A decision-making based feature for link prediction in signed social networks

Conference Paper · November 2013

DOI: 10.1109/RIVF.2013.6719888

3 authors, including:

Hung Thanh Vu
Deakin University
12 PUBLICATIONS **73** CITATIONS
SEE PROFILE

Bac Le
Ho Chi Minh City University of Science
63 PUBLICATIONS **306** CITATIONS
SEE PROFILE

A Decision-making based Feature for Link Prediction in Signed Social Networks

Tuyen-Thanh-Thi Ho
HCMUS
Ho Chi Minh City, Vietnam
htttuyen@fit.hcmus.edu.vn

Hung Thanh Vu
HCMUS
Ho Chi Minh City, Vietnam
vthung@fit.hcmus.edu.vn

Bac Hoai Le
HCMUS
Ho Chi Minh City, Vietnam
lhbac@fit.hcmus.edu.vn

***Abstract* — We are interested in signed link prediction where relationships should be predicted as positive (friendship, fan, like, etc) or negative (opposition, anti-fan, dislike, etc). In this problem, feature extraction is an essential step to encode the information needed for prediction. While most current features are based on balance or status theory, we consider the problem of link prediction in the different view of decision-making theory. Our main contribution is a novel feature called Positive-Negative Ratio feature (PNR) which is the ratio between positive and negative links. Our PNR feature, which is based on the strong theory of decision-making, reveals many advantages compared to existing features. It uses a two-dimensional feature but can defeat existing methods at least 3% in classification accuracy and AUC in all three standard databases (Epinions, Slashdot and Wikipedia), even training and testing databases are different. Furthermore, PNR is 5, 1.3 and 1.5 times faster than the other methods in extraction, training and prediction steps, respectively.**

Keywords-component: Link Prediction, Signed Social Network, Decision-making Theory.

I. INTRODUCTION

As well as the development of the Internet, social network becomes a hot topic and draw attention of research community. Besides the explicit information provided by users such as profiles, interests, families or companies, the social network includes complicated connections between people in real life. If we can understand the trend of link establishment inside a social network, a wide range of applications can be obtain such as: recommender systems, advance search engines, online marketing, criminal investigation, social network trend prediction [6, 12]. For this reason, link prediction becomes a core problem in the area of social network analysis. However, social networks are dynamic objects which frequently change under the daily activities of users. Furthermore, networks always have complicated structures and large sizes with millions of nodes and links. Therefore, link prediction is still a challenging problem for researchers at present.

Given a social network, link prediction will determine whether a relationship between two users or not, and what kinds of relationship from available information in the network. The predicted relationship can be a link occurring in the future or a missing link. There are many studies of link prediction [5, 11, 13] but most of them consider only positive relationships in networks and ignore negative ones. Meanwhile, people in human society have both positive and negative relationships. Furthermore, negative links also play an important role (even more important than positive links) in applications such as criminal investigation or user's feedback on products. Hence, if information of negative links is integrated, we can improve the accuracy of link prediction systems significantly. Social networks that contain both positive and negative relationships are called signed social networks. One of the first studies of signed social networks is based on status theory and was published by Guha et al. [10] in 2004. They developed a belief propagation model to predict the links between nodes in networks. The propagation is done by computing belief matrices using operations of atomic propagation to measure the probability of belief/unbelief between nodes. Kunegis et al. [14] extended the method of graph spectrum analysis by using kernels taken from signed Laplacian matrices on graphs.

Recently, Leskovec [7, 8] have used another approach. Unlike the two mentioned methods, Leskovec focused more on features than algorithms. Particularly, in their papers, they proposed 16-dimensional features corresponding to 16 types of triads in the balance theory and used logistic regression model to predict the sign of links. Another study tried to improve the quality of feature representation is the paper of Kai-Yang et al in [9]. Instead of using triads, the authors took all cycles of m vertices containing predicted links. Obviously, the complexity of feature vectors will increase with the number of cycles (when m is large). Therefore, in their paper, they observed only the cycles with 5 vertices ($m = 5$).

In this paper, we define the problem of signed link prediction as decision-making in which link signs are considered at the time of link creation: Why a user decide to create a positive, not negative, link and which factors affect his decision. We propose a novel feature called Positive-Negative Ratio feature (PNR) based on decision-making theory. The difference between PNR and existing features (based on balance theory and status theory) is that the PNR feature uses only the local information at nodes of a link while the other features require information of shared nodes (triads or cycles in networks). Since triads or cycles occupy a small proportion of networks, the accuracy of balance and status theories-based features are low. Besides high accuracy, PNR has other benefits: low-cost feature, simple implementation but high generalization and high speed. Our prediction system starts with extracting PNR features at nodes in networks. They are the ratios between positive and negative signs of incoming and outgoing edges. Next, a regression logistic model is applied to learn a weight vector which will be used in the prediction phase. Experiments in three common databases including Epinions, Slashdot and Wikipedia [15] show that the PNR feature outperforms most state-of-the-art features in three criteria: accuracy, generalization and speed.

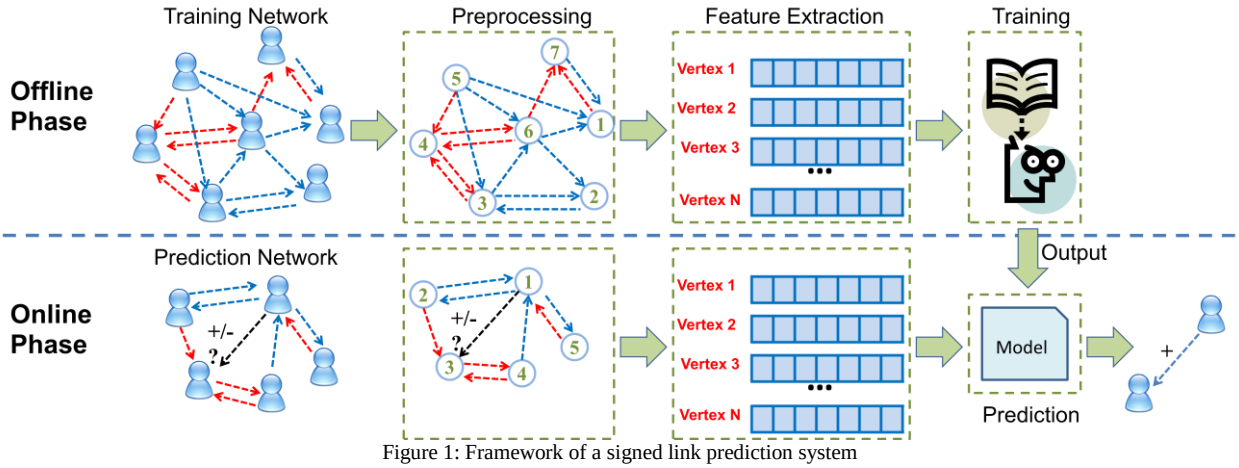

Figure 1: Framework of a signed link prediction system

II. SIGNED LINK PREDICTION

A. Problem Statement

We model social networks as graphs. Given a direct acyclic graph $G=(V,E,\Sigma)$, here $V=\{1,2,...,n\}$ is vertices and E is a set of edges $(u,v)\in V\times V$. Since G is a direct graph, we denote (u,v) as an edge from u to v. The function $\Sigma: E\rightarrow\{+1,-1\}$ maps an edge (u,v) in E to a set of signs: +1 denotes a positive link and -1 is a negative link. Signed link prediction can be stated that: given a full network with only one edge whose sign is unknown or missing, the prediction system uses the rest of the network to determine the edge to be positive or negative.

B. Framework Overview

A signed link prediction system works through two phases: the offline and the online phases. The overview of the whole framework is given in the Figure 1. In general, both phases share the steps of preprocessing and feature extraction. In this section, we will go briefly through steps in the framework.

Offline phase:

The purpose of this phase is to learn a prediction model from training data. The phase is divided into three steps: preprocessing, feature extraction and training.

+ **Preprocessing: S**ocial networks are represented as graphs in this step. Objects (individuals, products, organizations, etc) in networks are considered as nodes while relationships are edges in graphs. Friendships or antagonism can be encoded into graphs as the signs (+/-) of edges. The benefit of graph representation is that it demonstrates most information of social networks and is easy for visualization.

+ **Feature Extraction:** Since not all information inside networks is useful, feature extraction is needed to remove redundant data and encode the essential information of edges into feature vectors for training steps. Since the result of training step is affected by training data, the quality of feature vectors plays an important role in the framework.

+ **Learning methodology**: Training data provided by feature extraction step is the input for learning algorithms. Each feature vector is labeled as positive or negative. Feature vectors and their labels are used to train a prediction model.

Online phase:

The online phase begins with the steps of preprocessing and feature extraction similar to the offline phase. Next, the prediction model (trained in the offline phase) is used to assign an edge to positive or negative. It is noted that we only need to train the prediction model once in the offline phase and use it many times in the online phase without retraining it.

C. Existing theories

We begin by summarizing two main theories of balance theory [4, 7, 8] and status theory [8, 10] which are used widely in signed link prediction.

The first is structural balance theory. It is well-studied and based on simple principles "the friend of my friend is a friend" or "the enemy of my friend is my enemy" or "the friend of my enemy is my enemy". Suppose that three nodes *u*, *v* and *w* with links between them establish a triad. The structural balance theory states that the number of positive signs should be odd (one or three in this case).

Unlike balance theory, status theory assumes that there exists an ordering in the networks. In this theory, a positive link (*u*, *v*) means *u* makes a relationship with *v* since *v* has a higher status than *u*. Otherwise, a negative link (*u*, *v*) means *v* has lower status than *u*. Therefore, a positive link from *u* to *v* is similar a negative link from *v* to *u*. The flip of a link direction causes the flip in the link sign. After flipping the sign of directions so that edges point from *u* to *w* and from *w* to *v*, the sign of (*u*, *v*) is the summation of signs of (*u*, *w*) and (*w*, *v*).

Obviously both balance and status theories rely on triads in networks for link prediction. This means that they only predict effectively links in triads in networks. However, the proportion of triads in real networks are inconsiderable and therefore, the effects of the methods will be limited.

D. Proposed feature

Unlike two above approaches, we view link prediction as the voting problem of *u* for *v*. By this way, the sign of the link indicates the attitude of *u* toward *v*. A positive (negative) sign is produced if *u* votes for (against) *v*. Since the signs are formed when users show up their positive or negative attitude, we consider the signed link prediction as an act of voting

(like/dislike, agree/disagree, etc) of a user *u* to an object *v*. Furthermore, since this act is deeply related to human behaviours and psychology at the time he votes, we believe that the user's decision-making is essence of signed link prediction. The idea motivates us to investigate the theory of decision-making and introduce an effective feature with low computational cost using the theory.

1. Positive-Negative Ratio Feature (PNR)

Given *u* is a node in a network. Suppose that $d_{in}^{+}(u)$ denotes the number of positive incoming edges at *u* while $d_{in}^{-}(u)$ is the number of negative incoming edges at u. Similarly, $d_{out}^{+}(u)$ and $d_{out}^{-}(u)$ are the number of positive and negative outgoing edges of *u*. The four values $d_{in}^{+}(u), d_{in}^{-}(u), d_{out}^{+}(v), d_{out}^{-}(v)$ are called as degree features [7]. An n example of degree features is given in Figure 2. Given a link (*u*, *v*) from *u* to *v*, we are interested in the ratios between four terms at *u* and *v*:

$$R_{out}(u) = \frac{d_{out}^{+}(u)}{d_{out}^{-}(u) + \varepsilon} \quad (1)$$

$$R_{in}(v) = \frac{d_{in}^{+}(v)}{d_{in}^{-}(v) + \varepsilon} \quad (2)$$

where, $R_{out}(u)$ is the proportion between positive and negative outgoing edges at *u* while $R_{in}(v)$ is the proportion between positive and negative incoming edges at *v*. In the relationship between *u* and *v*, *u* is voter (giving a vote) while *v* receives the vote. Therefore, we only take $R_{out}(u)$ and $R_{in}(v)$ into account and exclude the ratios of $R_{in}(u)$ and $R_{out}(v)$. The term of ε is an extremely small value to ensure that the denominators are nonzero. Additionally, since R_{in} and R_{out} may reach positive infinity; their wide value ranges cause difficulties in the learning step. To overcome this, a threshold *t* is used to cut the ranges down.

$$R_{out}^{t}(u) = \min(R_{out}(u), t) \quad (3)$$

$$R_{in}^{t}(v) = \min(R_{in}(v), t) \quad (4)$$

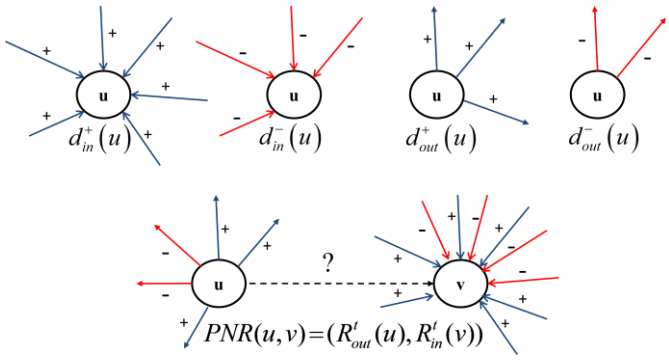

Figure 2: An example of $d_{in}^{+}(u),\ d_{in}^{-}(u), d_{out}^{+}(u), d_{out}^{-}(u)$ and PNR feature. In this example, $d_{in}^{+}(u) = 6,\ d_{in}^{-}(u) = 4,\ d_{out}^{+}(u) = 3,\ d_{out}^{-}(u) = 2$. Best view in color.

The balance ratio feature (PNR) of the edge *(u, v)* is given by concatenating two limited ratios (Figure 2):

$$PNR(u,v) = (R_{out}^{t}(u), R_{in}^{t}(v)) \quad (5)$$

Most computational time of PNR is to count degree features of nodes. However, these values can be computed by scanning the network and calculating in advance for each node in the offline phase. Moreover, since PNR use only two positive real values to encode the information of an edge, it will have more benefits in computation in the further steps of training and prediction. Consequently, the speed of the whole system will be reduced significantly. It is obvious that our proposed feature is absolutely low-cost as well as easy to implement. Due to two advantages, the PNR feature can work effectively in large-scale social networks in practice.

For generalization and accuracy, our method considers the information at two nodes of links, does not use the shared nodes in triads like balance and status theories. Essentially, decision-making theory focuses on individuals or objects themselves while balance and status theories are based on relationships with third parties. Hence, the PNR feature can handle any link in networks and therefore predict more correct link than the other theories. The theoretical foundation of PNR will be introduced in the next section.

2. Connections to decision-making theory

In this section, we show that our simple PNR feature implies decision making principles and therefore, it can yield high accuracy in prediction.

Past experience: One of the most important factors affecting decision makers is past experience. Results reported by Juliusson et al. in [1] indicate the influence of past decisions on future decisions. The reason is that when several decisions leads to more positive results, people tend to repeat their behaviours (here are decisions) and then repetitive behaviours cause a habit. An example is that a person usually gives more positive votes than negative votes. It becomes his habit that he tends to give positive votes - he can have a negative attitude toward something but he will ignore voting because he is familiar with positive votes. Therefore, he will continue having more positive votes in the future. For an edge (*u*, *v*) directing to *v*, the first element R_{out} indicates the voting history of the voter *u*, particularly whether his voting habit is positive or negative.

Benefit maximization: In economics, customers act to maximize the utility. They usually evaluate the quality of a product before they decide to spend money to buy it. Meanwhile, as the view of voting, voters seek to maximize their utility of votes. They consider that an object (video, candidate, article, etc) is worth voting for it. Therefore, the quality of the object affects directly the decisions of the voters and then, it should be considered for prediction. However, since most raw data of the object have been removed for security and copyright, viewing the object is impossible for prediction systems. Fortunately, we can evaluate its quality indirectly by taking its incoming edges into account. These edges show the attitude of other voters toward it and therefore, we can infer the quality of the object. That is what the ratio R_{in} means. The large value of R_{in} shows that there many people accept the object and this is an evidence for its good quality. By contrast, the small value means that many people complain about the object. Generally, a good object tends to receive more positive votes from new people and otherwise. Figure 3

demonstrates connections between video quality and the like-dislike ratio.

Herd behaviour: Individuals live in the society, communicate with other individuals and are influenced by surrounding people. Herd behaviour in human societies is defined as a phenomenon in which independent people observe and mimic the actions of others, even mistaken. This can be explained by the theory of *information and reputational cascade* [3]. *Information cascade* explains that a person tends to make the same choice with early responders because he believes in the choice of a crowd (many is better than one) and his idea is that "if I were wrong, all would be wrong and then, I will do as they do". Meanwhile, *reputational cascade* says that a person subjects himself to the crowd to maintain his reputation or avoid social disapproval. In the case of disagreement, voters tend to conform to the opinions of a majority or ignore voting. Hence, a person with many fans (large R_{in}) may usually receive more friend requests than hostile relationships. By contrast, an object with dozens of dislikes may continue obtaining more negative votes. Note that not all voters are affected by herd behaviour and therefore not 100% voters have the same choice. In addition, herd behaviour seems to not work when the positive and negative votes are similar. However, when the difference between positive and negative votes is clear, it enlarges the value of R_{in} and helps the PNR feature more discriminative.

Anchoring and Adjustment heuristic: Heuristics are general strategies which can help us make right decisions quickly [2]. One of the important and common heuristics is anchoring and adjustment heuristic in which people start with suggested reference information and make incremental adjustment to reach their decisions. In social networks, reference information includes the number of likes and dislikes, friends and enemies, fans and anti-fans, helpful and unhelpful votes. In fact, when a person wants to make friend with a stranger or votes for an article out of his area (e.g. a businessman searches for a technical report), such initial information will be useful anchors for him to reach his better estimate. In the PNR feature, these anchors (likes/dislikes, friends/enemies, etc) are encoded into the term of R_{in} to enrich prediction with necessary information.

Figure 3: The quality of videos can be expressed through the ratio of likes to dislikes: a) a hot video with the majority of likes and b) a video with dislikes.

While R_{out} represents the past experience of voters, R_{in} implies the principles of benefit maximization, herd behaviour, anchoring and adjustment heuristic. Among four principles, the two last principles (herd behaviour, anchoring and adjustment heuristic) only work in public networks where users can view the votes of early users. In other words, these principles rely on constraints (the security and policy of networks) and then, may become ineffective in some cases. Meanwhile, past experience and benefit maximization are personal factors. They can help a person reach his decisions themselves in general. Evidently, the PNR feature is deeply associated with decision-making theory. Large amount of vital clues for decision-making can be embedded inside two values of the PNR feature. As a result, we can obtain highly accurate prediction using a very low-cost feature.

E. Learning methodology

For the training step, we follow logistic regression model used by Leskovec in [7]. This model predicts the value of a binary dependent variable based on a set of predictor variables. Particularly, the sign (+/-) of link is considered as a dependent variable and each element of the feature is a predictor variable affecting the dependent variable. We choose this model for our study since it is a robust learning method successfully applied to signed link prediction. The logistic regression problem is formulated as a function:

$$P(+/x) = \frac{1}{1+e^{-\left(b_0+\sum_i^n b_i x_i\right)}} \qquad (6)$$

in which, $(x_1, x_2, \ldots, x_n)$ is a feature vector with n-dimensions; $(b_0, b_1, b_2, \ldots, b_n)$ are coefficients which will be learned in the training step. The function $P(+/x)$ maps an input vector $\boldsymbol{x}$ onto the interval [0, 1] and reveals the probability of a link being positive given $\boldsymbol{x}$. If $P(+/x) >= 0.5$ then the link should be more positive than negative and otherwise. In prediction, the feature vector x of a new link is predicted by using the function $P(+/\boldsymbol{x})$ with trained coefficients and comparing the output value with the threshold *0.5* to decide whether the sign is positive or negative.

III. EXPERIMENTS

1. Databases

We conduct our experiments in three standard databases including Epinions, Slashdot and Wikipedia [15]. In such databases, all edges are labeled positive or negative explicitly. Epinions is a trust network, in which the sign of a link (u, v) indicates that u trusts or distrusts v. The second database is the technology blog Slashdot, where the relationship of two users can be tagged either a "friend" or "foe" to show that the approval or disapproval between the users. Finally, Wikipedia is the voting network of Wikipedia community. The database was collected from election to the committee of the website. Some statistical information in these databases is summarized in TABLE I.

In the table, it notes that positive edges dominate all databases with approximately 80% of total edges. It is a feature of social networks where there exists an imbalance between positive and negative edges. Training with original unbalanced databases is biased since trained classifiers will tend to predict links are positive. Additionally, evaluation may be inaccurate since a naïve classifier that always returns positive signs (+)

can easily yield the high accuracy up to 80% in such databases. We overcome this bias by following the methodology of Guha et al. [10] to generate balanced databases. In particular, for each database, we keep all negative edges and sample a random positive edge for each negative one, which ensures that the number of positive and negative edges in data is balanced. These balanced databases are used in training and testing to ensure the objectivity of our experiments. However, since balanced databases are unreal and real databases are still our targets, we also test our learned models on original databases. Therefore, the models are evaluated in both balanced and original databases, one is to ensure that the models are not biased and the other is to evaluate their effectiveness in practice.

TABLE I. STATISTICS IN EPINIONS, SLASHDOT AND WIKIPEDIA

Galleries	#Nodes	#Edges	#Pos Edges	#Neg Edges	%Pos Edges
Epinions	131,828	841,372	717,667	123,705	85
Slashdot	82,144	549,202	425,072	124,130	77.4
Wikipedia	7,194	114,040	90,922	23,118	78.7

2. Experiments and discussion

We conduct experiments and compare our PNR with features proposed by Leskovec et al. in [7]. We choose these features because they are the most effective features at the present. For PNR, the threshold $t = 5$ is selected to yield the highest accuracy. We evaluate our proposed feature based on three criteria: the accuracy (classification accuracy and AUC), the generalization (classification accuracy and AUC across databases) and the speed (performance time).

TABLE II. CACC AND AUC ON AVERAGE IN 5 RUNS. EP, SL AND WI STAND FOR EPINIONS, SLASHDOT AND WIKIPEDIA RESPECTIVELY.

	Test / Train	Balanced Data (%)			Original Data(%)		
		Ep	**Sl**	**Wi**	**Ep**	**Sl**	**Wi**
CACC	**7 degree**	88.28	63.94	74.8	90.16	84.74	78.5
	16 triad	50.6	50.23	51.18	89.04	80.91	79.15
	All23	81.43	62.24	72.06	90.83	85.48	79.43
	PNR	**95.77**	**94.25**	**90.67**	**93.61**	**88**	**81.59**
AUC	**7 degree**	95.40	93.34	91.33	80.56	81.54	76.25
	16 triad	51.22	50.58	51.36	81.32	75.9	80.12
	All23	94.97	93.2	91.1	85.28	84.25	79.4
	PNR	**99.33**	**98.7**	**97.12**	**96.97**	**93.85**	**90.23**

2.1. Accuracy

For each original database (Epinions, Slashdot and Wikipedia), a balanced database is created by randomly selecting positive edges. We repeat the creation 5 times (called 5 runs) to reduce the effect of random selection. Each run includes 3 balanced databases corresponding to 3 original databases. We use k-fold cross validation (*k=10*) to evaluate the performance of features in each database. Experimental results in balanced databases are given in TABLE II. The table shows that the PNR feature outperforms the others at least 3% for classification accuracy (CACC) and AUC in both balanced and original databases. The exciting thing here is that although PNR has two dimensions, it can obtain high accuracy. Due to the solid theory behind, we need only a simple design for the effectiveness of the state-of-the-art features. These results show that decision-making theory not only helps to understand the link sign formulation but also provides an excellent guideline on extracting essential and compact information for signed link prediction.

TABLE III. CLASSIFICATION ACCURACY OF FEATURES ACROSS DATABASES

	Test / Train	Balanced Data (%)			Original Data (%)		
		Ep	**Sl**	**Wi**	**Ep**	**Sl**	**Wi**
Epinions	**7 degree**	86.66	81.39	75.29	90.18	83.7	75.48
	16 triad	67.82	59.27	67.72	91.6	81.47	78.6
	All23	85.86	81.05	75.60	90.79	83.44	76.43
	PNR	**91.28**	**87.25**	**82.80**	**93.62**	**88.62**	**81.69**
Slashdot	**7 degree**	82.67	79.86	68.92	90.44	84.74	74.87
	16 triad	67.31	59.74	65.54	91.91	80.99	75.82
	All23	82.79	79.57	70.49	92.4	85.55	76.15
	PNR	**91.26**	**87.55**	**83.54**	**93.24**	**87.99**	**81.33**
Wikipedia	**7 degree**	83.78	77.87	80.7	90.32	83.27	78.48
	16 triad	67.33	59.05	68.63	91.19	81	79.11
	All23	84.16	77.92	81.33	90.97	83.61	79.36
	PNR	**91.05**	**87.44**	**83.79**	**93.22**	**87.74**	**81.6**

TABLE IV. AUC OF FEATURES ACROSS DATABASES

	Test / Train	Balanced Data (%)			Original Data (%)		
		Ep	**Sl**	**Wi**	**Ep**	**Sl**	**Wi**
Epinions	**7 degree**	91.29	85.96	80.98	80.43	77.13	70.56
	16 triad	77.85	63.77	75.51	88.41	73.27	80.63
	All23	91.92	86.58	82.73	84.98	77.91	75.12
	PNR	**91.92**	**94.87**	**91.81**	**96.98**	**93.79**	**89.39**
Slashdot	**7 degree**	92.37	88.82	80.07	82.85	81.58	68.56
	16 triad	76.87	65.31	72.9	90.9	75.96	82.89
	All23	92.83	89.17	79.35	89.71	83.87	77.06
	PNR	**97.75**	**94.89**	**92.03**	**97.09**	**93.84**	**89.82**
Wikipedia	**7 degree**	90.76	86.39	86.29	83.94	76.27	76.18
	16 triad	76.8	64.1	76.69	87.56	73	80.1
	All23	91.44	86.67	87.03	86.42	77.55	79.44
	PNR	**97.56**	**94.75**	**92.22**	**97.01**	**93.54**	**90.24**

2.2. Generalization

Generalization is another measure to evaluate methods in practical applications. We define the generalization of a method as its quality in difference databases. That is, a method with high generalization has excellent results not only in the training databases but also other databases. This characteristic is very important because databases in practical applications are usually completely different from training databases. Therefore, a method with high generalization means with high applicability and reliability. To prove the generalization of PNR, we evaluate the trained model across databases. TABLE III and IV reveals CACC and AUC values of features tested in different databases. Among them, PNR is the best. For any combination of training and testing databases, our proposed PNR feature is always superior to the others. This is due to the fact that PNR does not reply on the particular structures of networks but is based on the general psychology of people to predict signs of links.

2.3. Speed

Now we turn to the speed evaluation for PNR. We measure the performance time in three steps: feature extraction, training and prediction steps. While the computational time of feature extraction and training step can be measured directly in each training data, the prediction time is the average of performance time in three unbalanced databases or three original databases (TABLE V). All experiments in this section are conducted on the same PC with 2.66 GHz CPU and 4G RAM. It is obvious that PNR is the fastest while the second is the 7 degree feature. PNR is 5, 1.3 and 1.5 times faster than 7 degree in extraction, training and prediction steps respectively. An explanation is that PNR has 2 dimensions compared to 7 of 7 degree, 16 of 16 triads and 23 of All23. Therefore, our PNR has the benefit of the speed through all steps of the system.

TABLE V. PERFORMANCE TIME (IN SECOND) OF EXTRACTION, TRAINING AND PREDICTION STEPS.

Training data		Extract	Train	Predict(Average)	
				Balanced	Original
Epinions	**7 degree**	81.68	104.23	0.033	0.063
	16 triad	511.59	269.42	0.055	0.109
	All23	599.01	541.17	0.072	0.146
	PNR	**16.17**	**2.32**	**0.021**	**0.041**
Slashdot	**7 degree**	80.7	2.77	0.034	0.065
	16 triad	503.26	262.62	0.054	0.107
	All23	589.01	542.63	0.071	0.148
	PNR	**16.1**	**1.88**	**0.02**	**0.041**
Wikipedia	**7 degree**	15.21	0.71	0.033	0.069
	16 triad	96.96	0.96	0.052	0.11
	All23	114.86	1.31	0.07	0.146
	PNR	**2.95**	**0.52**	**0.018**	**0.037**

IV. CONCLUSION

We have investigated the problem of signed link prediction that determines the signs of links which may be positive or negative. Due to some limitations of current theories for signed link prediction problem, we have applied decision-making theory to building a low-cost but effective feature, called PNR. The strength of our feature is that it has a close connection with the decision-making theory in terms of past experience; benefit maximization; herd behaviour; anchoring and adjustment heuristic. The theoretical foundation helps PNR significantly outperform the state-of-the-art features in all aspects: the accuracy, the generalization and the speed. Furthermore, PNR is very simple for implementation since only two float values are needed to form a PNR feature vector. Many reliable experiments with different merits were conducted in various databases (both balanced and original databases). The results show that our proposed feature really improves on previous approaches significantly.

REFERENCES

[1] E.A. Jullisson, N. Karlsson and T. Garling, "Weighing the past and the future in decision making", *European Journal of Cognitive Psychology, 17*(4), 561-575. DOI: 10.1080/09541440440000159, 2005.

[2] A.K. Shah, and D.M. Oppenheimer, "Heuristics made easy: An effort-reduction framework". *Psychological Bulletin, 134*(2), 207-222. DOI: 1.1037/0033-2909.134.2.207, 2008.

[3] Pierre Lemieux, "Following the Herd", Regulation, Cato Institute, 21, Retrieved 14 July 2010.

[4] D. Cartwright and F. Harary, "Structure Balance: A generalization of Heider's theory', Psych. Rev., 63, 1956.

[5] D. Corlette & F. Shipman, "Link Prediction Applied to an Open Large-Scale Online Social Network". HTACM (2010) , p. 135-140. 2010

[6] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks". American Society for Information Science and Technology, 58(2007).

[7] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks". In WWW, pages 641–650, 2010.

[8] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Signed networks in social media". In CHI, pages 1361–1370, 2010.

[9] K. Chiang, N. Nagarajan, Ambuj Tewari, and Inderjit S. Dhillon. "Exploiting longer cycles for link prediction in signed networks". In Proceedings of the 20th ACM Conference on Information and Knowledge Management, pages 1157-1162, 2011.

[10] R. V. Guha, R. Kumar, P. Raghavan, and A. Tomkins, "Propagation of trust and distrust". In Proc. 13th WWW, 2004.

[11] Nan Ma, Ee P. Lim, Viet A. Nguyen, Aixin Sun, Haifeng Liu, "Trust Relationship Prediction Using Online Product Review Data". In CIKM-CNIKM, pp. 47-54, 2009

[12] S. Wasserman and K. Faust, "Social Network Analysis: Methods and Applications". Cambridge Univ. Pr., 1994.

[13] T. Murata, S. Morivasu. "Link prediction of social networks based on weighted proximity measures". Proc. WIC. 2007

[14] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. W. D. Luca, and S. Albayrak. "Spectral analysis of signed graphs for clustering, prediction and visualization". In SDM, pages 559–570, 2010.

[15] http://snap.stanford.edu/data/