

Exploring Supervised Methods for Temporal Link Prediction in Heterogeneous Social Networks

Nataliia Rümmele
Vienna University of
Technology, Austria
pobiedina@ec.tuwien.ac.at

Ryutaro Ichise
National Institute of
Informatics, Japan
ichise@nii.ac.jp

Hannes Werthner
Vienna University of
Technology, Austria
werthner@ec.tuwien.ac.at

ABSTRACT

In the link prediction problem, formulated as a binary classification problem, we want to classify each pair of disconnected nodes in the network whether they will be connected by a link in the future. We study link formation in social networks with two types of links over several time periods. To solve the link prediction problem, we follow the approach of counting 3-node graphlets and suggest three extensions to the original method. By performing experiments on two real-world social networks we show that the new methods have a predictive power, however, network evolution cannot be explained by one specific feature at all time points. We also observe that some network properties can point at features which are more effective for temporal link prediction.

Categories and Subject Descriptors

H.2.8 [Database Management]: Database Applications—*Data mining*

Keywords

temporal evolution, heterogeneous networks, link prediction

1. INTRODUCTION

An important problem in social networks, called link prediction, is to predict the appearance of a link between two disconnected people, or nodes, in the network. Nowadays link prediction has applications in many different domains, such as in communication or collaboration networks.

Earlier works on link prediction have focused on simple network models disregarding the temporal aspects and heterogeneity of human relationships [2, 13]. Temporal aspects have often been omitted due to insufficient data [6]. But nowadays, with the availability of Web APIs on many platforms, it is possible to collect data about social interactions over considerable time periods on a large scale. Recent works show that models which consider temporality better capture link formation processes in the network [12, 5, 19]. As

for the second property, heterogeneity can be frequently observed in real-world networks. For example, on Facebook we can study the formation of friendship ties and communication interactions between users. Classical methods for link prediction in homogenous networks cannot fully capture the complex structure in such scenarios. Hereby, the two easiest ways to adapt these methods to a heterogeneous setting are either to treat all link types equally or to consider only one link type and disregard the others. Either way we might lose valuable information which becomes even more critical in the light of sparse nature of many social networks.

In this paper we formulate link prediction as a binary classification problem, where a pair of nodes is classified as a positive case if they form a link. To efficiently combine information about various types of links in the network, we follow the approach of counting 3-node graphlets [6]. In this approach, we identify for a pair of nodes the type of a pattern/triad they form with their common neighbors and put a score proportional to the frequency of this triad in the whole network. We introduce three modifications to the original weighting scheme of triads. The first modification is the introduction of time awareness which we achieve by using the time score [12]. The second modification is motivated by the work in the area of graph pattern mining where the stress is put on the fact that simple counts are not always a proper measure to estimate the frequency of patterns in the network [4]. Our last modification addresses the hypothesis that triad formation among more active (thus experienced) and ordinary users in the network might be different. Therefore, we differentiate the 3-node graphlets not only by link type, but also by node categories. We also raise the question whether there are certain network properties which might point out a suitable weighting scheme of triads for temporal link prediction.

The main contributions of the paper are the following:

1. We study the performance of several supervised methods for temporal link prediction in heterogeneous social networks at several time points.
2. We suggest three modifications to the weighting scheme of 3-node graphlets [6].
3. Our experiments illustrate that network evolution cannot be explained by one specific feature at all time points which emphasizes the importance of combining different features into efficient models.
4. We observe that some network properties can point out which weighting scheme for 3-node graphlets is more effective for temporal link prediction.

The rest of the paper is structured as follows. In the next

section we provide a formal definition of the problem. In Section 3 we discuss related work. We explain the calculation of our new features in Section 4. In Section 5 we present datasets and experimental setting. We also report and discuss the results of our experiments. Finally, the conclusion is drawn and possible directions for future work are outlined.

2. PROBLEM

We have a network $G = (V, E_1, E_2, \tau_1, \tau_2)$, where V is the set of nodes; E_1 and E_2 are the set of links of type 1 and 2 respectively; τ_1 assigns timestamps to links of the first type and τ_2 to links of the second type. The set of nodes does not need necessarily to be homogeneous, but we focus on the case when nodes represent the same type of objects, namely people. Let $G[t] = (V, E_1[t], E_2[t], \tau_1, \tau_2)$ denote the sub-network of G which exists at time $t \in T$. In other words, this sub-network contains only those links which have a timestamp not greater than t : $E_1[t] = \{e \in E_1 : \tau_1(e) \leq t\}$, $E_2[t] = \{e \in E_2 : \tau_2(e) \leq t\}$.

The link prediction problem consists either of finding hidden connections or predicting links which will appear in the future based on the previously observed network states [9]. The first problem is better studied in the link prediction community. The latter, predicting future links, has gained more interest recently. We focus on this problem in the following formulation: Given $G[t] = (V, E_1[t], E_2[t], \tau_1, \tau_2)$ predict $E_x[t']$ for some $t' > t$ and link type $x \in \{1, 2\}$.

3. RELATED WORK

There are many methods developed for the link prediction problem in networks with one type of links. We can categorize them into three classes: approaches based on node features, probabilistic models and topological patterns [18]. We focus on the last category since it is related to our work. The aim here is to calculate a similarity score between a pair of nodes based on the topological metrics of the network. Among the topological metrics which are used, the neighborhood based scores form the biggest category. The *preferential attachment* (PA) score for a node pair (s, t) is the product of their degrees: $|N_s| * |N_t|$ [13]. Here, N_x denotes the set of immediate neighbors of the node x . The *common neighbors* (CN) counts the number of common neighbors: $|N_s \cap N_t|$ [13]. *Jaccard's coefficient* (JC) is a normalized number of common neighbors: $\frac{|N_s \cap N_t|}{|N_s \cup N_t|}$ [16]. The *Adamic/Adar* (AA) measure weights the impact of neighbors inversely according to their degrees: $\sum_{n \in N_s \cap N_t} \frac{1}{\log |N_n|}$ [2].

Unsupervised and supervised methods can be used to build a prediction model based on these scores. Lichtenwalter et al. show that supervised learning provides better results due to the ability to reduce variance and to cope with high imbalance in the class distribution [11]. To overcome imbalance, they suggest to use skew-insensitive trees based on Hellinger distance or to undersample the negative class.

We want to efficiently combine information about different types of links in the same network to predict future links. For this purpose Davis et al. construct a weighting scheme for combinations of link types based on the frequencies of 3-node graphlets [6]. In the following work Lichtenwalter and Chawla extend the class of graphlets and introduce a new method for link prediction, called Vertex Collocation Profiles (VCP) [10]. However, time information is not considered.

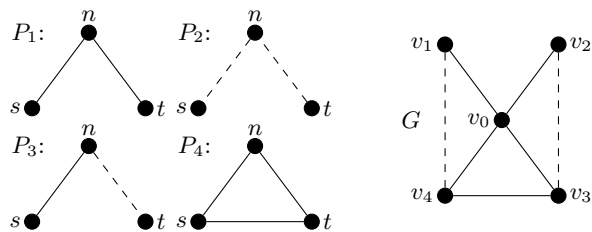


Figure 1: P_1 - P_4 are examples of 3-node graphlets for a network with links of two types; G is an example of a network with two types of links.

There are already works which incorporate time information into the calculation of similarity scores. For example, Munasinghe and Ichise suggest a time-score which combines the time and weight of links with common neighbors [12]. Their approach is based on two concepts: the strength of a link decreases with time, and the common neighbors are more effective if the interaction with them happens in a closer proximity of time. Soares and Prudencio do not construct one single score for a node of pairs, but rather explore the evolution of topological metrics by constructing a time series of scores [5]. Based on the obtained time series for a specific neighborhood based score, they predict the next value of the series and use the predicted value in the link prediction model. However, many works on temporal link prediction consider only homogeneous networks [5, 12].

We want to perform temporal link prediction in networks with two types of links. There is a recent work which solves this problem on a DBLP dataset by performing a multi-relational influence propagation [19]. To account for temporal dimension, the authors introduce several features. Firstly, they calculate recency as the time passed since a node made its last new link and activeness as the number of new links made in the last time step. These two features are somewhat similar to the time score [12]: instead of introducing time awareness for the links this work makes nodes time aware. Secondly, the authors calculate a degree preferential vector which can be perceived as a time series approach [5]. Finally, they design two new models (Temporal and MRIP models) and compare their performance against three state-of-the-art models (Homo Model, VCP3U and VCP4U). Our approach is based on the works of Davis et al. and Munasinghe et al. [6, 12]. Therefore, we use these works as baselines to compare our approach. We do not study the combinations of features. The future work includes this aspect and a comparison with the other models [19, 10].

4. TIME AND HETEROGENEITY BASED SCORES

Davis et al. introduce a prediction score for an appearance of a link between two nodes based on the frequency of 3-node graphlets which these two nodes form with their common neighbors [6]. We suggest to extend this methodology in three ways: (1) by introducing an additional weight based on time of interactions in the network; (2) by using support as a measure to account for frequency of graphlets; (3) by using node labels which indicate how experienced nodes are.

The original score which is introduced to predict a link of type x between nodes s and t is [6]: $score^{(i)}(s, t) =$

$\sum_{n \in N_s \cap N_t} w_n^{(i)}$, where $w_n^{(1)}$ corresponds to the weighting scheme from the previous work and $w_n^{(2)}$ is our modification. To calculate the first score, we put:

$$w_n^{(1)} = \frac{\sigma \cdot |P(x) - P(x \subset \text{edge_type}(s, t) | \text{pattern}(s, n, t))|}{P(\text{edge_type}(s, n)) \cdot P(\text{edge_type}(t, n))}$$

in which $\text{pattern}(s, n, t)$ describes one of possible 3-node graphlets formed by nodes s, n, t . Four out of 16 possible graphlets for a network with two types of links are illustrated in Figure 1. To calculate the conditional probability of the link x to appear in $\text{pattern}(s, n, t)$, we need to count the occurrence of $\text{pattern}(s, n, t)$ and the occurrence of this pattern with the added link x , and then divide the latter number by the former. $P(x)$ stands for the probability of the link type x in the network, and $\text{edge_type}(s, t)$ determines the link type between nodes s, t . N_s, N_t stand for the sets of neighbors for nodes s, t correspondingly. The value of $\sigma \in \{-1, 0, 1\}$ is determined upon statistical comparison of $P(x \subset \text{edge_type}(s, t) | \text{pattern}(s, n, t))$ and $P(x)$.

This approach is loosely related to the approach of mining graph evolution rules [3]. Unlike counting apriori known graphlets, Bringmann et al. discover frequent graph patterns in a network which are limited in their size. The frequency of the graph patterns is measured by the minimum image-based support. It roughly equals to the minimum number of different nodes in the network to which one of the nodes in the pattern can be matched. Bringmann and Siegfried argue that support is a better measure to estimate the frequency of a pattern than a simple count [4]. Consider the network G in Figure 1, where the first type of link is represented by the full line, and the second type is shown by the dashed line. 3-node graphlet P_1 has a count 16 in this network, and the corresponding closed triad P_4 has a count 6. The conditional probability of the appearance of the link of the first type between nodes v_1 and v_2 using counts [6]: $P(x \subset \text{edge_type}(v_1, v_2) | \text{pattern}(v_1, v_0, v_2)) = \frac{6}{16}$. The minimum image-based support for P_1 is three, since the node n can be matched only to v_0, v_3, v_4 in the network G . P_4 has the same support. Hence, the conditional probability of the appearance of the link of the first type between nodes v_1 and v_2 using support is 1. Note that such difference in these two calculation approaches arises if we have a high-degree node which connects otherwise disconnected nodes. Such kind of nodes are also referred to as structural holes. We do not know how exactly the temporal evolution is affected by structural holes in the network, but a previous work points out that structural holes can indicate the relationship type [17]. To account for such behavior in social networks, we calculate $w_n^{(2)}$ using the support [3]:

$$w_n^{(2)} = \frac{\text{support}^2(x \subset \text{edge_type}(s, t), \text{pattern}(s, n, t))}{\text{support}(\text{pattern}(s, n, t))}$$

This score has proven to perform well not only for link prediction in social networks [3], but also for prediction of citation counts over time [14].

To consider the temporal aspect of node interactions, we use the time score [12]: $TS = \sum_{n \in N_s \cap N_t} \frac{H_m * \beta^k}{|t_1 - t_2| + 1}$, where t_1, t_2 are recent interactions between n, s and n, t correspondingly, $k = \text{now} - \max(t_1, t_2)$, H_m is a harmonic mean of weights of these two interactions. We put β equal to 0.5. This score can be naturally applied to each 3-node graphlet

separately. Then we put a new time-dependent score:

$$tscore^{(i)}(s, t) = \sum_{n \in N_s \cap N_t} TS_n \cdot w_n^{(i)}.$$

The last extension is motivated by the work on link prediction across heterogeneous networks [8]. A more fine-grained mechanism to study network growth is proposed. First, nodes are categorized in two groups (elite and ordinary users) based on their PageRank score. Then triads (3-node graphlets) with respect to node categories are enumerated and weighted with regard to their frequencies. We also introduce two node categories, ordinary and experienced users, which we estimate based on user’s engagement in the network. We obtain a new score:

$$tscore_l^{(i)}(s, t) = \sum_{n \in N_s \cap N_t} TS_n \cdot W_n^{(i)}.$$

$W_n^{(i)}$ is calculated like $w_n^{(i)}$, except that patterns and link types include labels of the nodes. The new scores are similar to AA, since these features are based on the triad formation. However, they all provide different weighting scheme for a given triad. We hypothesize that our new weighting scheme will better capture the network evolution over time.

5. EXPERIMENTS

5.1 Datasets

We conduct experiments on two datasets: the *gaming network Dota2* and the *collaboration network HepTh*.

Gaming Network. The dataset was crawled using Steam API (friendship information) and Dota2 API (team membership information) from Valve. A detailed description of the dataset is available in our previous work [15]. Nodes correspond to users of Steam in both cases. The *mate* link indicates that two players were team mates in a Dota2 match, and the *friend* link means that two players are friends on Steam. The mate link has a weight as the number of matches where two players were team mates. We consider only links with weight > 1. We make this restriction for two reasons: firstly, if the weight is more than one, than the interaction is less likely to be random; secondly, to decrease the network volume. We introduce a weight on the friend links as the current friendship age. We select one calendar week as the time unit. The users in this network are classified at each time point in two groups based on their experience: if they played more than 60 matches, they are marked as “experienced”, otherwise “ordinary”. The threshold is identified by the distribution of users’ experience in the week 201145.

Collaboration network. The collaboration network is constructed from 27,732 papers from the years 1992 – 2003 from the arXiv which are categorized as High Energy Physics Theory. Nodes correspond to authors of these publications. There are *colleague* and *peer* types of links. If a scientist collaborates with another one on some paper, it indicates that they are colleagues and know each other on a personal level. If a scientist cites one of the works of the other author, it indicates that both of them are working on related topics, and, thus, they may be considered as peers. That is also the reason why we ignore the direction of this relationship. We delete self-loops. There are weights on both types of links which indicate how many times the corresponding interaction between the authors takes place. We select one

Table 1: Properties of Dota2 network over time.

year-week	<i>team mate link</i>					<i>friend link</i>				
	# nodes	# links	\bar{C}	T	# CC	# nodes	# links	\bar{C}	T	# CC
201147	7901	17544	0.216	0.112	584	69505	81733	0.058	0.021	1831
201149	18763	47145	0.214	0.084	668	75756	92656	0.069	0.026	1786
201151	28639	81237	0.208	0.069	526	81858	103983	0.079	0.03	1602
201201	34839	114426	0.205	0.061	271	86272	112574	0.083	0.033	1432
201203	40104	176195	0.196	0.063	88	90010	121424	0.087	0.035	1214
201205	43701	232289	0.192	0.07	48	93295	129767	0.092	0.038	1043
201207	46645	276768	0.19	0.076	34	96534	138030	0.095	0.039	903

Table 2: Properties of HepTh network over time.

year	<i>peer link</i>					<i>colleague link</i>				
	# nodes	# links	\bar{C}	T	# CC	# nodes	# links	\bar{C}	T	# CC
1993	529	1618	0.435	0.314	30	965	1185	0.521	0.592	217
1995	2500	21845	0.461	0.245	23	2579	3988	0.482	0.409	289
1997	4063	68510	0.471	0.248	14	3902	7173	0.472	0.313	318
1999	5325	140528	0.477	0.259	16	5025	10622	0.463	0.263	327
2001	6677	241091	0.485	0.252	16	6240	14534	0.466	0.229	349
2003	7978	368323	0.493	0.25	15	7416	18793	0.467	0.206	349

year as the time unit. The authors are classified at each time point in two groups based on how many papers they wrote before: if they authored more than 10 papers, they are marked as “experienced”. The threshold is identified by the distribution of users’ experience in the year 1993.

We present the properties for each link type in Tables 1 and 2 over the selected time periods. These properties include number of nodes (#nodes), number of links (#links), average clustering coefficient (\bar{C}), transitivity (T) and number of connected components (#CC). For simplicity we show every second time point to get the feeling about the trends.

5.2 Experimental setting

We select several state-of-the-art scores as our baseline: AA, CN, JC and PA which are described in Section 3. We also consider TS [12] and $score^{(1)}$ [6]. We predict each link type separately, thus features AA, CN, JC, PA and TS are calculated for the network with one link type. We construct the training and testing datasets based on link existence at different time units: t and $t+1$ correspondingly. If a link appears between two nodes at time t , then it is a positive class in the training dataset. The same rule is used for the testing dataset, except that we consider time $t+1$. The features are calculated based on $G[t-1]$ for the training dataset and $G[t]$ for the testing dataset. We predict links for the testing dataset at time $t+1$. We undersample the negative class to balance the class distributions [11, 6, 5, 10]. We choose to undersample the negative class so that the positive class represents 25% observations [6, 10]. We consider only those pairs of nodes which have common neighbors in the network.

We incorporate bagging into our model [6]. We build ten folds for the training datasets: each fold contains all observations of the positive class, but we take different observations of the negative class to reach the balanced ratio of classes; we use the same testing dataset for each fold of the training dataset. We experimented with three classification methods: logistic regression (LR), conditional inference trees (CIT) and random forests (RF). Due to consistently better performance of CIT, we report the results of this method.

Recent works indicate that in the case of a highly imbalanced class distribution Area under Precision-Recall curve (AUPR) is a better performance measure [7, 20, 6, 10]. We use R packages in our learning tasks: party and PerfMeas [1].

5.3 Results

We report AUPR results in Tables 3-4 for Dota2 network

and in Tables 5-6 for HepTh network. The best performing feature for each time period in the network (i.e., each row in the tables) is marked in bold. We noticed that, in case AUPR is more than 0.99, the learning algorithm fails to create a classification model (AUROC equals 0.5 in such cases). Closer analysis reveals that in such cases the distributions of the calculated scores are almost identical for positive and negative classes, resulting in all instances being classified into the negative class. Thus, we mark the corresponding numbers in the tables in italic and identify the second best results. Our general expectation is that the new time-dependent scores $tscore^{(i)}$ and $tscore_l^{(i)}$ will perform better than $score^{(i)}$. We do not expect $score^{(2)}$ to outperform $score^{(1)}$, but we want to define the cases when it does. Lastly, we expect that including additional information about users’ experience will improve the performance, i.e., $tscore_l^{(i)}$ is better than $tscore^{(i)}$.

The results indicate that there is no definite winner in all cases. The performance of features varies over time and across link types. However, if we compare the average of AUPR over the considered time periods, we obtain that $tscore_l^{(1)}$ provides the highest value for 3 out of 4 link types (we exclude cases with AUPR>0.99 from averaging). Prediction of friend links in Dota2 network is the only case when JC yields better average results. It is also the case when the new scores based on the support calculation outperform almost at all time points the scores which use counts for graphlets. In Table 1 we notice that the network based on friend links has very low average clustering coefficient and transitivity, especially compared to three other networks. These two network measures provide an insight how well the social balance theory is fulfilled within the network. Apparently, such configuration leads to the better performance of JC and PA which are not based on this theory. Furthermore, it also provides an explanation why using support for frequency calculation is a better choice in this situation. Remember the example network G in Figure 1 where the conditional probability of link appearance is higher if we use the support measure. Similar situation will arise every time we have structural holes in the network. Tang et al. include the information about structural holes for the task of link prediction across networks [17]. Our results indicate that such information might further improve the link prediction and help to choose the suitable weighting scheme.

We make another interesting observation for the network

Table 3: AUPR for predicting mate links.

week	AA	CN	JC	PA	TS	$score^{(1)}$	$score^{(2)}$	$tscore^{(1)}$	$tscore^{(2)}$	$tscore_i^{(1)}$	$tscore_i^{(2)}$
201148	0.192	0.21	0.999	0.489	0.201	0.141	0.126	0.113	0.169	0.117	0.191
201149	0.123	0.184	0.219	0.181	0.142	0.134	0.431	0.113	0.288	0.119	0.121
201150	0.082	0.092	0.168	0.08	0.146	0.095	0.129	0.087	0.092	0.094	0.143
201151	0.093	0.114	0.096	0.12	0.142	0.087	0.105	0.101	0.204	0.094	0.185
201152	0.083	0.129	0.828	0.121	0.128	0.103	0.085	0.999	0.108	0.999	0.087
201201	0.102	0.112	0.177	0.096	0.122	0.08	0.12	0.215	0.17	0.216	0.215
201202	0.074	0.079	0.109	0.076	0.19	0.11	0.081	0.122	0.082	0.19	0.095
201203	0.095	0.095	0.176	0.084	0.107	0.119	0.09	0.163	0.083	0.368	0.094
201204	0.078	0.09	0.069	0.102	0.088	0.112	0.08	0.237	0.108	0.336	0.112
201205	0.078	0.084	0.084	0.082	0.076	0.093	0.08	0.121	0.082	0.361	0.08
201206	0.067	0.07	0.117	0.084	0.123	0.068	0.072	0.18	0.07	0.338	0.066
Average	0.097	0.115	0.204	0.138	0.133	0.104	0.127	0.145	0.132	0.223	0.126

Table 4: AUPR for predicting friend links.

week	AA	CN	JC	PA	TS	$score^{(1)}$	$score^{(2)}$	$tscore^{(1)}$	$tscore^{(2)}$	$tscore_i^{(1)}$	$tscore_i^{(2)}$
201148	0.276	0.496	0.797	0.503	0.27	0.26	0.346	0.465	0.398	0.389	0.385
201149	0.249	0.343	0.941	0.594	0.195	0.35	0.154	0.379	0.711	0.329	0.259
201150	0.203	0.182	0.694	0.604	0.14	0.244	0.339	0.307	0.799	0.325	0.5
201151	0.546	0.524	0.323	0.649	0.212	0.276	0.115	0.363	0.969	0.999	0.817
201152	0.385	0.383	0.831	0.767	0.626	0.274	0.74	0.359	0.956	0.174	0.949
201201	0.195	0.284	0.953	0.915	0.458	0.195	0.07	0.999	0.939	0.4	0.943
201202	0.355	0.204	0.923	0.616	0.924	0.999	0.622	0.199	0.942	0.19	0.769
201203	0.179	0.203	0.601	0.799	0.258	0.214	0.237	0.173	0.962	0.273	0.215
201204	0.152	0.226	0.956	0.444	0.318	0.152	0.615	0.178	0.379	0.264	0.504
201205	0.23	0.249	0.57	0.361	0.385	0.195	0.653	0.228	0.251	0.192	0.343
201206	0.194	0.202	0.681	0.667	0.181	0.183	0.697	0.227	0.262	0.282	0.17
Average	0.27	0.3	0.752	0.629	0.361	0.234	0.417	0.288	0.688	0.282	0.532

based on colleague links. This is the only case where the number of connected components grows over time (Table 2). It might be explained by the trend of writing papers within specified groups at hosting institutes and of having little collaboration with outside groups. Though the features based on our new scores perform overall good, there is no consistency in results with $tscore_i^{(1)}$ leading often to extreme AUPR values. Including more information about authors, e.g., which institute they belong to, might improve the performance like it was done in some previous works [19].

Overall, introducing the time awareness improves the performance. However, not in all cases $tscore^{(i)}$ yields better AUPR compared to $score^{(i)}$, especially on the network with colleague links. Still if we use 3-node graphlets which distinguish between ordinary and experienced authors, we gain advantage over $score^{(i)}$. Nevertheless, in case of the Dota2 network with friend links $tscore_i^{(i)}$ reduces the average AUPR in comparison to $tscore^{(i)}$. We think that the gaming experience of users does not really impact the friendship formation among them. It might be more effective to categorize users in this network with regard to being a structural hole. Judging from results both for Dota2 and HepTh networks, we believe that the choice of node categories fits well team mate and peer link types, but we could introduce better node categories for friend and colleague link types, thus reducing the inconsistency.

Friendship and colleague networks are sparse compared to team mate and peer networks. We see that the new score performs well on the latter while the results for the former are inconsistent. Previous works outline that a technique based on counting graphlets is better suited for information networks [6]. It might be the explanation for such inconsistency, however, we show that even in such cases the new scores perform on average well and might further improve supervised models in combination with other features.

Our results lead us to the conclusion that the network evolution is more complex and is not completely captured by one specific feature. Even within the same type of net-

work we observe quite a lot of variance in the performance of features. Similar observation has been already outlined by Davis et al. [6]. However, we want to stress that even combining efficiently the considered features on one dataset at a specific time point does not guarantee that this model will perform equally well over time. It is worth noting that except for the network structure and content we do not use any additional information. However, there can be outside factors which influence the network evolution. For example, Valve (the company which develops Dota2 and Steam) performed many marketing activities to attract new players to their game Dota2 in 2011. Note that we have currently no means to include them into our model. This fact might explain why till the beginning of 2012 we observe quite drastic changes in the performance of features (see Table 3).

6. CONCLUSION

We have studied the performance of several state-of-the-art and 5 new scores for temporal link prediction in social networks with two types of links. We have performed experiments on two real-world social networks: a gaming network Dota2 with team mate and friend links; a co-author network HepTh with colleague and peer links. We have confirmed once again that considering the temporal aspect of links when studying network evolution is important and leads to the improved link prediction performance. However, our results indicate that the performance of link prediction methods varies over time, and in two cases there is considerable inconsistency in results within the same network type.

We have noticed that the methods which use some variation of a triad counting technique do not perform well on the network with very low average clustering and transitivity coefficients. By using a support measure to estimate the frequencies of graphlets we achieve better performance for temporal link prediction.

The results of our work point out several directions for the future work. First of all, the categorization of nodes according to structural hole spanning could lead to the fur-

Table 5: AUPR for predicting peer links.

year	AA	CN	JC	PA	TS	$score^{(1)}$	$score^{(2)}$	$tscore^{(1)}$	$tscore^{(2)}$	$tscore_j^{(1)}$	$tscore_j^{(2)}$
1994	0.202	0.225	0.095	0.168	0.167	0.13	0.325	0.265	0.195	0.336	0.131
1995	0.223	0.23	0.166	0.18	0.196	0.163	0.327	0.183	0.27	0.423	0.109
1996	0.208	0.21	0.116	0.154	0.212	0.169	0.239	0.241	0.242	0.418	0.219
1997	0.203	0.19	0.221	0.141	0.181	0.171	0.193	0.242	0.193	0.715	0.235
1998	0.182	0.183	0.168	0.143	0.196	0.182	0.185	0.202	0.184	0.328	0.19
1999	0.198	0.195	0.179	0.155	0.194	0.184	0.165	0.214	0.197	0.342	0.203
2000	0.192	0.194	0.217	0.154	0.184	0.186	0.172	0.257	0.199	0.249	0.205
2001	0.243	0.242	0.227	0.18	0.234	0.245	0.22	0.332	0.236	0.718	0.244
Average	0.206	0.209	0.174	0.159	0.195	0.179	0.228	0.242	0.215	0.441	0.192

Table 6: AUPR for predicting colleague links.

year	AA	CN	JC	PA	TS	$score^{(1)}$	$score^{(2)}$	$tscore^{(1)}$	$tscore^{(2)}$	$tscore_j^{(1)}$	$tscore_j^{(2)}$
1994	0.161	0.356	0.146	<i>0.991</i>	0.287	0.813	0.579	0.566	0.546	0.359	0.772
1995	0.154	0.195	<i>0.992</i>	0.224	0.254	0.805	0.641	0.86	0.573	<i>0.993</i>	0.835
1996	0.296	0.37	0.261	0.226	0.541	0.562	0.289	0.591	0.294	<i>0.995</i>	0.648
1997	0.276	0.214	0.148	0.164	0.216	0.468	0.311	0.215	0.298	<i>0.996</i>	0.358
1998	0.182	0.136	0.138	0.186	0.109	0.4	0.214	0.138	0.321	0.769	0.31
1999	0.385	0.199	0.123	0.138	0.145	0.271	0.123	0.145	0.189	0.585	0.204
2000	0.376	0.186	0.692	<i>0.997</i>	0.136	0.11	0.135	0.1	0.158	<i>0.997</i>	0.127
2001	0.116	0.223	0.166	<i>0.997</i>	0.122	0.306	0.792	0.17	0.085	<i>0.997</i>	0.086
Average	0.243	0.235	0.239	0.188	0.226	0.467	0.385	0.348	0.308	0.571	0.418

ther improvement in performance of our new scores $tscore_l^{(i)}$. Secondly, we could introduce time series techniques [19, 5]. Lastly, by efficiently combining the new features we could compare their performance with the state-of-the-art models like Homo, MRIP and VCP [10, 19].

There is currently a trend to develop supervised models for link prediction by combining classical unsupervised scores (e.g., AA, JC, PA, CN) with new features [10, 19]. These models are then tested on a variety of datasets with the goal to fit as many as possible. We noticed that there might be innate network properties (e.g, average clustering coefficient, presence of structural holes) which could point out appropriate features to explain its future evolution. Instead of designing a general model to fit different networks, we could guide the process of model development with regard to these properties. For example, the methodology based on 3-node graphlets, which we applied in our current work, and more generally the approach of VCPs [10] provide considerable flexibility as to how much additional information (besides the local structure among nodes) is captured by the graphlets, like node labels, link labels, frequency estimation and weighting scheme. However, to tune the parameter configuration, we need a bigger pool of networks.

7. REFERENCES

- [1] The R project for statistical computing. <http://www.r-project.org/>, January 2013.
- [2] L. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, 25:211–230, 2001.
- [3] B. Bringmann, M. Berlingerio, F. Bonchi, and A. Gionis. Learning and predicting the evolution of social networks. *IEEE Intelligent Systems*, 25:26–35, 2010.
- [4] B. Bringmann and S. Nijssen. What is frequent in a single graph? In *Proc. PAKDD*, pages 858–863, 2008.
- [5] P. da Silva Soares and R. Bastos Cavalcante Prudencio. Time series based link prediction. In *Proc. IJCNN*, pages 1–7, June 2012.
- [6] D. Davis, R. Lichtenwalter, and N. Chawla. Supervised methods for multi-relational link prediction. *Social Network Analysis and Mining*, 3(2):127–141, 2013.
- [7] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *Proc. ICML*, pages 233–240. ACM, 2006.
- [8] Y. Dong, J. Tang, S. Wu, J. Tian, N. Chawla, J. Rao, and H. Cao. Link prediction and recommendation across heterogeneous social networks. In *Proc. ICDM*, pages 181–190, Dec 2012.
- [9] L. Getoor and C. P. Diehl. Link mining: A survey. *SIGKDD Explor. Newsl.*, 7(2):3–12, Dec. 2005.
- [10] R. N. Lichtenwalter and N. V. Chawla. Vertex collocation profiles: Subgraph counting for link analysis and prediction. In *Proc. WWW*, pages 1019–1028, 2012.
- [11] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *Proc. KDD*, pages 243–252, New York, NY, USA, 2010. ACM.
- [12] L. Munasinghe and R. Ichise. Time score: A new feature for link prediction in social networks. *IEICE Trans.*, 95-D(3):821–828, 2012.
- [13] M. Newman. Clustering and preferential attachment in growing networks. *Phys. Rev. E* 64, 2:025102, 2001.
- [14] N. Pobiedina and R. Ichise. Predicting citation counts for academic literature using graph pattern mining. In *IEA-AIE*, pages 109–119, 2014.
- [15] N. Pobiedina, J. Neidhardt, M. del Carmen Calatrava Moreno, and H. Werthner. Ranking factors of team success. In *WWW (Companion Volume)*, pages 1185–1194, 2013.
- [16] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. New York, NY, USA, 1986.
- [17] J. Tang, T. Lou, and J. Kleinberg. Inferring social ties across heterogeneous networks. In *Proc. WSDM*, pages 743–752, 2012.
- [18] E. W. Xiang. A Survey on Link Prediction Models for Social Network Data. Technical report, The Hong Kong University of Science and Technology, 2008.
- [19] Y. Yang, N. V. Chawla, Y. Sun, and J. Han? Predicting links in multi-relational and heterogeneous networks. In *Proc. ICDM*, 2012.
- [20] Y. Yang, R. Lichtenwalter, and N. Chawla. Evaluating link prediction methods. *Knowledge and Information Systems*, pages 1–32, 2014.