SES may overestimate or underestimate the real observed values of the series, thus harming the forecasting accuracy [17]. The LES model refines the SES by adding a component β that takes into account short trends in the series. In this model, three equations are used to generate a forecast. They are:

$$\hat{Z}_{t+h} = S_t + hT_t \tag{11}$$

$$S_t = \alpha Z_t + (1-\alpha)(S_{t-1} + T_{t-1}) \tag{11a}$$

$$T_t = \beta(S_t - S_{t-1}) + (1-\beta)T_{t-1} \tag{11b}$$

  The basic equation (11) bears resemblance to LR model, but the components *intercept (level)* and *slope (trend)* are adjusted at each new observation of the series. The former is a smoothed estimate of the value of the data at the end of each period. The latter is a smoothed estimate of average growth at the end of each period [13]. In this work, α and β were estimated by minimizing the sum of squared errors produced by the model. Additionally, the initial values of S_t and T_t were set to Z_1 and $Z_2 - Z_1$ respectively.

IV. Experiments and Results

In this section, we describe the social network data used in our experiments, the methods adopted in the link prediction task and the results obtained by our proposed approach for link prediction compared to the traditional one.

A. Data

For the experiments developed in this work, we adopted co-authorship networks, which are social networks where the nodes represent the authors, who are connected to each other if they collaborated in a paper. This kind of network is widely used to understand topology and dynamics of complex networks, since it corresponds to the largest publicly available digitalized social network [20], [4].

We adopted two co-authorship networks from two sections of Arxiv[2]. The first network is composed by authors that collaborated in theoretical high energy physics area (hep-th[3]). The second one is formed by authors who published papers in high energy physics - lattice area (hep-lat[4]) (see Table I for information about the size of the networks). We extracted data from the year 1991 to 2010 for hep-th network and from 1993 to 2010 for hep-lat one.

TABLE I: Network size in terms of nodes and edges.

	hep-th	**hep-lat**
Authors	17917	4631
Collaborations	59013	31738

Regarding the proposed approach, we made predictions two years ahead. Therefore, snapshots from 2009 to 2010 were used to generate the test network, whereas the others were used to build $\mathcal{S}$ as described in section III.

In order to generate the test set, first we take all non-connected pairs in the training network (union of all frames in $\mathcal{S}$) as initial candidates to be analyzed. Then we assign each pair of nodes that is actually linked in test network to the positive class, or to the negative class otherwise.

The supervised learning requires a labeled training set so that a classifier can use it to build its classification model. To generate it, initially we set the last frame in $\mathcal{S}$ to be the validation network and take the other frames in $\mathcal{S}$ to compose the training network. Then we apply the same process we did to build the test set.

Since co-authorship networks are highly sparse, we need to reduce the number of candidate pairs in order to make computation feasible. Aiming to decrease it, from the list of initial candidates we chose only the ones that are two steps apart at most. Since the majority of the metrics we adopted are based on common neighbors, consequently, it has no sense to compute scores for pairs of nodes more than two steps apart [15].

[2]http://www.arxiv.org
[3]http://arxiv.org/archive/hep-th
[4]http://arxiv.org/archive/hep-lat

Table II and Table III present the class distribution of examples in the training and test sets. As it can be seen, class distributions are highly imbalanced, which is a common problem in link prediction task. Whereas unsupervised learning methods are unable to deal with imbalanced problem since they have no information about class distributions by definition, supervised learning strategies are able to balance data [15].

TABLE II: Test pairs of nodes distribution by class.

	hep-th	**hep-lat**
Non-linked (class 0)	458.933	295.111
Linked (class 1)	917	851
Total	459.850	295.962

TABLE III: Training pairs of nodes distribution by class.

	hep-th	**hep-lat**
Non-linked (class 0)	327.586	197.855
Linked (class 1)	983	1122
Total	328.569	198.977

Bearing that in mind, we used the strategy described in [15] to deal with the imbalance between negative and positive classes in the training set, which consisted of undersampling the majority class. This process resulted in a final training set containing 983 negative and 983 positive instances for the hep-th network, and 1122 negative and 1122 positive for the hep-lat one.

B. Experiments with Unsupervised Link Prediction

As described in section II, in the traditional unsupervised approach for link prediction, the pairs of non-connected nodes are ranked according to their scores defined by a chosen similarity metric. The top ranked pairs are then considered as the ones with highest probability of being connected in the future. In this experiment, we perform a similar approach however using the forecasts of a chosen similarity measure to rank the non-connected pairs of nodes.

Given a similarity metric and a forecasting model, we computed the forecast associated to each pair of nodes following the steps described in section III. Then, we ranked the scores and built ROC curves by evaluating the pairs from the top to the bottom of the ranked list. Finally, we computed the the Area Under the ROC curve (AUC), which was used as a measure of performance. The AUC is an important performance measure which has been traditionally used in imbalanced classification problems. It relates the sensitivity (true positive rate) and specificity (true negative rate) of a classifier [7]. The AUC was computed for each combination of similarity metrics and forecasting models described respectively in section III-A and III-B.

The results are summarized in Table IV and Table V. As it can be seen, the proposed approach outperforms the traditional unsupervised approach (last column) for almost all combinations of similarity metric and forecasting model adopted in our experiments. The traditional approach was better than the proposed approach only for some settings of experiments which adopted exponential smoothing methods.

TABLE IV: Results for unsupervised learning on test set from hep-th network.

	RW	MA	Av	LR	SES	LES	Trad
PA	**0.6889**	0.6768	0.5718	0.6456	0.6095	0.6202	*0.5039*
CN	0.6521	0.6720	0.6593	**0.6929**	*0.5427*	0.5477	0.6158
AA	0.6545	0.6761	0.6656	**0.6957**	*0.5434*	0.5484	0.6684
JC	0.6536	0.6736	0.6567	**0.6888**	*0.5418*	0.5478	0.5680

TABLE V: Results for unsupervised learning on test set from hep-lat network.

	RW	MA	Av	LR	SES	LES	Trad
PA	0.7761	**0.7902**	0.6932	0.7452	0.7007	0.7036	*0.5877*
CN	0.7429	0.7716	0.7580	**0.7730**	0.5745	*0.5741*	0.7318
AA	0.7468	0.7755	0.7637	**0.7768**	*0.5729*	0.5738	0.7531
JC	0.7442	**0.7712**	0.7483	0.7688	0.5707	*0.5702*	0.6634

The results suggest that the prediction is better with more conservative forecasting models, that is, the ones that make predictions by considering a gradual increase or decrease in the trend rather than a sharp raise or fall caused by the stronger influence of the most recent series' values (which is the case of the models based on exponential smoothing).

For more reliable results, the forecasting models were also evaluated with stratified 5-fold cross-validation. Since there is no training set in the unsupervised learning, the test set was split into folds and the AUC measure was computed for each fold. The results are summarized in Table VI and Table VII.

As in the absolute results, the LR was the best forecasting model for all metrics, except for the PA measure (for which the RW model provided the best results) in the hep-th network. In the hep-lat network, the absolute results showed better performance of the LR model against the others for the metrics CN and AA. On the other hand, statistical results pointed MA as the best forecasting model for CN metric and Av for AA metric.

In general, LR was the best forecasting model for hep-th network and MA had better performance for hep-lat network. This suggests that, for the former network, the activeness of the pairs of nodes along the network evolution is really important to make a good prediction. Since LR model can explore long-term trend in a series, it can reach better results.

For the hep-lat network, it seems that most recent data carries more information about future connections. That is why MA performs better than other forecasting models.

C. Experiments with Supervised Link Prediction

The traditional supervised approach treats the link prediction problem as a classification task, i.e., given a pair of nodes, classify it as belonging to a positive class if a link is likely to

TABLE VI: Relative average performance of forecasting models on the hep-th network with unsupervised learning.

	RW	MA	Av	LR	SES	LES	Trad
PA	**0.6840±0.0006**	0.6578±0.0008	0.5375±0.0003	0.6296±0.0001	0.6167±0.0016	0.6196±0.0001	*0.5165±0.0006*
CN	0.6482±0.0004	0.6705±0.0014	0.6754±0.0014	**0.7045±0.0003**	0.5464±0.0002	*0.5381±0.0001*	0.5943±0.0005
AA	0.6598±0.0014	0.6810±0.0013	0.6804±0.0004	**0.7274±0.0005**	*0.5549±0.0003*	0.5552±0.0009	0.6696±0.0002
JC	0.6691±0.0009	0.6678±0.0002	0.6484±0.0001	**0.6812±0.0001**	0.5697±0.0008	*0.5444±0.0002*	0.5829±0.0002

TABLE VII: Relative average performance of forecasting models on the hep-lat network with unsupervised learning.

	RW	MA	Av	LR	SES	LES	Trad
PA	0.7687±0.0004	**0.8128±0.0004**	0.6768±0.0014	0.7632±0.0003	0.7247±0.0010	0.7152±0.0003	*0.5933±0.0004*
CN	0.7303±0.0007	**0.7854±0.0004**	0.7421±0.0005	0.7678±0.0002	0.5725±0.0002	*0.5622±0.0001*	0.7184±0.0010
AA	0.7493±0.0008	0.7701±0.0009	**0.7766±0.0003**	0.7721±0.0004	*0.5654±0.0002*	0.5668±0.0001	0.7559±0.0006
JC	0.7726±0.0008	**0.7869±0.0005**	0.7743±0.0001	0.7703±0.0005	*0.5722±0.0002*	0.5791±0.0002	0.6612±0.0001

be observed or a negative class otherwise. Each pair of nodes in this approach is represented as a feature vector which is used in the classification. The features adopted in the traditional supervised approach are commonly the similarity measures describing the pair of nodes, computed using all the network data.

In our proposed supervised approach, we considered two distinct feature vectors. Initially, we adopted as feature vector the set of forecasts of all metrics described in section III-A (this feature vector is referred here as *fv*). Additionally, we combined both the aforementioned predicted scores and the measures adopted in the traditional approach. This feature vector is named here as *h-fv* (hybrid feature vector).

As classifier, we adopted the implementation of Support Vector Machine (SVM) from the WEKA environment [9] with default parameters (the SMO algorithm). As performance measure, we adopted the AUC measure as in the previous section.

The results of our experiments are presented in Table VIII and Table IX. From these tables, we can see that the supervised learning outperforms the unsupervised one (see previous section) for all forecasting models used. This result suggests that, when combined by the SVM, the predicted scores carry more information about the network evolution compared to the scores used in isolation.

TABLE VIII: Results for supervised learning on test set from hep-th network.

	RW	MA	Av	LR	SES	LES	Trad
fv	0.7392	**0.7541**	0.6871	0.7418	0.6228	0.6291	*0.6091*
h-fv	0.7185	**0.7450**	0.6997	0.7415	0.6420	0.6363	–

TABLE IX: Results for supervised learning on test set from hep-lat network.

	RW	MA	Av	LR	SES	LES	Trad
fv	0.8149	**0.8514**	0.7953	0.8297	*0.6215*	0.6303	0.7518
h-fv	**0.8401**	0.8393	0.8106	0.8337	0.7723	0.7701	–

For hep-th network, MA was the best forecasting model for both feature vectors (*fv* and *h-fv*). In turn, for the hep-lat network, RW showed better results considering the *h-fv* method. There was also an improvement in performance when we considered hybrid feature vectors in the hep-lat network, suggesting that both information sources (time-aware as well as static information) are important for the link prediction task on that network.

As we did with the unsupervised method, we also performed a stratified 5-fold cross-validation for the supervised learning. In order to preserve temporal information, the stratification was done such that the test folds always contained data more recent than the training folds. The results are presented in Table X and Table XI.

As it can be seen, the statistical results are not much different from the absolute ones. The only difference is in the forecasting model that had the best performance considering the *h-fv* method in hep-lat network, which in absolute terms was RW and regarding the relative results was MA.

It is also important to highlight that the assumption we did in the unsupervised learning regarding the best forecasting models for hep-th network is not valid for the supervised method. Here, MA showed better results than LR model considering both methods (*fv* and *h-fv*).

In general, forecasting models that take into account most recent observed values in the series obtained better performance for the supervised network. Compared to the results obtained in the unsupervised method, this suggests that when topological metrics are used in isolation, more temporal information is needed to perform a good prediction. On the other hand, when combined to each other, the need for strong time-aware background is reduced. Thus, forecasting models which analyze short-term trend in a series can reach better results.

V. Conclusion

This work focused on performing a link prediction task by using time series to model the evolution of topological metrics. Most of previous works are based on the classical approach, which deals with the problem by making punctual analysis in

TABLE X: Relative average performance of forecasting models on the hep-th network with supervised learning.

	RW	MA	Av	LR	SES	LES	Trad
fv	0.7393±0.0212	**0.7542±0.0132**	0.6870±0.0239	0.7414±0.0136	0.6228±0.0193	0.6284±0.0120	*0.6070±0.0263*
h-fv	0.7177±0.0301	**0.7418±0.0143**	0.6969±0.0135	0.7400±0.0232	0.6378±0.0222	0.6327±0.0213	–

TABLE XI: Relative average performance of forecasting models on the hep-lat network with supervised learning.

	RW	MA	Av	LR	SES	LES	Trad
fv	0.8152±0.0102	**0.8246±0.0616**	0.7952±0.0151	0.8230±0.0090	*0.6216±0.0228*	0.6295±0.0132	0.7520±0.0065
h-fv	0.8396±0.0139	**0.8402±0.0157**	0.8086±0.0123	0.8330±0.0189	0.7732±0.0172	0.7703±0.0175	–

the network [18], [16], [6], not considering its evolution over time.

From our analysis, we could notice that time-aware information brings a performance gain to the link prediction problem considering both unsupervised and supervised learning. Furthermore, when this kind of information is gathered with static one (obtained by the traditional approach) the results are improved in some networks.

We also noticed that the combination of time-aware information reduces the need for strong temporal background, which benefits forecasting models that explore most recent values in the series, i.e., most recent states of the network.

This work still shows limitations regarding the number of networks used in the experiments and their domains. Although co-authorship networks have been used in several works in the literature [14], [15], [6], our aim is to broaden our investigations by exploring the consequences of time-aware link prediction in other social networks.

REFERENCES

[1] Lada A. Adamic and Eytan Adar. Friends and neighbors on the web. *Social Networks*, 25(3):211–230, 2003.

[2] M Al Hasan, Vineet Chaoji, Saeed Salem, and Mohammed Zaki. *Link Prediction using Supervised Learning*. Citeseer, 2006.

[3] L. A. Amaral, A. Scala, M. Barthelemy, and H. E. Stanley. Classes of small-world networks. *Proc Natl Acad Sci U S A*, 97(21):11149–11152, 2000.

[4] A. L. Barabási, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek. Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and its Applications*, 311(3-4):590 – 614, 2002.

[5] Michele Berlingerio, Francesco Bonchi, Björn Bringmann, and Aristides Gionis. Mining graph evolution rules. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part I*, ECML PKDD '09, pages 115–130. Springer-Verlag, 2009.

[6] H.R. de Sa and R.B.C. Prudencio. Supervised link prediction in weighted networks. In *Proceedings of the The 2011 International Joint Conference on Neural Networks*, IJCNN '11, pages 2281–2288. IEEE Computer Society, 2011.

[7] Tom Fawcett. An introduction to roc analysis. *Pattern Recogn. Lett.*, 27:861–874, June 2006.

[8] Lise Getoor and Christopher P. Diehl. Link mining: A survey. *SigKDD Explorations Special Issue on Link Mining*, 2005.

[9] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11:10–18, November 2009.

[10] Zan Huan. Link prediction based on graph topology: The predictive value of the generalized clustering coefficient. aug 2006.

[11] Zan Huang and Dennis K. J. Lin. The time-series link prediction problem with applications in communication surveillance. *INFORMS J. on Computing*, 21:286–303, April 2009.

[12] P.A. Jensen and J.F. Bard. *Operations research: models and methods*. Number v. 1. Wiley, 2003.

[13] Prajakta S Kalekar and Bernard. Time series forecasting using holt-winters exponential smoothing under the guidance of. *Technology*, (04329008):1–13, 2004.

[14] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.*, 58:1019–1031, May 2007.

[15] Ryan N. Lichtenwalter, Jake T. Lussier, and Nitesh V. Chawla. New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 243–252. ACM, 2010.

[16] Linyuan L and Tao Zhou. Link prediction in weighted networks: The role of weak ties. *EPL (Europhysics Letters)*, 89(1):18001, 2010.

[17] P.A. Morettin and C.M.C. Toloi. *Modelos para previsão de séries temporais*. Modelos para previsão de séries temporais. Instituto de Matemática Pura e Aplicada, 1981.

[18] Tsuyoshi Murata and Sakiko Moriyasu. Link prediction based on structural properties of online social networks. *New Generation Computing*, 26(3):245–257, 2008.

[19] M. E. J. Newman. Clustering and preferential attachment in growing networks, April 2001.

[20] M. E. J. Newman. The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences of the United States of America*, 98(2):404–409, January 2001.

[21] Dr. A. Potgieter, Prof K. A. April, R. J. E. Cooke, and I. O. Osunmakinde. Temporality in link prediction: Understanding social complexity, 2007.

[22] Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[23] Chao Wang, Venu Satuluri, and Srinivasan Parthasarathy. Local probabilistic models for link prediction. In *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, pages 322–331, Washington, DC, USA, 2007. IEEE Computer Society.

[24] Fei-Yue Wang, Kathleen M. Carley, Daniel Zeng, and Wenji Mao. Social computing: From social informatics to social intelligence. *IEEE Intelligent Systems*, 22:79–83, 2007.

[25] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications*. Number 8 in Structural analysis in the social sciences. Cambridge University Press, 1 edition, 1994.

[26] S.C. Wheelwright and S.G. Makridakis. *Forecasting methods for management*. Systems and controls for financial management series. Wiley, 1985.

[27] Evan Wei Xiang. A survey on link prediction models for social network data. *Science And Technology*, 2008.