Ranking Learning on the Web by Integrating Network-based Features

Yingzi Jin
The University of Tokyo,
2-11-16 Yayoi, Bunkyou-ku, Tokyo, Japan
eiko-kin@biz-model.t.u-tokyo.ac.jp

Yutaka Matsuo
matsuo@biz-model.t.u-tokyo.ac.jp
Mitsuru Ishizuka
ishizuka@i.u-tokyo.ac.jp
The University of Tokyo

Abstract

Many efforts are undertaken by people and companies to improve their popularity, growth, and power, the outcomes of which are all expressed as rankings (designated as target rankings). Are these rankings merely the results of its elements' own attributes? In the theory of social network analysis (SNA), the performance and power of actors are usually interpreted as relations and the relational structures they embedded. In this study, we propose an algorithm to generate and integrate network-based features systematically from a given social network that mined from the Web to learn a model for explaining target rankings. Experimental results for learning to rank researchers' productivity based on social networks confirm the effectiveness of our models. This paper specifically examines the application of a social network that provides an example of advanced utilization of social networks mined from the Web.

1. Introduction

People prefer to use rankings for comparing companies, discussing elections, and evaluating goods. For example, investors seek to invest funds in fast-growing and stable companies; consumers tend to buy highly popular products. Therefore, many efforts are undertaken by people and companies to improve their popularity, growth, and power, the outcomes of which are all expressed as rankings. Conventionally, these rankings are evaluated and ranked by values from statistical data and attributes of actors such as income, education, personality, and social status.

In the theory of social network analysis (SNA), social networks are used to analyze the performance and valuation of social actors [9]. Network researchers have argued that relational and structural embeddedness influence individual's behavior and performance, and that a successful person must therefore emphasize relation management. Actually, a number of relations exist in the world with different impacts; the actors might be tied together closely in one relational network, but can differ greatly from one to another in a different relational network. The question arises: *relations of what kind are important for entities?* Unfortunately, the answers of important relations have been decided according to the judgment of researchers themselves.

To identify the prominence or importance of an individual actor embedded in a network, centrality measures have been used in social sciences: degree centrality, betweenness centrality, and closeness centrality. These measures often engender distinct results with different perspectives of "actor location? i.e., local (e.g. degree) and global (e.g. eigenvector) locations, in a social network [9]. Another question arises: *what kind of centrality indices are most appropriate for ranking actors?* That question can be extended as *what kind of structural embeddedness of actors makes them more powerful?*

This paper presents a description of an attempt to learn the ranking of named entities from a social network that has been mined from the Web. It enables us to have a model to rank entities for various purposes: one might wish to rank entities for search and recommendation, or might want to have the ranking model for prediction. Given a list of entities, we first extract different types of relations from the Web based on our previous work [6, 3]. Subsequently, we rank the entities on these networks using different network indices. In this paper, we propose a systematic algorithm which integrates features generated from networks (designated as *network-based features*) for each and then use these features to learn and predict rankings. We conducted experiments related to social networks among researchers to learn and predict the ranking of researchers' productivity.

The contributions of this study can be summarized as follows. We provide an example of advanced utilization of a social network mined from the Web. The results illustrate the usefulness of our approach, by which we can understand the important relations as well as the important structural embeddedness to predict ranking of entities. The model can be combined with a conventional attribute-based approach.

978-0-7695-3689-7/09 $25.00 © 2009 IEEE
DOI 10.1109/ASONAM.2009.39

Results of this study will provide a bridge between relation extraction and ranking learning for advanced knowledge acquisition for Web intelligence.

The following section presents a description of an overview of the ranking learning model. Section 3 briefly introduces our previous work for extracting social networks from the Web. Section 4 describes proposed ranking learning models based on extracted social networks. Section 5 describes the experimental settings and results. Section 6 presents some related works before the paper concludes.

2 System Overview

Our study explores the integration of mining relations (and structures) among entities and the learning ranking of entities. For that reason, we first extract relations and then determine a model based on those relations. Our reasoning is that important relations can be recognized only when we define some tasks. These tasks include ranking or scores for entities, i.e., *target ranking* such as ranking of companies, CD sales, popular blogs, and sales of products.

Step 1: Constructing Social Networks Given a list of entities with a target ranking, we extract a set of social networks among these entities from the Web.

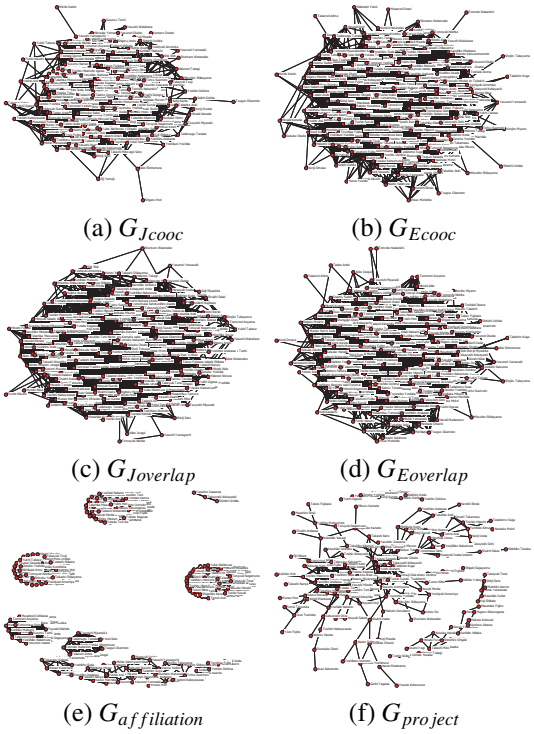
Step 2: Ranking learning Learn a ranking model based on the relations and structural features generated from the networks.

Once we obtain a ranking model, we use it for prediction for unknown entities. Additionally, we can obtain the weights for each relation type as well as relation structure, which can be considered as important for target rankings. The social network can be visualized by specifically examining its relations if the important relations are identified. Alternatively, social network analysis can be executed based on the relations.

3 Constructing Social Networks

In this step, our task is, given a list of entities $V = \{v_1, \ldots, v_n\}$, construct a set of social networks $G_i(V, E_i)$, $i \in \{1, \ldots, m\}$, where m signifies the number of relations, and $E_i = \{e_i(v_x, v_y) | v_x \in V, v_y \in V, v_x \neq v_y\}$ denotes a set of edges with respect to the *i-th* relation.

A social network is obtainable through various approaches [3, 7, 6]. In this paper, we detail the Web mining approach: *co-occurrence-based approach* and *classification-based approach* as a basis of our study. For the *co-occurrence-based approach* [7, 6], given a person name list, the strength of relevance of two persons, x and y, is estimated by putting a query x AND y to a search engine.



(a) G_{Jcooc} (b) G_{Ecooc} (c) $G_{Joverlap}$ (d) $G_{Eoverlap}$ (e) $G_{affiliation}$ (f) $G_{project}$

Figure 1. Web-based social networks for researchers with different relational indices or types.

An edge will be invented when the relation strength by the co-occurrence measure is higher than a predefined threshold. Subsequently, we extract two kinds of co-occurrence-based networks: cooc network (G_{cooc}), and overlap network ($G_{overlap}$). The relational indices are calculated respectively using the matching coefficient $n_{x \wedge y}$, and the overlap coefficient $n_{x \wedge y} / \min(n_x, n_y)$, where n_k means the number of hit by issuing query k to a search engine. For the *classification-based approach* [6], based on Web co-occurrence networks edges are classified into several relations using C4.5 as a classifier. In our experiments, we first extract overlap network among researchers, then classify the edges into two kinds of relational networks: an co-affiliation network ($G_{affiliation}$) and a co-project network ($G_{project}$). Because of space limitation, we do not show the details of construction algorithms. Details can be shown from [6]. Extracted networks for 253 researchers are portrayed in Fig. 1. We can see that the social networks vary with different relational indices or types even though they contain the same list of entities.

4 Ranking Leaning Model

For the list of nodes $V = \{v_1, \ldots, v_n\}$, given a set of networks $G_i(V, E_i)$, $i \in \{1, \ldots, m\}$ (constructed by section 3) with a target ranking $\mathbf{r}^*$ ($\in R^t$) (where $t \leq n$, and r_k^* denotes

k-th element of the vector $\mathbf{r}^*$ and means the target ranking score of entity v_k), the goal is to learn a ranking model based on these networks.

First, as a baseline approach, we follow the intuitive idea of simply using approach from SNA (i.e. centrality) to learn ranking. Then we propose a more systematic algorithm that generates various network features for individuals from social networks.

4.1 Baseline Model

Based on the intuitive approach, we first overview commonly used indices in social network analysis and complex network studies. Given a set of social networks, we rank entities on these networks using different network centrality indices. We designate these rankings as *network rankings* because they are calculated directly from relational networks.

To address the question of what kind of relation is most important for entities, we intuitively compare rankings caused by relations of various types. Although simple, it can be considered as an implicit step of social network analysis given a set of relational networks. We merely choose the type of relation that maximally explains the given ranking. We rank each type of relational network; then we compare the *network ranking* with the *target ranking*. Intuitively, if the correlation to the network ranking $\mathbf{r}_{\hat{i}}$ is high, then the relation $\hat{i}$ represents the important influences among entities for the given target ranking. Therefore, this model is designed to find an optimal relation $\hat{i}$ from a set of relations:

$$\hat{i} = \underset{i \in \{1,\ldots,m\}}{\operatorname{argmax}} Cor(\mathbf{r}_i, \mathbf{r}^*) \tag{1}$$

For different relational networks with different centrality indices, the network ranking from i-th network with j-th centrality ranking can be presented as $\mathbf{r}_{i,j}$ ($\in R^n$), where $i \in \{1,\ldots,m\}$, and $j \in \{1,\ldots,s\}$. Therefore, the first method can be extended simply to find a pair of optimal parameters $< \hat{i}, \hat{j} >$ (i.e., i-th network by j-th centrality rankings) that maximizes the coefficient between network rankings with a target ranking.

$$< \hat{i}, \hat{j} >= \underset{i \in \{1,\ldots,m\}\ j \in \{1,\ldots,s\}}{\operatorname{argmax}} Cor(\mathbf{r}_{i,j}, \mathbf{r}^*) \tag{2}$$

4.2 Network-based Feature Integration Model

The proposed method in our research is to integrate multiple indices that are obtained from multiple social networks to learn the target rankings. A feature by itself (e.g. a centrality value) may have little correlation with the target ranking, but when it is combined with some other features, they may be strongly correlated with the target rankings [10]. Simply, we can integrate various centrality values for each actor, thereby combining different meanings of importance to learn the ranking. Furthermore, we can generate more relational and structural features from a network for each, such as how many nodes are reachable, how many connections one's friends have, and the connection status in one's friends. We might understand some about the behavior and power about the individual as well as we predict their ranking if we could know the structural position of individuals. Herein, we designate these features generated from networks as *network-based features*. The interesting question is *how to generate network-based features from networks for each*, and *how to integrate these features to learn and predict rankings*. Below we will describe the approach.

4.2.1 Generating Network-based Features for nodes

For each x, we first define node sets with relations that might effect x. We define a set of nodes $C_x^{(k)}$ as a set of nodes within distance k from x. We choose a node set adjacent to node x (designated as C_x^1), and also choose a node set that contains all of reachable nodes from x (designated as $C_x^{(\infty)}$) as influential nodes for x.

Then we apply some operators to the set of nodes to produce a list of values. The simple operation for two nodes is to check whether the two nodes are adjacent or not. We denote these operators as $s^{(1)}(x, y)$, which returns 1 if nodes x and y are mutually connected, and 0 otherwise. We also define operator $t(x, y) = argmin_k\{s^{(k)}(x, y) = 1\}$ to measure the geodesic distance between the two nodes on the graph. These two operations are applied to each pair of nodes in nodeset N, and which can be defined as $Operator \circ N = \{Operator(x, y) | x \in N, y \in N, x \neq y\}$. For example, if we are given a node set $\{ n_1, n_2, n_3\}$, we can calculate $s^{(1)}(n_1, n_2)$, $s^{(1)}(n_1, n_3)$, and $s^{(1)}(n_2, n_3)$ and return a list of three values, e.g., $(1, 0, 1)$. We denote this operation as $s^{(1)} \circ N$. In addition, to s and t operations, we define two other operations. One operation is to measure the distance from node x to each node, denoted as t_x. Instead of measuring the distance between two nodes, $t_x \circ N$ measures the distance of each node in N from node x. Another operation is to check the shortest path between two nodes. Operator $u_x(y, z)$ returns 1 if the shortest path between y and z includes node x. Consequently, $u_x \circ N$ returns a set of values for each pair of $y \in N$ and $z \in N$.

After that, the values calculated by above operations are aggregated into a single feature value. Given a list of values, we can take the summation (Sum), average (Avg), maximum (Max), and minimum (Min). For example, if we apply Sum aggregation to a value list $(1, 0, 1)$, we obtain a value of 2. We can write the aggregation as e.g., $Sum \circ s^{(1)} \circ N$. Although other operations can be performed, such as taking the variance or taking the mean, we limit the operations to the four described above. The value obtained here results in the network-based feature for node x. Additionally, we

Table 1. Operator list

Notation	Description
$C_x^{(1)}$	adjacent nodes to x
$C_x^{(k)}$	nodes within distance k from x
$s^{(1)}$	1 if connected, 0 otherwise
t	distance between a pair of nodes
t_x	distance between node x and other nodes
γ	number of links in each node
u_x	1 if the shortest path includes node x, 0 otherwise
Avg	average of values
Sum	summation of values
Min	minimum of values
Max	maximum of values
$Ratio$	ratio of value on nodeset $C_x^{(1)}$ by nodeset $C_x^{(k)}$

can take the difference or the ratio of two obtained values. For example, if we obtain 2 by $Sum \circ s^{(1)} \circ C_x^{(1)}$ and 1 by $Sum \circ s^{(1)} \circ C_x^{(k)}$, the ratio is $2/1 = 2.0$.

The nodesets, operators, and aggregations are presented in Table 1. We have $2(nodesets) \times 5(operators) \times 4(aggregations) = 40$ combinations. There are ratios for $C_x^{(1)}$ to $C_x^{(k)}$ if we consider the ratio. In all, there are 4×5 more combinations: there are 60 in all. Each combination corresponds to a feature of node x. The resultant value sometimes corresponds to a well-known index, as we intended in the design of the operators. For example, the degree centrality can be denoted as $Sum \circ s_x^{(1)} \circ C_x^{(1)}$, and closeness centrality is as $Avg \circ t_x \circ C_x^{(\infty)}$. These features represent some possible combinations. Some lesser-known features might actually be effective.

4.2.2 Network-based feature Integration

After we generate various network-based features for individual nodes, we integrate them to learn ranking. We introduce an f-dimensional feature vector F, in which each element represents a network-based feature for each node. We identify the f-dimensional combination vector $\mathbf{u} = [u_1, \ldots, u_f]^T$ to combine network-based features for each node. The inter-product $\mathbf{u}^T\mathbf{F}$ for each node produces n-dimensional ranking. For relational networks of m kinds, the feature vector can be expanded to $m \times 60$-dimensions. In this case, the purpose is finding out whether optimal combination weight $\hat{\mathbf{u}}$ maximally explains the target ranking:

$$\hat{\mathbf{u}} = \underset{\mathbf{u}}{\operatorname{argmax}}\, Cor(\mathbf{u}^T \bullet F, \quad \mathbf{r}^*) \tag{3}$$

This model can be extended easily to add traditional attributes of entities as features. We can use any technique, such as SVM, boosting and neural network, to implement the optimization problem. For multi-relational networks, we can generate features for each single-relational network. Subsequently, we can compare the performance among them to understand which relational network produces more reasonable features. Thereby, we can see which relation(s) is important for the target ranking.

5 Experimental Results

In this section, we describe results to clarify the effectiveness of ranking learning on extracted social networks. We use 253 researchers from The University of Tokyo to predict a ranking of researchers. In our experiments, we conducted three-fold cross-validation. In each trial, two folds of actors are used for training, and one fold for prediction. The results we report in this section are those averaged over three trials. We use Spearman's rank correlation coefficient to measure the pairwise ranking correlation between predicted rankings and target ranking.

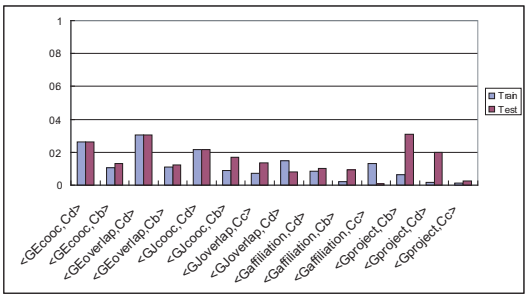
5.1 Datasets

We extract social networks for researchers (253 professors of The University of Tokyo) to learn and predict the ranking of researchers. In this paper, we use the ranking by the number of publications (designated as Paper) as a target ranking. Academic papers are often the product of several researchers' collaboration. Therefore, a good position in a social network is derived through good performance. Is there any relation that is important to predict productivity?

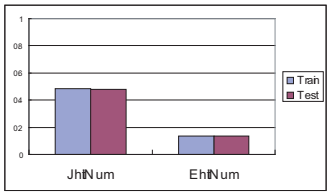
We construct social networks among researchers from the Web using a general search engine. We detail the co-occurrence-based approach to extract networks of two kinds in English-language Web sites and Japanese Web sites respectively: cooc network (G_{Ecooc}, G_{Jcooc}) and overlap network ($G_{Eoverlap}$, $G_{Joverlap}$). Actually, we used English/romanized names of researchers as a query to obtain co-occurrence information for G_{Ecooc} and $G_{Eoverlap}$, and used Japanese names of researchers as a query to obtain co-occurrence information for G_{Jcooc} and $G_{Joverlap}$. Then, we use the classification-based approach based on Web co-occurrence networks (in Japanese Web sites) to classify the edges into relational networks of two kinds: a co-affiliation network ($G_{affiliation}$) and a co-project network ($G_{project}$). Extracted networks for 253 researchers are portrayed in Fig. 1. For this experiment, we also use two types of researchers attributes available from the Web: the number of hits on Japanese Web sites JhitNum (using Japanese names as a query) and the number of hits on the English-language Web sites EhitNum (using English/romanized names as a query). More kinds of networks and attributes can be added for improving the effectiveness of results in the future.

5.2 Ranking Results

For the baseline model, three centrality indices (degree centrality C_d, closeness centrality C_c, and betweenness cen-



(a) centrality-based network ranking



(b) attribute-based ranking

Figure 2. Evaluation for each attribute-based ranking as well as centrality-based network ranking with target ranking among researchers.

trality C_b) are used on different networks as network rankings. We calculate the correlation between network rankings with each target ranking of Paper. For comparison, we also rank companies according to attributes (i.e., JhitNum and EhitNum), and take correlation with target ranking. Fig. 2 portrays correlations (mean of three tries) of each network rankings as well as each attribute-based rankings with different target rankings on training and testing data among researchers. Results show that the JhitNum is a good attribute of researchers for predicting the creditability of publications. Furthermore, degree centralities in $G_{Eoverlap}$ as well as in G_{Ecooc} (i.e., $\mathbf{r}_{G_{Eoverlap},C_d}$ and $\mathbf{r}_{G_{Ecooc},Cd}$, respectively) exhibit a good correlation with target ranking. We can say that researchers who are famous on Japanese Web sites and who frequently co-occur with other researchers on English-language Web sites are the more creative researchers.

We execute our feature integration ranking model (with several varies) to single and multiple social networks to train and predict rankings of researchers' Paper. We use Ranking SVM to learn the ranking model which minimize pairwise training error in the training data. Then we apply the model to predict the rankings on training data (again) and on testing data. Comparable results on several variations of model are presented in Table 2. First, we integrate proposed network-based features obtained from each type of single network as well as multiple networks among researchers to train and predict the rankings. The co-occurrence-based networks especially on English-language Web sites G_{Ecooc}, $G_{Eoverlap}$, and $G_{Joverlap}$ appear to be a better explanation of target ranking of Paper than the co-affiliation network $G_{affiliation}$ or the co-project network

Table 2. Results of feature integration among researchers.

Model	Feature	PaperNum	
		Train	Test
Network	G_{Ecooc}	0.470	0.413
	$G_{Eoverlap}$	0.508	0.411
	G_{Jcooc}	0.443	0.261
	$G_{Joverlap}$	0.585	0.325
	$G_{affiliation}$	0.178	-0.011
	$G_{project}$	0.540	0.043
	G_{ALL}	0.821	**0.417**
Attributes	ALL	0.491	**0.448**
Network + Attributes	G_{Ecooc}+A	0.514	0.429
	$G_{Eoverlap}$+A	0.544	0.404
	G_{Jcooc}+A	0.481	0.284
	$G_{Joverlap}$+A	0.519	0.420
	$G_{affiliation}$+A	0.497	0.159
	$G_{project}$+A	0.548	0.304
	G_{ALL}+A	0.811	**0.456**

$G_{projext}$. Using features from all kinds of networks G_{ALL}, the prediction results are better than for any other single-relational network. Then, as comparison, we integrate attribute-indices (i.e., JhitNum and EhitNum) of researchers as features as a baseline of this model to learn and predict the target ranking of Paper. We can obtain a 0.448 correlation coefficient between predicted rankings and target rankings, which is explainable: famous researchers are also famous on the Web sites. Furthermore, when we combine network-based features with attribute-based features to learn the model, the results outperform each using attribute-based features only and network-based features only.

5.3 Detailed Analysis of Useful Features

To clarify the usefulness of network-based features, we use them separately to train and expect the target rankings. Leaving out one feature, the others are used to train and predict the rankings. In fact, k-th feature is a useful feature for explaining the target ranking if the result worsens much when leaving out the feature k from the feature set. Table 3 presents the effective features for the target ranking of Paper in the researcher field. For example, the maximum number of links in the reachable nodeset of x from cooc network from English-language Web sites $Max \circ \gamma \circ C_x^{(\infty)} \circ G_{Ecooc}$ is effective for the target ranking, which means that if a famous researcher is reachable from a person, that person can be more productive. The minimum number of links in the neighbor nodeset of x from the cooc network from Japanese Web sites $Min \circ \gamma \circ C_x^{(1)} \circ G_{Jcooc}$ is also effective, which means that if a direct neighbor is productive, then x will be more productive. The ratio of the number of edges among neighbors to the number

Table 3. Effective features in various networks for Paper among researchers.

Top	Features from social networks
1	$Max \circ \gamma \circ C_x^{(\infty)} \circ G_{Ecooc}$
2	$Min \circ \gamma \circ C_x^{(1)} \circ G_{Jcooc}$
3	$Avg \circ \gamma \circ C_x^{(\infty)} \circ G_{Eoverlap}$
4	$Max \circ t \circ C_x^{(\infty)} \circ G_{Joverlap}$
5	$Avg \circ u_x \circ C_x^{(1)} \circ G_{Eoverlap}$
6	$Min \circ \gamma \circ C_x^{(1)} \circ G_{Eoverlap}$
7	$Min \circ \gamma \circ C_x^{(\infty)} \circ G_{Jcooc}$
8	$Ratio \circ (Sum \circ s^{(1)} \circ C_x^{(1)}, Sum \circ s^{(1)} \circ C_x^{(\infty)}) \circ G_{project}$
9	$Avg \circ \gamma \circ C_x^{(1)} \circ G_{Joverlap}$
10	$Min \circ \gamma \circ C_x^{(1)} \circ G_{Ecooc}$

of edges among reachable nodes from co-project network $Ratio \circ (Sum \circ s^{(1)} \circ C_x^{(1)}, Sum \circ s^{(1)} \circ C_x^{(\infty)}) \circ G_{project}$ means that binding neighbors from all of reachable nodes in projects makes the researcher more productive.

We understand that various features have been shown to be important for real-world rankings (i.e. target ranking). Some of them correspond to well-known indices in social network analysis. The results support the usefulness of the indices that are commonly used in the social network literature, and underscore the potential for additional composition of useful features.

6 Related Works

In the context of information retrieval, PageRank and HITS algorithms can be considered as well known examples for ranking Web pages based on the link structure. Recently, more advanced algorithms have been proposed for learning to ranking entities. Although quite a few studies of learning-to-rank fields (particularly targeted on information retrieval) have investigated many attribute-based ranking functions learned from given preference orders, only a few studies have concluded that such an impact arises from relations and structures [8, 1]. Furthermore, our model is target-dependent: the important features of relations and structural embeddedness vary among different tasks.

Relations and structural embeddedness influence behavior of individuals and growth and change of the group [9]. Several researchers use network-based features for analyses. Backstrom et al. [2] describe analyses of community evolution, and show some structural features characterizing individuals—positions in the network. Liben-Nowell et al. [5] elucidate features using network structures in the link prediction problem. We specifically examine relations and structural features for individuals (previously for link prediction in [4]) and deal with various structural features from multiple networks systematically for learning the real world rankings (i.e. target rankings).

7 Conclusion

This paper described methods of learning the ranking of entities from social networks mined from the Web. We first extracted social networks of different kinds from the Web. Subsequently, we used these networks and a given target ranking to learn a ranking model. We proposed an algorithm to learn the model by integrating network-based features from a given social network that mined from the Web. Results of experiments on researcher field reveal that effectiveness of our models for explaining target ranking of researchers' productivity using multiple social networks mined from the Web. The results underscore the usefulness of our approach, by which we can understand the important relations as well as important structural embeddedness to predict the rankings. Our model provides an example of advanced utilization of a social network mined from the Web.

References

[1] A. Agarwal, S. Chakrabarti, and S. Aggarwal. Learning to rank networked entities. In *Proc. KDD'06*, 2006.

[2] L. Backstrom, D. Huttenlocher, X. Lan, and J. Kleinberg. Group formation in large social networks: Membership, growth, and evolution. In *Proc. SIGKDD'06*, 2006.

[3] Y. Jin, Y. Matsuo, and M. Ishizuka. Extracting social networks among various entities on the web. In *ESWC2007*, 2007.

[4] J. Karamon, Y. Matsuo, and M. Ishizuka. Generating useful network-based features for analyzing social network. In *Proc. AAAI2008*, 2008.

[5] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proc. CIKM*, pages 556–559, 2003.

[6] Y. Matsuo, J. Mori, M. Hamasaki, K. Ishida, T. Nishimura, H. Takeda, K. Hasida, and M. Ishizuka. POLYPHONET: an advanced social network extraction system. In *Proc. WWW2006*, 2006.

[7] P. Mika. Flink: semantic web technology for the extraction and analysis of social networks. *Journal of Web Semantics*, 3(2):211–223, 2005.

[8] T. Qin, T. Liu, X. Zhang, D. Wang, W. Xiong, and H. Li. Learning to rank relational objects and its application to web search. In *Proc. WWW2008*, 2008.

[9] S. Wasserman and K. Faust. *Social network analysis. methods and applications*. Cambridge University Press, Cambridge, 1994.

[10] Z. Zhao and H. Liu. Searching for interacting features. In *Proc. IJCAI2007*, 2007.