Interaction Prediction in Dynamic Networks exploiting Community Discovery

Giulio Rossetti*†, Riccardo Guidotti*†, Diego Pennacchioli†, Dino Pedreschi* and Fosca Giannotti†
*University of Pisa, Italy Email: {rossetti,guidotti,pedre}@di.unipi.it
†ISTI-CNR, Pisa Italy Email: {name.surname}@isti.cnr.it

***Abstract*—Due to the growing availability of online social services, interactions between people became more and more easy to establish and track. Online social human activities generate digital footprints, that describe complex, rapidly evolving, dynamic networks. In such scenario one of the most challenging task to address involves the prediction of future interactions between couples of actors. In this study, we want to leverage networks dynamics and community structure to predict which are the future interactions more likely to appear. To this extent, we propose a supervised learning approach which exploit features computed by time-aware forecasts of topological measures calculated between pair of nodes belonging to the same community. Our experiments on real dynamic networks show that the designed analytical process is able to achieve interesting results.**

I. Introduction

Networks are rarely used to model static entities: i.e if we consider a social context we can observe that as time goes by new users appear and disappear, new interactions take place and existing ones fell apart disrupting existing paths. Understanding these dynamics is the first step to obtain insights on the real nature of the phenomenon modeled by the observed network. Moreover, almost all the network problems can be reformulated in order to take into account the temporal dimension: communities can be tracked through all their life cycle to unveil their history; incremental ranking can be computed in order to optimize execution costs; links can be predicted using information obtained by the analysis of topology changes in the local surroundings of nodes. Networks taking into account the temporal dimension are called *dynamic*.

In order to analyze dynamic networks in a reliable way, the social features affecting their structure and behavior must be considered. The problem of predicting the existence of hidden links or the creation of new ones in social networks is commonly referred to as the *link prediction* problem. In this work we propose an analytical process which, exploiting well-known state of art techniques, is able to tackle this challenging task in dynamic networks.

Several works highlight that, when addressing link prediction through supervised learning, it does not appear to exist a set of features or a similarity index that is outperforming in all settings: depending on the network analyzed various measures could be particularly promising or not [13]. This suggests that the predictors which work best for a given network may be related to the structure within the network rather than a universal best set of predictors. Moreover, it has been extensively observed that social networks exhibit community structure: such topologies provide bounds to the sociality of the users within them. In a dynamic scenario, more than in a static one, the evolution of such bounds describe changes in people' social behaviors.

To this extent, in this paper we propose a data mining process which addresses a particular formulation of the link prediction problem for dynamic networks, called Interaction Prediction. Our approach predicts future interactions by combining dynamic social networks analysis, time series forecast, feature selection and network community structure. The proposed method takes advantage on the prediction by focusing the attention on the links within the communities. This choice has two primary goals: *(i)* to restrict the prediction to the edges among users belonging to the same social context, and *(ii)* to overcome the computational drawbacks caused by the sparseness of dynamic social structures. Our experiments on real world interaction networks show that the proposed approach achieves encouraging results both in case of balanced and unbalanced class distribution.

II. Link Prediction

The classic formulation of *Link Prediction* involves the use of the observed network status to predict new edges that are likely to appear in the future or to unveil hidden connections among existing nodes. However, graph structures are often used to describe rapid-scale human dynamics: social interactions, call graphs, buyer-seller scenarios and scientific collaborations are only few examples. For this reason, in this work our aim is to exploit the temporal information carried by the appearance and disappearance of edges in a fully dynamic context. To model rapid scale dynamics we will adopt the *interaction network* model:

Definition 1 (Interaction Network)**.** *An interaction network $G = (V, E, T)$, is defined by a set of nodes V and a set of timestamped edges $E \subseteq V \times V \times T$ describing the interactions among them. An edge $e \in E$ is thus described by the triple (u, v, t) where $u, v \in V$ and $t \in T$. Each edge e represents an interaction between nodes u and v that took place at time t.*

To easily analyze an interaction network G we discretize it into τ consecutive snapshots of the same duration, thus obtaining a set of graphs $\mathcal{G} = \{G_0 \ldots, G_\tau\}$. Moreover, we assume that the interaction belonging to G_t are only the ones that appear in the interval $(t, t + 1)$. Such modeling choice allows us to make predictions not only for interactions that will take place among previously unconnected nodes, but also

to predict edges that have already appeared in the past. This decision is made in order to better simulate the dynamics that real interaction networks exhibit allowing nodes and edges both to rise and fall. In real interaction networks this model is a better proxy for structural dynamics since it allows to implicitly assign a time to leave to node-node links (i.e. in a call graphs it enables to weight more recent interactions w.r.t. older ones when predicting future contacts among a pair of nodes). Due to the adoption of this more complex model, hereafter we will refer to a peculiar problem formulation: *Interaction Prediction* problem:

Definition 2 (Interaction Prediction)**.** *Given a set* $\mathcal{G} = \{G_0, \dots G_t, \dots G_\tau\}$ *of ordered network observations, with* $t \in T = \{0 \dots \tau\}$*, the* interaction prediction *problem aims to predict new interactions that will took place at time* $\tau + 1$ *thus composing* $G_{\tau+1}$.

III. Time-Aware Interaction Prediction

In this section we introduce our analytical workflow, built upon a supervised learning strategy, designed to solve the Interaction Prediction problem. Our idea is to make use of time stamped network observations and community knowledge besides classical features in order to learn a robust machine learning model able to forecast new interactions. We design our approach to follow four steps:

Step 1: For each temporal snapshot $t \in T$ we compute a partition $\mathcal{C}_t = \{C_{t,0}, \dots, C_{t,k}\}$ of G_t using a community discovery algorithm. Then we define, for each t and C, $G_{C_t} = (V_{t,C}, E_{t,C})$ as the sub-graph induced on G_t by the nodes in C_t, such that $V_{t,C} \subseteq V_t$ and $E_{t,C} \subseteq E_t$.

Step 2: For each $t \in T$, we consider the interaction communities $\mathcal{C}_t$ of G_t and compute a set of measures F for each pair of nodes pair $(u, v) \in W_{t,\mathcal{C}}$ such that $W_{t,\mathcal{C}} = \{(u, v) : u, v \in V_{t,C} \wedge C_t \in \mathcal{C}_t\}$, that is (u, v) belong to the same community at time t. Thus we obtain values $f_t^{u,v}$ describing structural features, topological features and community features of the node pairs (u, v) at time t.

Step 3: With these values, for each couple of nodes $(u, v) \in W_{t,\mathcal{C}}$ and feature $f \in F$ we build a time series $S_f^{u,v}$ using the sequence of measures $f_0^{u,v}, f_1^{u,v}, \dots, f_\tau^{u,v}$. Then, we apply well-known forecasting techniques in order to obtain its future expected value $f_{\tau+1}^{u,v}$.

Step 4: Finally, we use the set of expected values $f_{\tau+1}^{u,v}$ for each feature $f \in F$ to build a classifier that will be able to predict future intra-community interactions.

A. Step 1: Community Discovery

In order to evaluate the impact of community structure on the predictive power of the proposed supervised learning strategy, we tested three different CD algorithms, namely: *Louvain*, *Infohiermap* and *DEMON*. *Louvain* is an heuristic method based on modularity optimization [4]. It is fast and scalable on very large networks and reach high accuracy on ad hoc modular networks. *Infohiermap* is one of the most accurate and best performing hierarchical non-overlapping clustering algorithm for community discovery [16] studied to optimize community conductance. *DEMON* is an incremental and limited time complexity algorithm for community discovery [6].

We chose to use the aforementioned algorithms since, due to their formulations, they cover three different kind of community definitions: modularity, conductance and density based ones. Since in our test we vary the structural properties of the communities used to extract the classification features, in the experimental analysis we will be able to discuss which network partitioning approach is able to provide more useful insights on future interactions.

B. Step 2: Features Design

In order to efficiently approach the Interaction Prediction task using a supervised learning strategy, it is crucial to identify and calculate a valuable set of features to train the classifier. Moving from the results of [8], [23] , we decided to use information belonging to three different families: *pairwise structural features*, *global topological features* and *community features*.

a) Pairwise Structural Features. In this class fall all the measures used in literature to score the likelihood of new links in unsupervised scenarios. Starting from the measures proposed in [13] we restricted our set to the one in Table I. Given a graph G, $\Gamma(u)$ identifies the set of neighbors of a node u in G; $|\bullet|$ represents the cardinality of the set $\bullet$.

b) Global Topological Features. In literature a wide set of measures were proposed to estimate the centrality of nodes and edges as well as their rank within a network. These scores are, often, computationally expensive to calculate: for this reason we have decided to make use only of two of them whose

Measure	Description
Common Neighbors[12]	$CN(u,v) = \lvert\Gamma(u) \cap \Gamma(v)\rvert$
Jaccard Coefficient[17]	$JC(u,v) = \frac{\lvert\Gamma(u)\cap\Gamma(v)\rvert}{\lvert\Gamma(u)\cup\Gamma(v)\rvert}$
Adamic Adar[1]	$AA(u,v) = \sum_{w\in\Gamma(u)\cap\Gamma(v)} \frac{1}{\log\lvert\Gamma(w)\rvert}$
Preferential Attachment[2]	$PA(u,v) = \lvert\Gamma(u)\rvert \times \lvert\Gamma(v)\rvert$

TABLE I: Pairwise Structural Features

Measure	Description
Degree Centrality	$DC(u) = \lvert\Gamma(u)\rvert$
Page Rank[14]	$PR(u) = \frac{1-d}{N} + d\sum_{(u,v)\in E} \frac{PR(v)}{\lvert\Gamma(v)\rvert}$

TABLE II: Global Topological Features

Measure	Description
Community Size	$CE(G_C) = \lvert E_C\rvert$
Community Edges	$CE(G_C) = \lvert E_C\rvert$
Shared Communities	$CS(u,v,\mathcal{C}) =$ number of communities shared by u and v
Community Density	$D(C) = \frac{\lvert E_C\rvert}{\lvert V_C\rvert\times(\lvert V_C\rvert-1)}$
Transitivity	$T = 3\frac{\lvert triangles(G_C)\rvert}{\lvert triads(G_C)\rvert}$
Max Degree	$MD(C) = max\{\lvert\Gamma(u)\rvert : u \in V_C\}$
Average Degree	$AD(C) = \frac{\sum_{u\in V_C}\lvert\Gamma(u)\rvert}{\lvert V_C\rvert}$

TABLE III: Community Features

Network	Nodes	Interactions	#Snapshots	μ_{CC}	σ_{CC}	μ_D	σ_D
DBLP	747,700	5,319,654	10 (years)	0.665	0.018	3.113e-05	9.602e-06
Social	1,899	113,145	6 (months)	0.105	0.015	8.600e-03	1.400e-03

TABLE IV: Networks statistics: average density μ_D, average clustering coefficient μ_{CC} and their standard deviations, σ_D and σ_{CC}.

Measure	Description
Last Value (Lv)	$\Theta_t = Z_{t-1}$
Average (Av)	$\Theta_t = \frac{\sum_{i=1}^{T} Z_i}{}$
Moving Average (Ma)	$\Theta_t = \frac{\sum_{i=\tau-n}^{\tau} Z_i}{n}$
Linear Regression (LR)	$\Theta_{t+h} = \alpha_t + h\beta_t$

TABLE V: Time Series Forecasting Approaches

definition is reported in Table II. DC and PR[1] scores were computed for both the endpoints of possible edges pairs.

c) Community Features. One of the most pressing issue related to LP regards the reduction of *false positive* forecasts. We exploit community discovery as a way to reduce the number of predictions provided by the chose pairwise structural features. Making predictions only between nodes belonging to the same community allows the predictive process to focus on connections that are more likely to appear, thus discarding the ones connecting different graph substructures. With this aim we introduce a third set of features, summarized in Table III.

C. Step 3: Forecasting Models

The third step of our approach involves the adoption of time series forecasting models to obtain, given subsequent observation of the same feature for the same pair of nodes, an estimation of its future value. Since the behavior of the observed time series is not known in advance, we adopt several forecasting models based on different underlying assumptions. In Table V we summarize the forecasting approaches tested: in our definitions we identify with $Z_t = (t = 1 \dots \tau)$ a time series with τ observations and with Θ_t its forecast at time t. In particular in Ma the prediction is made by taking the mean of the n most recent observed values of a series Z_t. In our experiments we have ranged n in the interval $[1, \tau]$. LR, on the other hand, fits the time series to a straight line. The level α and the trend β parameter (used to estimate the slope of the line) were defined by minimizing the sum of squared errors between the observed values of the series and the expected ones estimated by the model.

D. Step 4: Classifier Models

Predicting correctly new interactions is not an easy task. The complexity is mainly due to the highly unbalanced class distribution that characterizes the solution space: real world networks are generally sparse, thus the number of new interactions over the total possible ones tend to be small. We have discussed how it is possiblemitigate this problem by restricting the prediction set (i.e. predicting only new edges among nodes that, during the network history, were involved at least in a common community).

However, even adopting such precautions we can expect a substantial unevenness between the positive and the negative classes. This translates into a very high threshold for the baseline model (i.e. in case of a network having density 0.1, which identifies the presence of "only" $1/10$ of the possible edges, the majority classifier is capable to reach more than 0.9 ACC by simply predicting the absence of new interactions).

To overcome this issue we adopted class balancing through downsampling (as performed in previous works [10]), thus obtaining balanced classes and a baseline model having 0.5 accuracy. Indeed we pursued such approach but, in order to provide an estimate of the real predictive power expressed by our methodology, we also tested it against real highly unbalanced class distribution. In the following section we will discuss results achieved by an ensemble of classifiers showing the scores for the best performing classifier among Decision Tree (C4.5, C&R, CHAID, QUEST), Neural Network and Logistic Regression.

IV. Experiments and Results

We tested our approach on two networks: an interaction network obtained from a Facebook-like[2] *Social* network, and a co-authorship graph extracted from *DBLP*[3]. The general statistics of the datasets are shown in Table IV, while a brief resume is the following.

Social: The Facebook-like social network originates from an online community for students at UC Irvine. The dataset includes the users that sent or received at least one message during 6 months. We discretized the network in 6 monthly snapshot and we used the first 5 to compute the features needed to predict the edges present in the last one.

DBLP: We extracted author-author relationships if two authors collaborated at least in one paper. The co-authorship relations fall in temporal window of 10 years (2001-2010). The network is discretized on yearly basis: we used the first 9 years to compute the features and set as target for the prediction the edges belonging to the last one.

A. Balanced scenario

It happens frequently that the two classed to be predicted, i.e. there will be a link or not, are highly unbalanced. In our case we have highly unbalanced dataset with a proportion of unlinked-linked of $95.95\% - 4.055\%$ for Social, and of $98.13\% - 1.87$ for DBLP. As a first test, following what is generally done in literature, we balanced every snapshot G_t for Social and DBLP.

To evaluate the performances of the classifiers we used the *accuracy* and *AUC* which are defined in terms of the confusion matrix of a binary classifier:

Accuracy, defined as $ACC = \frac{TP+TN}{TP+FN+TN+FP}$, measures the ratio of correct prediction over the total;

[1] Dumping factor fixed at 0.85.

[2] http://toreopsahl.com/datasets/

[3] http://dblp.org

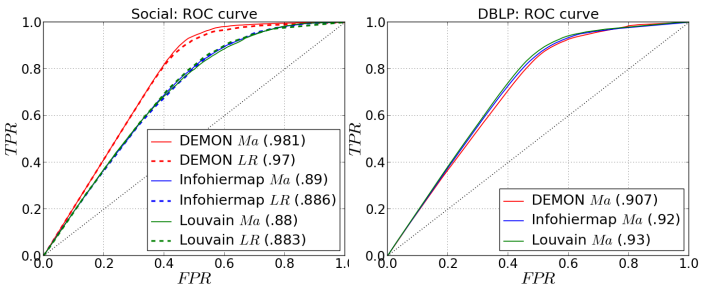

Fig. 1: Balanced Scenario: ROC curves.

Network	***DBLP***		***Social***	
Algorithm	**AUC**	**ACC**	**AUC**	**ACC**
DEMON Ma	0.907	85.58%	**0.981**	93.55%
DEMON LR	0.901	84.35%	0.970	91.87%
LOUVAIN Ma	**0.930**	87.72%	0.880	80.27%
LOUVAIN LR	0.926	87.48%	0.883	81.37%
INFOHIERMAP Ma	0.920	86.69%	0.890	81.34%
INFOHIERMAP LR	0.917	86.18%	0.886	80.89%

TABLE VI: Compared algorithms performances.

AUC identifies the area under the receiver operating characteristic (ROC). It illustrates the performances of binary classifiers relating the True Positive Rate $TPR = \frac{TP}{TP+FN}$ with the False Positive Rate $FPR = \frac{FP}{FP+TN}$ and providing a visual interpretation useful to compare different models. We carried out a preliminary study aimed at identifying the optimal window size n for the moving average (Ma) forecast having fixed the community discovery algorithm. By definition the Lv and Av are special cases of the more general Ma: particularly, the former is equivalent to Ma when $n = 1$ while the latter when $n = \tau$. By observing these results we realized that to obtain higher performances using Ma, two strategies are consistent: (i) minimize n using as forecast the last value (Lv) in order to make inference approximating the future with the actual network status, or (ii) use $n \simeq \tau$ in order to have a better estimation of the whole historical trends. Hereafter, we make use of the best scoring classifiers to detail our analysis. We will refer to them as the Ma models for each specific network and community definition.

As second step we compare the outcomes of the classifiers built using the LR forecast models with the Ma ones. In Fig. 1 are shown the ROC curves for both Social and DBLP datasets. In the former network we can observe how LR and Ma provide very similar results even if the moving average is always capable to obtain slightly better performances. DBLP shows the same trend with a small gap between the two approaches (for this reason we omit the LR curve). We report in Tab. VI the AUC and the ACC for all the comparisons.

In order to understand the boost provided to the classifier by the adoption of the right community discovery algorithm, we designed two different baselines: Structural Forecast (*SF*) and Filtered Structural Forecast (*FSF*). The SF model trains the classifier using only the forecasts for the pairwise structural features (CN, AA, PA and JC) computed on all the couple of nodes at distance at most 3 hops present in the whole network, not taking into account the presence/absence of shared communities among them. On the other hand, the FSF model restricts the computation to the pair of nodes belonging to the same community as the proposed approach does. As case study we report in Tab. VII AUC and ACC of the best Ma and LR baselines for the Social dataset.

Algorithm	Ma		LR	
	AUC	**ACC**	**AUC**	**ACC**
SF	0.901	82.88%	0.895	82.18%
FSF	**0.956**	90.10%	0.937	88.09%

TABLE VII: Balanced Scenario: baselines on Social using Structural Forecast (*SF*) and Filtered Structural Forecast (*FSF*).

Algorithm	*Structural*		*Topology*		*Community*	
	AUC	**ACC**	**AUC**	**ACC**	**AUC**	**ACC**
DEMON	0.957	90.59%	0.962	91.44%	0.903	83.53%
LOUVAIN	0.850	78.63%	0.875	79.38%	0.724	66.64%
INFOHIERMAP	0.876	79.85%	0.887	80.81%	0.667	62.11%

TABLE VIII: Balanced Scenario: compared classifier performances for different class of features on Social.

Since in Social our best performing approach is the one built upon DEMON communities, the structural features for the FSF baseline were computed using such partition of the network. The obtained results show that, using features extracted from the communities, we are able to gain 0.025 in AUC and 3.45% in ACC with respect to the FSF Ma baseline, and 0.08 in AUC and 10.67% in ACC with respect to the FS Ma one. These results highlight the importance of communities for the interaction prediction task, not only in providing features for pair of nodes, but also in filtering the dataset in order to determine a more accurate selection of nodes for the prediction. Without loss of generality, in the following of this section, in order to reduce the number of comparisons, we will report a full analysis only for the Social dataset. Furthermore, the results obtained for the DBLP scenario do not differ significantly from the ones discussed with the exception, as seen previously, of the best community discovery algorithm (Louvain instead of DEMON). This divergence is due to the different nature and topology of the networks analyzed.

Since our models are built upon three different classes of features (structural, topological and community related), it is mandatory to test their results against the classifiers that use separately each one of them. Such analysis allows us to assess the predictive power of each class of features, giving an idea of their overall importance for the complete model. We built a classifier for each community discovery algorithm and each feature class by using together all the forecasted versions of the features belonging to it. As shown in Tab. VIII, regardless the community discovery algorithm used, the most predictive features are the ones belonging to the *topology* class, followed by *structural* and *community* ones. However, we can observe how the AUC and ACC are always higher for the model based on DEMON: this trend suggests that such algorithm is the one that better bounds, at least for this network, the nodes that are more likely to establish future interactions.

As consequence to the good performances expressed by the analyzed classifier an interrogative arises. *Can results be improved by combining all the features obtained at the end of the forecasting stage?* We investigated if the performances of

Algorithm	AUC	ACC
DEMON *All*	**0.981**	93.90%
LOUVAIN *All*	0.901	83.05%
INFOHIERMAP *All*	0.894	81.91%
FS *All*	0.959	90.44%

TABLE IX: Balanced Scenario: compared classifier performances using all the features on Social dataset.

the analyzed classifiers can be improved by combining all the features obtained at the end of the forecasting stage (i.e. all the time series forecasts computed with Ma and LR). As we can see in Table IX, the performance boost is negligible with respect to DEMON Ma, in fact we are able to gain only 0.35% in ACC maintaining the same AUC w.r.t. the results shown in Table VI. This means that the feature set used by our best classifier is "stable": its extension do not produce advantages that justify an increase of model complexity. Conversely, for Louvain and Infohiermap the gain in AUC and ACC is more evident: this is due to the different degree of approximation introduced for each feature in the forecasting stage.

B. Unbalanced scenario

Unfortunately the balanced scenario is not common when addressing the Interaction Prediction problem. Social networks are generally sparse, and this led to an high rate of False Positive predictions (in case of unsupervised approaches) or to models that maintain high accuracy just predicting the absence of new links (the majority classifier in case of supervised learning). Indeed, predicting every object as belonging to the most frequent class, that is "no edge", guarantee high performances, but in general it leads to useless classification results. For this reason evaluating the performances of classifiers in highly unbalanced scenarios is not an easy, but highly important, task.

Since we want to predict correctly new links, our primary purpose is to reach high precision avoiding the generation of false positive predictions. This is why in the unbalanced scenario we will discuss the *Lift Chart* and *precision* of the tested classifiers.

Precision is defined as $PPV = \frac{TP}{TP+FP}$. It represents the ratio of correct predictions for a specific class (in our case the one representing the presence of the edge in the test set) with respect to the total predictions provided for it.

Lift Chart graphically represents the improvement that a mining model provides when compared against a random guess, and measures the change in terms of lift score.

We preserved the original ratio between the node pairs with and without a future interaction in Social and DBLP dataset. For both networks we used the DEMON algorithm to extract communities because (i) *Social*: DEMON reach the best performances in the balanced scenario, thus we expect that it will behave well even in unbalanced scenario; (ii) *DBLP*: using Louvain (i.e. the best performer in the balanced scenario) in the unbalanced scenario, all the classification models outputs the majority classifier.

In Social, the ratio of negative class over the total amount of possible pairs is 95.947%, that means that a majority classifiers predicting no edge for all the pairs would have an accuracy of almost 96%. As output from the classification phase with

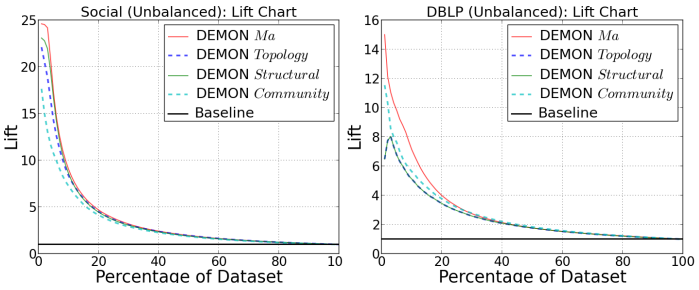

Fig. 2: Unbalanced Scenario: lift charts of the compared methods.

Algorithm	Ma		LR	
	AUC	PPV	AUC	PPV
SF	0.897	64.06%	0.893	62.62%
FSF	0.918	**74.71%**	**0.932**	72.45%

TABLE X: Unbalanced Scenario: baselines on Social using Structural Forecast (*SF*) and Filtered Structural Forecast (*FSF*).

Ma we have a model which reaches an AUC of 0.966 with a prediction accuracy of 98.75% and a precision w.r.t. the positive class of 95.61%. These two are very significant results: on one hand we have an accuracy improvement of 2.803% in an ideal window of 4.053% ($100\% - 95.947\%$) with respect to the majority classifier while, on the other, we have a very high precision on the positive class, considering that a classifier predicting always an edge would have a precision of 4.053%.

In Fig. 2-left we show the Lift Chart for Social. From the chart emerges that after the Ma model, the most promising is the one built upon the topological features followed by structural and community ones.

Also in this unbalanced scenario, we want to "measure" how much the community approach provide efficiency just filtering in the "promising pairs". By building the dataset with all possible pairs without make leverage of community information, we get a majority class, i.e.the absence of link, with a ratio of 98.96% over the total number of entries. In order to better compare the two cases, we filter out randomly some pair with no edge, bringing the accuracy of the majority classifier at 95.947% (like in the case with community discovery). Again we compare the performances for the SF and FSF, which are reported in Tab. X, but now considering the precision instead of the accuracy. We can see that we gain almost a 10% of precision just filtering out, in any time slot, all the pairs not belonging to the same community.

In DBLP case study, the resulting classifier has an AUC of 0.86, an ACC of 98.135% and a precision with respect to the positive class of 44.78%. The majority class (no link) has a ratio of 98.13% over all the instances of the dataset. We get a precision of 44.78%, starting from a ratio of positive class of 1.865% ($100\% - 98.135\%$), that is 24 times better than predicting for any pair the presence of the edge. Finally, we can observe from the Lift Chart in Fig. 2-right how, differently from the Social case, the most predictive set of features are the community ones, over the structural and topological.

V. Related Works

In literature there is a wide study of the link prediction problem. The methods used to solve LP apply supervised and/or unsupervised approaches [11]. The proposed method belongs to the category of methods which employ supervised machine learning techniques. LP through supervised learning algorithms was introduced in [13]. The authors studied the usefulness of graph topological features by testing them on co-authorship networks. A classifier is trained according to the knowledge that a link will be present or not in future. Then the classifier is used to predict new links.

After [13] a wide range of models exploiting several different strategies have been proposed. In order to build an efficient classifier many works focused on finding an efficient set of features. In [9] is shown that only small set of features are essential for predicting new edges and that contacts between nodes with high centrality are more predictable than nodes with low centrality. In [15] the authors rank the list of unlinked nodes according to topological measures, then each measure is weighted according to its performance in predicting new links. Finally in [21] tensor factorization is used to select the more predictive attributes. Like we did with community features, many works reinforce the classifier with other kind of knowledge. The authors of [19] used textual features besides the topological ones and apply SVM as supervised learning method. In [22] spatial and mobility information are used to help the classifier. Some works considering dynamic networks are [5] and [3]. In [5] association rules and frequent-pattern mining are used to search for typical patterns of structural changes in dynamic networks. In [3] the prediction is optimized through weights which are used in a linear combination of sixteen neighborhoods and node similarity features by applying the Covariance Matrix Adaptation Evolution Strategy. Finally, other works like [18], [7] show how an approach based on time series modeling the evolution of continue univariate features can help in solving the link prediction task. As shown in [11], despite the high precision, supervised approaches can be prohibitively time consuming for a large networks having over $10,000$ nodes. In order to reduce the computational complexity, several approaches such as [20] make use of clustering and community information. These analyses suggest that clustering information, no matter the algorithm used, improves link prediction accuracy.

VI. Conclusions

In this work we have tackled Interaction Prediction by designing a supervised learning strategy which uses time series forecasting models and community discovery. Time series forecasting models have been applied to analyze the evolution of link features. Community discovery algorithms allows the method to introduce community features which improve the performances and, at the same time, bound the training set by reducing the list of candidates. We have shown how this mining approach is able to achieve high performances both in balanced and unbalanced class distribution scenarios.

Acknowledgements. This work was partially funded by the European Community's H2020 Program under the funding scheme "FETPROACT-1-2014: Global Systems Science (GSS)", grant agreement #641191 CIMPLEX[4].

References

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

[2] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[3] C. A. Bliss, M. R. Frank, C. M. Danforth, and P. S. Dodds. An evolutionary algorithm approach to link prediction in dynamic social networks. *arXiv preprint arXiv:1304.6257*, 2013.

[4] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.

[5] B. Bringmann, M. Berlingerio, F. Bonchi, and A. Gionis. Learning and predicting the evolution of social networks. *IEEE Intelligent Systems*, 2010.

[6] M. Coscia, G. Rossetti, D. Pedreschi, and F. Giannotti. Demon: a local-first discovery method for overlapping communities. In *ACM SIGKDD*, 2012.

[7] P. R. da Silva Soares and R. Bastos Cavalcante Prudencio. Time series based link prediction. In *IEEE IJCNN*, 2012.

[8] X. Feng, J. Zhao, and K. Xu. Link prediction in complex networks: a clustering perspective. *The European Physical Journal B*, 2012.

[9] K. Jahanbakhsh, V. King, and G. C. Shoja. Predicting human contacts in mobile social networks using supervised learning. In *SIMPLEX workshop*. ACM, 2012.

[10] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *ACM SIGKDD*, 2010.

[11] L. Lü and T. Zhou. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, 2011.

[12] M. E. Newman. Clustering and preferential attachment in growing networks. *Physical Review E*, 2001.

[13] L. Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM*, 2003.

[14] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999.

[15] M. Pujari and R. Kanawati. Supervised rank aggregation approach for link prediction in complex networks. In *WWW*. ACM, 2012.

[16] M. Rosvall and C. T. Bergstrom. Multilevel compression of random walks on networks reveals hierarchical organization in large integrated systems. *PloS one*, 2011.

[17] G. Salton and M. J. McGill. Introduction to modern information retrieval. 1983.

[18] P. Sarkar, D. Chakrabarti, and M. Jordan. Nonparametric link prediction in dynamic networks. *arXiv preprint arXiv:1206.6394*, 2012.

[19] N. Shibata, Y. Kajikawa, and I. Sakata. Link prediction in citation networks. *JASIST*, 2012.

[20] S. Soundarajan and J. Hopcroft. Using community information to improve the precision of link prediction methods. In *WWW*. ACM, 2012.

[21] S. Spiegel, J. Clausen, S. Albayrak, and J. Kunegis. Link prediction on evolving data using tensor factorization. In *New Frontiers in Applied Data Mining*, pages 100–110. Springer, 2012.

[22] D. Wang, D. Pedreschi, C. Song, F. Giannotti, and A.-L. Barabasi. Human mobility, social ties, and link prediction. In *ACM SIGKDD*. ACM, 2011.

[23] Y. Xu and D. Rockmore. Feature selection for link prediction. In *CIKM workshop*. ACM, 2012.

[4] "Bringing CItizens, Models and Data together in Participatory, Interactive SociaL EXploratories'", https://www.cimplex-project.eu