

Outcome Aware Ranking in Interaction Networks

Sampath Kameshwaran^{*}
IBM Research, India
Bengaluru, India

Vinayaka Pandit
IBM Research, India
Bengaluru, India

Sameep Mehta
IBM Research, India
New Delhi, India

Nukala Viswanadham
Indian School of Business
Hyderabad, India

Kashyap Dixit[†]
Pennsylvania State University
Pennsylvania, United States

ABSTRACT

In this paper, we present a novel ranking technique that we developed in the context of an application that arose in a Service Delivery setting. We consider the problem of ranking agents of a service organization. The service agents typically need to interact with other service agents to accomplish the end goal of resolving customer requests. Their ranking needs to take into account two aspects: firstly, their importance in the network structure that arises as a result of their interactions, and secondly, the value generated by the interactions involving them. We highlight several other applications which have the common theme of ranking the participants of a value creation process based on the network structure of their interactions and the value generated by their interactions. We formally present the problem and describe the modeling technique which enables us to encode the value of interaction in the graph. Our ranking algorithm is based on extension of eigen value methods. We present experimental results on real-life, public domain datasets from the Internet Movie DataBase. This makes our experiments replicable and verifiable.

Categories and Subject Descriptors

H.2.8 [Database Management]: Database Applications—
Data Mining

General Terms

Algorithms, Experimentation

^{*}{kameshwaran.s, pvinayak, sameepmehta}@in.ibm.com

[†]The work was done while the author was at IBM Research - India

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CIKM-10 October 26–30, 2010, Toronto, Ontario, Canada.
Copyright 2010 ACM 978-1-4503-0099-5/10/10 ...\$10.00.

Keywords

Node Ranking, Interaction Networks, Social Network Analysis, Centrality Measures

1. INTRODUCTION

There are many situations from quite diverse domains which display the following common characteristics. A group of people or agents are involved in a set of activities that are intended to create value. The process of value creation is rarely based on individual ability and typically involves interaction or collaboration. So, the end result of the process is two pronged: firstly, there is a cumulative value created due to the interactions between the agents and secondly, an interaction network that captures the pattern of historical interactions between the agents involved in the value creation process. Therefore, when we want to rank the agents in such settings we have to take into account their influence in the network as well as their contribution to the value creation process.

We begin with two motivating applications. First one is an application in software services setting which directly motivated our research work. One of the distinguishing features of the Services industry is that it involves people interacting with each other to solve a client's specific problem rather than transforming goods (or raw materials) to produce a product. One of the predominant components of the software services industry is that of providing maintenance and support services. Essentially, the service organization takes the responsibility for resolving requests for maintenance and support activities of a software product (or a specific module of a software product). The requests may come from either the users of the software product or a testing team of the software product. A service delivery manager's view of the operations is as follows: a series of requests arrive; for each request, the manager puts together a team of agents to resolve the request; an *outcome* indicating the effectiveness with which the request is resolved is recorded. After a sufficiently long duration of delivering such a service with a stable set of agents, the delivery manager may wish to rank the agents. One might be tempted to believe that the manager would like to rank the agents based on their average effectiveness. But, our interviews with domain experts revealed that the importance of an agent is not just based on average effectiveness. Domain experts would like to take into account an agent's connection pattern in the delivery

network as well due to the following reasons. A typical request requires modifications to one or more submodules of the product and also it needs agents with different roles such as developer, reviewer, tester etc. Since each request requires handoffs between the agents responsible for them, an agent’s connections indirectly capture the agent’s influence: familiarity with the subdomains of the connected agents due to the handoffs and ability (or inability) to work with the other agents (depending on the outcomes). When a new service request arrives, estimates of the agents’ influence (or rank) and their role/expertise can be utilized to make a decision on the composition of the team to resolve the request.

As mentioned before, similar problem arises in diverse scenarios. As a second example, we consider the problem of ranking authors based on academic publications. The goal of academic publication is to disseminate new knowledge and novel insights obtained from academic research. So, the outcome of every academic publication can be measured based on parameters like, the conference or journal in which it appeared, sustained citations it gets over a period of time, awards it wins and so on. At the same time, the influence of an academic researcher is not determined merely by the outcomes of the papers. The connections an author develops and the influence of coauthors play a major role in the overall influence of an academic researcher. So, the ranking of academic researchers needs to take into account the structure of their interactions (obtained from a database like DBLP) and the impact of their papers (obtained from citations, impact ratings of conferences etc). Some of the works in this direction can be seen in [10, 11, 9].

The literature of social network analysis (SNA) is rich on ranking nodes [13, 7, 12, 8]. However, most of the techniques are aimed at capturing the structural properties of the networks. To the best of our knowledge, the task of ranking nodes while simultaneously taking into account both the structure and the outcomes has not been formalized. We summarize the SNA literature and shortcomings of the traditional ranking techniques.

The notion of *service interaction networks* was introduced in [6] to capture and analyze the phenomena of agents interacting for a shared goal of value creation. In this paper, we develop a novel technique for ranking nodes that takes both the structure and outcomes into account. We first build an augmented network that varies from standard social networks by introducing special nodes corresponding to outcomes and creating appropriate directed edges (as opposed to undirected networks that are common in social network analysis). In the augmented network, the special status of the outcomes is captured by associating certain explicit values (or utilities) with the outcome nodes. We then develop an eigenvector-like method to rank the nodes of the augmented network such that the special outcome nodes are not influenced by the ranking process (as they are priors). We explore interesting mathematical and semantic properties of our technique.

We present experiments that show the progress our method makes in the direction of outcome aware ranking. We compare it against a baseline algorithm derived from traditional eigenvector ranking approach. As outlined in the conclusions, this is a limitation and we are working on other baseline algorithms for comparison. The nature and sensitivity of the business data prevents us from sharing the details of our findings in the service delivery application. In this

paper, we present our experiments and findings on real-life datasets from the Internet Movie DataBase (IMDB)¹. These datasets are such that, it is easier for wider audience to relate to, and also to replicate and verify our experiments. The instances we have been able to extract from the IMDB dataset are of moderate size. We leave the tasks of rigorous experimentation on even larger datasets and comparisons against larger set of baseline heuristics for future work.

The rest of the paper is organized as follows. In Section 2.1, we discuss related literature on node ranking in graphs. In Section 2.2, we illustrate the inadequacies of current ranking schemes, taking the interactions in Figures 1 and 2 as an example. In Section 2.3, we formally define the problem. In Section 3, we present our ranking algorithm for outcome and structure aware ranking. In Section 4 we present our experimental setup and results.

2. NODE RANKING IN NETWORKS

Ranking nodes of a network with respect to some *importance* or *influence* measure is an active research topic in the fields of social network analysis, network data mining, and in general complex networks. A node refers to an *entity* that could be an individual, role, group, or an organization. Edges encode relationships between nodes, which are of two kinds: *Persistent* (weblinks, friendship, membership, affiliations) and *discrete interactions* (e-mails, collaboration, authorship, team work). In the following, we briefly review some of the works in node ranking related to our problem.

2.1 Related Literature

Various methods exist for ranking nodes in a network. Degree Centrality is a measure in which the nodes in the network are ranked based on the number of nodes to which they are connected. This translates to ranking nodes purely based on their *experience* or *number of interactions*. Eigenvector centrality [8, 12] is a popular measure for ranking nodes in a network. It measures the centrality of a node as a linear combination of the centralities of the nodes to which it is connected. Unlike degree, which weighs every adjacent node equally, the eigenvector weighs adjacent nodes according to their centralities. Let $A = [a_{ij}]_{N \times N}$ be the (weighted) adjacency matrix of a network with N nodes. The measure x_i for node i depends on the status measures of the interacting nodes; $x_i \propto \sum_j a_{ij}x_j$. This can be expressed in matrix notation as $\lambda x = Ax$. Here, λ is the largest eigenvalue and x is the corresponding eigenvector. The agents can then be ranked based on the component wise value of the eigenvector x . The idea of using the eigenvector to do ranking dates back to the 1950’s [13, 7]. However, in recent times, this concept has been applied on a massive scale in ranking the web-pages in the HITS [8] and the PageRank [12] algorithms. A small representative sample of relevant work on node ranking with relevance to the problem considered in this paper, are presented below.

Co-authorship networks have been used extensively to determine the status of individual researchers and the structure of scientific collaboration [10, 11]. To rank the authors in a co-authorship network, *AuthorRank* algorithm was proposed in [9]. It is a modification of *PageRank* [12] algorithm for a weighted, directed network. PageRank is originally designed to rank retrieval results based on the hyperlink

¹<http://www.imdb.com/interfaces>

Illustrative Outcomes	Individual Interactions			
	C \nearrow D	A \searrow B	B $\xrightarrow{\quad}$ C	C \searrow E
Service Interactions Satisfaction Index	30	32	15	24
Co-authorship Citation Index	32	12	15	47
Movie co-stars User Rating > 8?	Y	N	N	Y

Figure 1: Individual interactions with illustrative outcomes from different communities

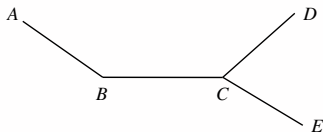


Figure 2: Interaction network of interactions from Figure 1

structure (persistent relationship) of the web, which is a directed, binary graph. The ranking is similar to eigenvector ranking where a page has high rank if the sum of its *backlinks* is high. In AuthorRank, the construction of the weights on the links between the authors is different from that of the PageRank. The co-authorship frequency and the number of authors per publication are taken into account in the network construction. Some context aware measure have also been proposed. Haveliwala [5] developed topic-sensitive PageRanks by computing several eigenvectors biased by specific topics. The rankings for a query was obtained by appropriate combination of precomputed eigenvectors. White and Smyth [14] developed an algorithm to rank the relative importance of nodes in a network with respect to a set of “root nodes”. Their technique is based on the notion of weighted paths. A node is considered to be important to a root node if they share many short paths. Chitrapura and Kashyap [3] presented a flow based algorithm to rank webpages based on their relevance to the user query. DeLong et al. [4] developed conceptual encoding of the links of the webgraph by extracting keywords and repeating phrases from anchor data of the web-pages. They exploited the augmented webgraph to present improved concept-aware page ranking. Asur et al [1] presented algorithms to find important nodes in dynamic networks by studying the node evolution and interactions. We observe that none of these works explicitly take the “outcome” of the events that give rise to the structure while ranking the nodes.

2.2 Ranking based on Structure and Outcomes of Interactions

In this section, we present some of the limitations of the previous approaches to measuring the influence or importance of the nodes of a network. Consider the example shown in Figure 1 which models agent interactions in three different applications. There are five agents A , B , C , D , and E and four interactions involving subsets of them. In case

Methodology	Ranking
Degree	$C, B, \{A, D, E\}$
Eigenvector	$C, B, \{D, E\}, A$
Outcome	C, B, A, D, E
Expected (Structure + Outcomes)	C, B, D, A, E

Table 1: Rankings based on different methodologies

of service delivery, outcome is measured based on customer satisfaction index, in case of academic collaboration, citation is assumed to be the outcome, and in case of movies, the average IMDB rating with a threshold of 8/10 is taken to be a binary outcome. The structure of their interactions is captured in Figure 2. For the sake of exposition, we consider the case of service delivery outcomes (first row of Figure 1).

We begin with the ideal or expected ranking that takes both the structural importance of the nodes and the outcomes into account.

2.2.1 Expected Ranking based on Structure and Outcomes of Interactions

Considering both structural importance and contribution to outcomes, it is imperative that C should be ranked first and B should be ranked second. In terms of structural importance, E and D have the advantage of being connected to the most important node. However, the node A has the distinction of having the highest outcome score. But, A ’s outcome score is only marginally higher than D ’s. E ’s outcome score is much lower than A ’s. So, we consider D, A, E to be the expected ordering between them that takes both the aspects into account; thus, giving the final ordering of C, B, D, A, E .

The results of known ranking methods is tabulated in Table1. Next, we explain different rankings in some detail.

2.2.2 Degree Ranking

Degree ranking merely counts the number of interactions and hence quantifies only the experience. The ranking due to degree is: $C, B, \{A, D, E\}$.

2.2.3 Eigenvector Ranking

Eigenvector ranking reflects the structural importance quite well by modeling the *inherited* or *transferred* status and implicitly takes into account the degree ranking. The largest eigenvector for the adjacency matrix of the network is 1.8477 and the corresponding eigenvector is 0.271 (A), 0.5(B), 0.653 (C), 0.354 (D), and 0.354 (E). The ranking is: $C, B, \{D, E\}, A$. With respect to this example, its limitation is exposed by the fact that it fails to distinguish between D and E which have a clear separation in terms of outcomes.

2.2.4 Outcome based Ranking

The ranking obtained by just considering the average satisfaction indices for each of the nodes is, A, D, E, B, C . This ranking completely ignores the structural importance of nodes B and C .

From the above discussion, we observe that the intended ranking mechanism should be similar in spirit to eigenvector ranking. However, it should be able to take into account value creation aspects of nodes to distinguish between nodes which seem to display similar or equal structural importance.

2.3 Problem Definition

In this section, we formalize the problem considered. Let $V = \{1, 2, \dots, N\}$ be the set of agents in the system. Let $\{1, 2, \dots, T\}$ be the set of interactions that have taken place in the system. Each interaction results in one of the possible outcomes $R = \{1, \dots, M\}$. Following indices are used: i and j for agents, t for interactions, and m for the outcomes.

An interaction $t \in \{1, 2, \dots, T\}$ involves a subset of agents $V_t \subseteq \{1, 2, \dots, N\}$. The pattern of the interaction is given by the edge set E_t with a non-negative δ_{ij}^t denoting the weight on the edge $(i, j) \in E_t$. The weight for non-existent edges is zero: $\delta_{ij}^t = 0$, $(i, j) \notin E_t$. As t is an interaction among the V_t agents, the graph (V_t, E_t) is connected. The specific structure of the graph is given by the nature of the interaction. If the interaction is that of a *group* work with one interaction involving all, then the graph is complete. For a hierarchical interaction, the graph will have a tree structure. Let $R_t \in \{1, 2, \dots, M\}$ be the observed outcome of the interaction. The interaction t can thus be completely characterized by the tuples $(V_t, E_t, \{\delta_{ij}^t\}, R_t)$. For the kind of applications of interest to us, whether it is service delivery or academic collaboration or IMDB rating, the interactions are mainly of group work.

Let $\rho_m \in \mathbb{R}$ denote the utility or value of an outcome $m \in R$. The set $\{\rho_m : m \in R\}$ can be cardinally ordered and if $\rho_{m'} > \rho_{m''}$, then the outcome m' is preferable to m'' . Typically, the utility of the outcomes is ascertained based on domain knowledge and also varies over time. Even more importantly, the user of a ranking scheme may want to experiment with different options of outcomes. So, the representation of the data and the technique should ideally allow the user to change just the outcome values and observe the corresponding changes in ranking.

Given the past t interactions $\{(V_t, E_t, \{\delta_{ij}^t\}, R_t)\}$ and the outcome utilities $\{\rho_m : m \in R\}$, goal is to rank the agents in a manner that captures both their structural importance as well as their contribution to the value creation process.

3. OUR METHOD

3.1 Network Construction

Firstly, the individual interactions are aggregated to obtain the *agent interaction network*, which, with a slight abuse of notation, is given by the graph $(V, E, \{\delta_{ij}\})$:

$$V = \cup_t V_t \quad (1)$$

$$E = \cup_t E_t \quad (2)$$

$$\delta_{ij} = \sum_t \mu^t \delta_{ij}^t \quad \forall i, j \quad (3)$$

The overall strength of interaction between any two agents is the linear combination of the individual strength of interactions. If the interactions are chronologically ordered, then μ_t can be used as the *past influence factor* to model the relative importance of the interactions with respect to time. By judiciously choosing $\{\mu^t\}$, one can model various kinds of past influences: *Uniform*, *sliding window*, etc. We call the edge weight matrix $A = [\delta_{ij}]$ of order N as the *agent interaction matrix*. This matrix aggregates the past interactions, but does not capture any information on the outcomes of the interactions. We now present a method that augments this matrix in a way that appropriately aggregates the contribution of the nodes to the value creation process.

3.1.1 Augmenting the Outcomes as Nodes

The main reason the eigenvector based approach captures the structural aspects so well is the manner in which a node transfers part of its status to its neighbors and derives its own status as a linear combination of the status of its neighbors. Consider an interaction which involves V_t nodes and have R_t as outcome. We need a mechanism by which the utility of R_t can be transferred in parts to the nodes in V_t . At the same time, the utility of the outcome R_t itself is a *prior* and should not be affected by the status of the nodes in V_t . This suggests the following natural directed construction. Let the outcome of interaction t be m . The graph $(V_t, E_t, \{\delta_{ij}^t\})$ is updated as follows:

$$V_t \leftarrow V_t \cup \{m\} \quad (4)$$

$$E_t \leftarrow E_t \cup \{(m, i)\}, \forall i \in V_t \quad (5)$$

$$\delta_{mi}^t \geq 0, \forall i \in V_t \quad (6)$$

$$\sum_{i \in V_t} \delta_{mi}^t = 1 \quad (7)$$

The outcome m is added as a node (and is called as *outcome-node*). A directed edge is added from m to each of the other agents that participated in the interaction. The weights on the newly added edges that are given by (6) and (7), captures the relative contribution of agents in realizing the outcome m . The weights on the non-existent edges are zero: $\delta_{im}^t = 0$, $\forall i$ and $\delta_{mi}^t = 0$, $\forall i \notin V_t$. The status or the influence of the outcome node is later used in the algorithm to transfer it to the participating nodes via the directed edges.

Without loss of generality, we can assume that each of the outcomes is realized in at least one of the interactions. The aggregation of the graphs augmented with outcomes is given by:

$$V \leftarrow V \cup \{1, 2, \dots, M\} \quad (8)$$

$$E \leftarrow E \cup \{(m, i) : \exists t, R_t = m \wedge i \in V_t\} \quad (9)$$

$$\delta_{mi} = \sum_t \omega^t \delta_{mi}^t, \forall i, m \quad (10)$$

$$\delta_{im} = 0 \quad (11)$$

The outcomes $\{1, 2, \dots, M\}$ are added as nodes to the interaction network and an edge from outcome m to a node i exists if i had been a part of at least one interaction with outcome m . The weights on the outcome-agent edges are taken as linear combination of the corresponding weights in the individual interactions. We call the above network as the *agent-outcome interaction network* and the corresponding edge weight matrix as the *agent-outcome interaction matrix* Δ of order $(N + M)$. For the network in figure 2 with outcome $O1$ of figure 1, the agent-outcome interaction network is shown in figure 3. The four outcomes $O1, O2, O3, O4$ correspond to the outcomes of the four interactions. Assuming equal contribution from each agent, weight on the directed edge from an outcome to an agent is $1/2$.

The matrix Δ captures both the inter-agent interactions and agent-outcome interactions. It is asymmetric. The overall intended effect of the directed construction is to let the outcome-nodes transfer their utilities to the agents and the utilities of the outcome-nodes are not altered. The matrix Δ , however does not take into account the utilities of the outcomes $\{\rho_m\}$. We treat the utilities as exogenous status of the outcomes and combine with Δ to obtain the ranking. Note that, in our representation, to experiment with

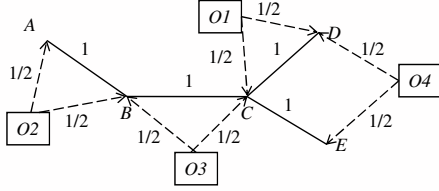


Figure 3: Interaction network with outcomes augmented as nodes

different valuations of outcomes, all we need to do is specify different e vectors.

3.2 Outcome Augmented Ranking Algorithm (OARA)

The traditional eigenvector ranking has a limitation that it can only be applied to non-negative, symmetric matrices. Thus, if the underlying graph is asymmetric, especially with *unchosen* nodes (nodes with zero in-degree), the ranking provided by eigenvector centrality is inconsistent. The unchosen nodes receive no status from the other nodes and hence contribute nothing to the nodes to which they are connected. But, in our construction the zero in-degree nodes play a critical role of transferring the status of outcomes to the nodes.

Let e be the vector that captures the status of the outcomes nodes. That is, $e[i]$ for agent nodes are made identical and for outcomes nodes, it will be a function of their values. Let x be the vector that an iterative technique computes as the final influence of the nodes. In other words, the difference vector $(x - e)$ should be expressible in an eigenvector-like form with respect to a scaling parameter α and the incident matrix Δ of the augmented network.

$$(x - e) = \alpha \Delta^T x \quad (12)$$

In other words, we are looking for an x such that,

$$x = \alpha \Delta^T x + e \quad (13)$$

Observe that the Equation (13) captures the effect of interactions and the values associated with the outcomes nodes. Our measure of influence of the nodes is similar to a centrality measure used by Bonacich [2] in finding influence of nodes in unweighted, directed graphs in very small social networks coming from marriage data among elite families. In what follows we establish some important properties of our approach.

Firstly, it is easy to see that, it reduces to the eigenvector ranking when the relationships are symmetric and the value of all the outcomes are identical (say zero). In the general setting as above, the value of x is given by

$$x = (I - \alpha \Delta^T)^{-1} e \quad (14)$$

The vectors e and x , and identity matrix I are of order $(N + M)$. We can show that the method works best when α is in the range $(0, 1/\lambda)$ where λ is the largest eigenvalue of Δ (when it exists). We omit the details due to space considerations. We now present the effect of both the free parameters of the iterative approach: α and the vector of outcome values, e .

3.2.1 Vector of Outcome Values

In this section, we first show that our usage of the vector e is semantically sound. In particular, we show that the value of outcome nodes themselves do not undergo any change during the algorithm. We then show how to set e rigorously.

Let us assume for now that we have chosen $\alpha \in (0, 1/\lambda)$. The vector e for our problem is $(e_1, e_2, \dots, e_N, e_{N+1}, \dots, e_{N+M})$, where the first N components correspond to the exogenous status of the nodes and the last M correspond to the exogenous status of different outcomes. We define e as follows:

$$e_i = 1, \forall i \quad (15)$$

$$e_{N+m} = \theta \rho_m, \forall m \quad (16)$$

The assignments in Equation 15 reflect the fact that the exogenous status of all the agent nodes are uniform. The outcomes on the other hand have their utilities as the external status. The utilities are unique up to a linear transformation and hence the external status of an outcome is defined as proportional to its utility by a non-zero scalar θ . In other words, $\{\theta \rho_m\}$ still preserves the cardinal structure and the relative magnitude of the utilities are unchanged.

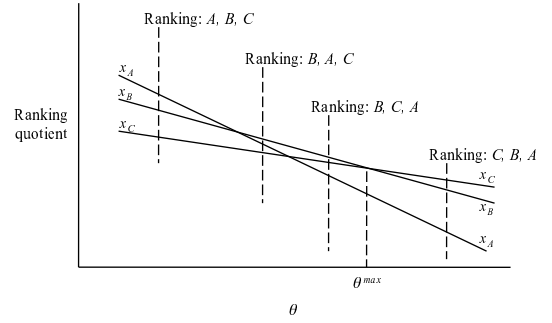


Figure 4: Rank quotients of vertices as a function of θ

The justification for associating non-uniform exogenous status with the outcome nodes is: *the utilities of the different outcomes are independent of the way they are achieved*. For example, the utility of a research article is not determined by the authors of the paper, rather it is decided by its impact, say the number of citations it gets. Therefore, while the exogenous status of the outcome-nodes should influence the status of the other nodes, the solution x of Equation 14 should not alter the status of the outcome nodes. The following derivations show this property to be indeed true. Define matrix Y of order $(N + M)$:

$$Y = (I - \alpha \Delta^T)^{-1} \quad (17)$$

By definition of Δ ,

$$y_{N+m, N+m} = 1, \forall m \quad (18)$$

$$y_{N+m, N+m'} = 0, \forall m, m' \neq m \quad (19)$$

Rephrasing x vector in terms of Y ,

$$x_i = \sum_j y_{ij} e_j + \sum_m y_{i, N+m} e_{N+m}, \forall i \quad (20)$$

$$x_{N+m} = e_{N+m}, \forall m \quad (21)$$

α	Ranking
0.054, 0.102, 0.15, 0.198	C, B, A, D, E
0.247, 0.295, 0.343	C, B, D, A, E
0.391, 0.439, 0.487	C, B, D, E, A

Table 2: Ranking for different α values

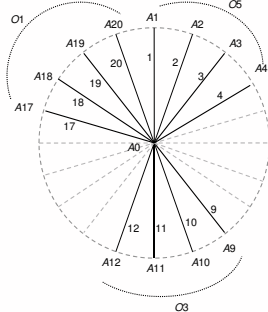


Figure 5: A star interaction network with 21 agents and 5 outcomes

Substituting for the e , as defined in (15) and (16),

$$x_i = \sum_j y_{ij} + \theta \left(\sum_m y_{i,N+m} \rho_m \right), \forall i \quad (22)$$

$$x_{N+m} = \theta \rho_m, \forall m \quad (23)$$

Thus the components of x corresponding to outcome nodes have the same value as their external status. On the other hand, x_i s (called *ranking quotients*) for the other nodes are dependent on θ and rankings could possibly change with respect to the choice of θ . As the other terms in (22) are constants, we have N lines given by equations (22) as a function of θ . If all these lines are parallel, then ranking is independent of θ . Otherwise, the rankings will depend on θ as shown in figure 4. The ambiguity in ranking is due to the magnitudes of Δ , θ , and $\{\rho_m\}$. The Δ and $\{\rho_m\}$ are instance specific and hence θ should be chosen such that ranking is not affected. One possibility is to choose a positive $\theta > \theta^{max}$ (greater than all the intersecting points, as shown in figure 4), so that the ranking will remain unchanged with further increase in θ value. As each line will possibly intersect at $N - 1$ points with $N - 1$ lines, the total number of possible intersecting points are $(N - 1)(N - 2)/2$. Hence, the maximum of the intersection points, θ^{max} can be estimated in $O(N^2)$ time.

3.2.2 The α Value

Another free parameter that affects the ranking is $\alpha \in (0, 1/\lambda)$. The parameter α reflects the relative importance of the structure versus outcomes in determining the rank of the agents. For the network in Figure 3, the largest eigenvalue of the Δ matrix is 1.8477 and hence α was chosen in the range $(0, 1/1.8477)$. The table 2 shows the rankings obtained for ten different equally spaced α values in the above range. The ranking given by mid-range α values is that of the expected ranking, whereas the low range is same as that of outcome rank and the higher range is more similar to eigenvalue and degree rank (also see table 1). In the fol-

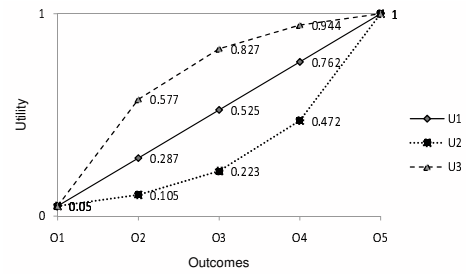


Figure 6: Three different utilities

lowing, we propose a novel way of analyzing the influence of outcomes and interactions over the ranking and thereby choosing α judiciously.

Let an $\bar{\alpha} \in (0, 1/\lambda)$ be chosen. Let R be the ranking for the utility $\{\rho_m\}$. Construct a new set of utility $\{\mu_m\}$ with the opposite order of preference. Thus, if the outcome has the highest utility in $\{\rho_m\}$, then it has the lowest in $\{\mu_m\}$. Similarly, the outcome with second highest in $\{\rho_m\}$ is one but last in $\{\mu_m\}$. Let \bar{R} be the ranking with utility $\{\mu_m\}$ and $\bar{\alpha}$. For a given value of $\bar{\alpha} \in (0, 1/\lambda)$, we would like to estimate the influence of structure and outcomes in the final rankings. We do this by exploiting the concept of Kendall correlation [7]. Given two complete rankings, it computes a score in the range of $[-1, 1]$. If the score is close to 1, it means the two rankings are identical and if it is close to -1, it means the two rankings are near opposite orders. If the Kendall correlation, $\tau(R, \bar{R})$ is close to 1, then the outcomes have negligible influence, as the rankings are not modified with the reversed utilities. Similarly, if the correlation coefficient is close to -1, one can infer otherwise. An equal influence of outcomes and interactions is achieved for an α with which zero correlation is achieved. Such an α may not exist for a given interaction network, as shown in the following example.

Consider the star interaction network of figure 5. The agents $A1, A2, \dots, A20$ had interacted with agent $A0$. The strength of interactions with $A0$ are $1, 2, \dots, 20$ for agents $A1, A2, \dots, A20$, respectively. Five outcomes $O1, O2, \dots, O5$ were observed. As shown in the figure, all the individual interactions of $A0$ with $A20, A19, A18$, and $A17$ had resulted in $O1$. Similarly, interactions with the next four agents in sequence resulted in $O4$ and so on. Each interaction has equal contribution from $A0$ and the interacting agent for the resulting outcome. Hence, the weight of the edge from outcome node $O1$ to $A20$ is 10 (half of 20 interactions) and that of to $A0$ is 37 (half of $20 + 19 + 18 + 17$). For brevity, the network augmented with outcome nodes and edges are not shown in the figure.

Figure 6 shows three different utility functions $U1, U2$, and $U3$ for the outcomes. The utilities are scaled to range $(0, 1]$ as they are unique up to a multiplication by a positive scalar (θ^{max} ensures the uniqueness). All the three functions are increasing over $O1, O2, O3, O4$, and $O5$. The rate of increase, however, is different: *linear* for $U1$, *marginally increasing* for $U2$, and *marginally decreasing* for $U3$. Thus, one can expect the difference in rate of increase to be reflected in the rankings. Let the utilities $U1, U2$, and $U3$ be reversed. For example, reversed utilities of $U1$ for $O1, O2, O3, O4$, and $O5$ are 1, 0.762, 0.525, 0.287, and 0.05. It is easy to verify that the ranking for the reversed utilities of

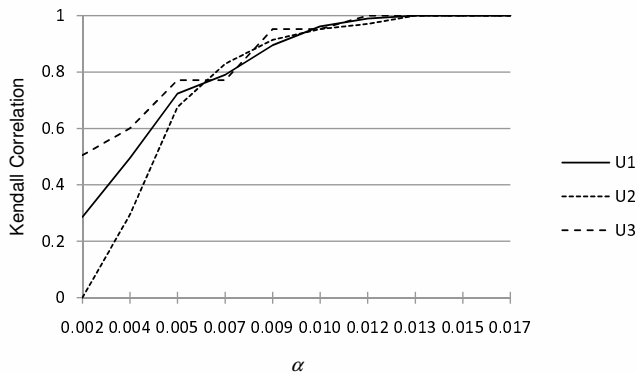


Figure 7: Trade-off of interactions and outcomes for utilities U1, U2, and U3

U1, U2, and U3, for all $\alpha \in (0, 1/\lambda)$ will be A0, A20, A19, ..., A1. Incidentally for this network, this ranking is also the same as that of degree based ranking and eigen ranking. Figure 7 shows the variations in Kendall correlation coefficient for the rankings with opposite utilities, with respect to $\alpha \in (0, 0.018)$ (the maximum eigen value $\lambda = 53.572$). As expected, the correlation is increasing towards 1. For all the three utility functions, the minimum possible correlation coefficient is positive and hence influence of the structure of agent interactions is considerably more than that of outcomes. For U2, Kendall correlation coefficient of zero is obtained for $\alpha = 0.02$ and a ranking obtained for this α takes into account both the interactions and the outcomes in equal strengths. Note that the above discussions are for illustrative purposes and the inferences (with respect to U1, U2, and U3) are limited to the given star network. However, the proposed methodology of reversing the utilities and computing the correlation of the rankings for all $\alpha \in (0, 1/\lambda)$ enables one to analyze the trade-off of influence of outcomes and interactions. As it needs reversal of utilities and the rankings are derived from the same network, there are no anomalies in the trade-off analysis and requires no additional routines for the algorithm. Based on our analysis of the impact of α for the star network and for the network shown in Figure 3, we follow a heuristic of picking α in the lower half of the range of $(0, 1/\lambda)$ to give a slight bias towards outcomes.

3.3 Baseline Algorithm for Comparison

As shown during the problem formulation naive application of known ranking techniques to our problem gives unsatisfactory results as they were not designed keeping these requirements in mind. However, as observed during the formulation, several characteristics of the eigenvector ranking are essential in our setting too. So, a natural question, can a slight modification to the eigenvector ranking give meaningful rankings in our context?

Consider the matrix Δ obtained from the special construction described in Section 3.1.1. This matrix is an asymmetric matrix. This is inevitable as the outcomes play a special role. So, we cannot directly use eigenvector computation which is meant for symmetric matrices. So, a natural heuristic is to consider the ranking according to the left singular vector corresponding to the largest singular value of Δ . In this case, Δ is modified as follows: if there is an edge of

weight w from an agent node i to an outcome node of value v , the corresponding Δ entry is set to be $w \cdot v$. Intuitively, this heuristic does not address the subtle technical point we covered in depth: that of ensuring that the outcomes are used to influence the ranking of agents and not vice versa. Our experiments show that this approach produces results which are inferior to the ones obtained by our technique.

4. EXPERIMENTS

In this section, we present experimental results that show the efficacy of our approach. In the service delivery application for which we developed our approach, ranking is used in conjunction with the roles and responsibilities of the agents to determine ideal team composition for new service requests. Since an automated tool cannot capture all the nuances of the service delivery, the output of the tool is used as a guidance rather than in a strict sense. This, and the sensitivity of the business data, makes it difficult to report our field trials in a consistent way. Moreover, it is difficult to replicate and verify without sharing the data which reveals certain statistical properties pertaining to the service delivery organization. So, for the purpose of verifiability and replicability, we conduct and report our experiments on the datasets from IMDB.

We consider the problem of ranking movie actors. Typical SNA approach would rank the agents based on the ‘‘co-actor’’ network structure. Typical outcome oriented ranking would rank the actors based on the average ‘‘outcome’’ of their movies (with appropriate statistical filters). However, as argued in the introduction, our approach of simultaneously considering both the structure and the outcomes captures the dynamics better (as demonstrated in the experiments as well). We view each movie as an interaction involving the leading actors of the movie. The outcome of a movie should be indicative of its success or failure. We choose the IMDB dataset as it provides us with an appropriate proxy for the outcomes. Arguably, the outcome of a movie is impacted quite a lot by participants such as the director and rest of the technical team. However, in a mass rating medium like IMDB where a single movie gets thousands of votes, it is the performance of the main actors which influences its rating to a great extent. So, for a movie, we consider its overall user rating in the IMDB as its outcome. To ensure that our assumptions hold, we consider only those movies that have a certain minimum number of votes. It is also common to see multiple actors sharing screen in multiple movies which is common with the features of the service delivery application that motivated our work. Another advantage with the IMDB dataset is, we can use familiar, identifiable names in small experiments and conduct controlled acts like changing the outcomes of their movies or including/excluding specific movies (or time periods) to show the conceptual strength of our formulation.

IMDB allows us to programmatically extract the following data: list of all movies, list of movies according to genre and time period, within a genre top user rating movies, within a genre a random mix of movies and so on. We also get the user ratings corresponding to the movies in the lists that we work with. Also, for each movie, it lists the list of actors in the movie. So, for a given set of movies, we construct the interaction network as follows: each actor who appears in any of the movies is an agent, each movie represents an interaction that is incident on all its main actors (we choose

first 6 listed names) and the outcome associated with the interaction is the average user ratings for the movie. Given this interaction data, we construct the Δ matrix as described in Section 3.1.1. Our experimentation explores the effects of the different constructions for the vector outcome values. As explained in last section, we choose α in the lower range of $(0, 1/\lambda)$. Some of the important conceptual aspects that we exhibit from our experiments are:

- The baseline algorithm described in Section 3.3 is not very sensitive to the changes in the outcomes.
- When the outcomes associated with a certain subset of movies is changed, the rankings of the actors obtained by OARA also undergoes corresponding change, thus indicating the sensitivity of our technique for changes in outcomes.
- When we increase/decrease the importance of a subset of actors, then, in the ranking obtained by OARA, the ranking of the actors who are well connected to those actors also see a corresponding change. This indicates that our approach takes the structure into account while ranking the agents.
- The choice of e is important to obtain good results. We demonstrate some effective and some not so effective constructions for e .

The IMDB dataset, due to the presence of large number of widely recognizable actors makes it easy to verify intuitive correctness of the experiments on small datasets. We have extracted two special instances (*List 1* and *List 2*) each having 28 and 30 actors respectively (small enough to be easily interpreted and large enough to need computation) which can be used to highlight conceptual strengths. *List 1* and *List 2* are enumerated in Table 3. The first instance consists of 28 actors from contemporary times. The second instance consists of 30 actors whose prime era was before 1970s. Since these two sets have fairly well known actors, we make our conceptual points using these instances.

Results of ranking algorithms are generally meaningful only for connected interaction networks. When the graph is disconnected, the rankings across different components are not comparable. Therefore, we limit our instances to only connected interaction networks. We employ simple traversal techniques over the movie and actor lists to construct instances which are connected. To demonstrate the stability of the computational approach we construct interaction networks with 200, and 400 actors (11114 movies). To automatically assess the sensitivity of our technique for changes in outcomes, we use the Kendall *tau* distance (τ from now) between the following two rankings: before and after changes in the outcomes. As for the feasibility of our approach for very large graphs, it should be noted that the computation of an inverse is the most time consuming part; $O(n^3)$. Therefore, for most interaction networks like the completely cleaned, connected IMDB or DBLP, our approach can be used without any problem. For truly massive graphs like the web graph, exact computation is not feasible. Developing a technique that can be used even on massive graphs is an important challenge for future work.

Actor Index	List 1	List 2
1	Marlon Brando	Marlon Brando
2	Al Pacino	James Mason (I)
3	Robert De Nero	Louis Calhern
4	Sean Bean	Glenn Ford (I)
5	Jean Reno	Karl Malden
6	Don Cheadle	Ben Johnson (I)
7	John Travolta	Timothy Carey
8	Hugh Jackman	Richard Harris (I)
9	George Clooney	Montgomery Clift
10	Casey Affleck	Dean Martin (I)
11	Brad Pitt	Frank Overton
12	Matt Damon	Malcolm Atterbury
13	Dan Fredenburgh	Robert Ryan (I)
14	Bill Nighy	Burt Lancaster
15	Johnny Depp	Frank Sinatra
16	Orlando Bloom	Ernest Borgnine
17	Jack Davenport	Lee Marvin
18	Lee Arenberg	Rhys Williams (I)
19	Tom Hollander	DeForest Kelley
20	Jude Law	John Wayne (I)
21	Anthony Hopkins	Walter Brennan
22	Sean Penn	Ed Wynn
23	Samuel L. Jackson	Stephen Boyd (I)
24	Kevin Bacon	Milton Berle
25	Tom Hanks	Tony Bennett (I)
26	Steve Buscemi	Al Pacino
27	Clive Owen	Robert De Niro
28	Nicolas Cage	Broderick Crawford
29		Ricky Nelson (I)
30		Buddy Ebsen

Table 3: List of actors used in experiments

4.1 Experiments when outcomes are changed

The goal of these experiments is to evaluate the ability of our OARA approach and SVD to take into account outcomes while ranking the actors. We conduct our experiment as follows. Let Δ be the matrix constructed according to the method in Section 3.1.1. Let e be the vector of outcome values. We generate the OARA and SVD rankings under four different conditions as defined below:

- R_1 is the ranking by OARA(Δ, e). Table 4 and Table 7 show the R_1 ranking for both the list of actors.
- R_2 is the ranking by SVD(Δ).
- R_3 is generated after making the following modifications to the original data. We pick two highly ranked actors in both the rankings, say A_1 and A_2 . For each of the movies in which either of them appears, we artificially reduce the averaging rating by 2. We then pick the two middle-ranked actors, A_3 and A_4 . We increase the ratings of those movies in which either of them appears by 2. Let Δ' be the corresponding matrix. Note that e does not change. Let R_3 be the ranking by OARA(Δ', e).
- R_4 is the ranking obtained by SVD(Δ').

For the rest of the experiments we use R_1, R_2, R_3, R_4 to denote the rankings obtained by OARA and SVD by this process for the different lists and different e s considered in the experiments. The different rankings obtained for List 1 are as in Table 4 (With $A_1 = 9, A_2 = 23, A_3 = 28, A_4 = 16$).

$R_1 = \text{OARA}(\Delta, e)$	23 9 25 12 10 7 21 15 3 6 20 8 24 16 28 2 22 26 27 11 1 4 14 18 5 17 19 13
$R_2 = \text{SVD}(\Delta)$	9 23 25 12 7 10 21 15 3 20 8 6 24 28 16 2 22 26 27 11 4 1 14 5 17 18 19 13
$R_3 = \text{OARA}(\Delta', e)$	25 28 23 9 12 10 7 21 15 3 16 20 6 8 24 2 22 26 27 11 14 1 4 18 17 5 19 13
$R_4 = \text{SVD}(\Delta')$	9 23 25 12 7 10 21 15 3 20 8 6 24 28 16 2 22 26 27 11 1 4 14 5 17 18 19 13

Table 4: Rankings of actors in List1 under different conditions

$\text{OARA}(\Delta_{list1}, e)$	23 9 25 12 10 7 21 15 3 [*] 6 20 8 24 16 28 2 [◇] 22 26 27 11 1 4 14 18 5 17 19 13
$\text{OARA}(\Delta_{list2}, e)$	17 12 24 30 19 28 16 20 15 18 22 10 11 25 2 21 6 13 4 29 23 14 8 7 9 5 1 3 27 [*] 26 [◇]

Table 5: Rankings of Al Pacino[◇] and Robert de Niro^{*} in both the lists

The rankings are given in the ascending order of ranks; the actor with the first rank appears first and so on.

Let us now check how the two methods dealt with changes in outcomes (refer Table 4). One would expect the rankings of A_1 and A_2 to go down and those A_3 and A_4 to go up. Notice (by comparing R_1 and R_3) that OARA rankings of the both top actors (number 23 and 9) have gone down while those of the two chosen mid-ranked actors A_3 and A_4 (number 28 and 16) have gone up. Notice (by comparing R_2 and R_4) that there is hardly a noticeable change in the SVD rankings before and after modification. This shows that our formulation is more sensitive to changes in outcomes.

4.2 Experiments for relative importance across time periods

We conducted experiments to check if our approach can indicate relative influence of actors across time periods (Refer to Table 5). Since the List 2 involves mainly old actors, only movies up to 1980 are contained in their interaction networks. We consider two actors, Robert De Nero and Al Pacino who are present in both the lists. Since their interaction with the old actors is limited, they are ranked towards the end in List 2 whereas they are ranked high in List 1.

4.3 Experimentation with e

The vector of outcome values plays a crucial role in the ranking obtained by OARA. Conceptually, there are two main settings. One, in which every outcome has some positive value (degree varies) and the other, in which some of the outcomes may have negative values. As an example of a setting with negative outcomes is IMDB where a user rating of 7 or below is treated as an indication of its failure and a rating above 7 is treated as an indication of its success. In such a case, it is meaningful to associate positive and negative weights with different outcomes. An example of setting of only positive outcomes whose degree varies is the academic publications by researchers in refereed conferences and journals. The value varies depending on the forum and citations and so on. So, in this case, the weights associated with outcomes are appropriately weighted positive values (for example, 100 citations is more than 10 times better than 10 citations!). Below, we show how different ways of setting the vector e influences the rankings.

Case 1. When all outcomes were positive. In this case, every movie is treated as having a non-zero positive outcome. The value of a movie with rating R the outcome was set to be 2^R . Refer to Table 6. Here, ranking R_1 is the ranking on original data. Ranking R_3 is obtained by reducing

$R_1 = \text{OARA}(\Delta, e)$	17 24 12 30 20 28 16 19 15 10 22 18 25 11 2 21 6 13 4 23 29 14 8 7 9 5 1 3 27 26
$R_3 = \text{OARA}(\Delta', e)$	17 24 12 30 20 28 16 19 15 10 18 22 21 25 2 11 6 13 4 23 29 14 8 7 5 9 1 3 27 26

Table 6: Rankings of actors in List2 generated for Case 1

$R_1 = \text{OARA}(\Delta, e)$	17 24 12 30 28 19 16 20 15 10 22 18 25 11 2 21 6 13 4 23 29 14 8 7 9 5 1 3 27 26
$R_3 = \text{OARA}(\Delta', e)$	15 24 30 20 21 10 16 17 25 12 28 19 22 18 2 11 6 13 4 14 29 5 23 8 1 27 9 7 26 3

Table 7: Rankings generated of actors in List2 for Case 2

the outcomes of a subset of the movies of actors 17 and 24. In this case, although OARA shows changes in some of the connected actors like 21 and 11, it does not change the ranking of 17 and 24 themselves as they had other movies with high positive value and since the values of movies with reduced rating was still positive, they did not suffer in ranking (contrary to expectation).

Case 2. When the outcomes are mixed. This is the most general case where the outcomes are treated as positive and negative. For example, a movie whose average rating is 5 is deemed a failure and thus having a negative outcome, and a movie with a rating of 8 is considered to be a positive outcome. Specifically, we set the weights for different outcomes as follows. If a movie has the corresponding rating R greater than 7 then the associated was 2^R , else the outcome is -2^{8-R} . We repeated the experiment for Case 1 with this setting (Refer to Table 7). Observe that R_1 and R_3 were noticeably different. Note that relative ranks of 17 (A_1) has dropped while that of 21 (A_4) has improved. The reason for the stability of 24 in the ranking is his close collaboration with actors 2 and 11 with whom he has acted in 14 movies. Table 7 shows the ratings from experiment on List 2.

Case 3. Vector e can be used to take into account any known special status of the agents themselves. In this case, the status of an actor may be enhanced due to an award or a rise in popularity. In this case, non-uniform setting of $e[i]$ for nodes corresponding to agents to take this into account. Observe original ranking for List 2. Actors with indices 27 (Robert De Nero) and 26 (Al Pacino) are appearing towards end in R_1 . If we now increase the importance of these two actors (because of award winning performances) and keep the importance of rest of the actors uniform, then, we notice that not only these two actors, but, also those who interacted with them (like Marlon Brando) benefit in the ranking. The new ranking after changing importance of the two actors is denoted by R_5 . Rankings are shown in Table 8.

4.4 Experiments on larger networks

We extracted larger, connected networks consisting of 200 and 400 actors respectively. The fact that we used only the first few of the listed actors for each movie limited the size of the connected networks that we could detect. We use Kendall correlation [7] to see how rankings are correlated (Refer to Section 3.2.2 for interpreting this measure) across different experiments. As before, we decrease the value of

$R_1 = \text{OARA}(\Delta, e)$	17 24 12 30 28 19 16 20 15 10 22 18 25 11 2 21 6 13 4 23 29 14 8 7 9 5 1 3 27 26
$R_5 = \text{OARA}(\Delta, e')$	27 26 24 25 20 17 28 30 16 15 10 12 22 2 19 14 29 18 6 4 11 21 13 1 8 23 7 9 5 3

Table 8: Rankings generated of actors in List2 for Case 3

outcomes of two top actors and increase the outcome of two middle ranked actors. This process affects a large fraction of the nodes in the network because of the cascading effect of the “co-actor” network structure. As a result, the rankings of most actors are likely to see major shifts, and hence, the two rankings are likely to look almost unrelated. We notice that, for the OARA rankings, the Kendall correlation measure is close to zero and for the SVD rankings it is in the range of 0.6. These experiments provide important empirical evidence that our method accounts both the structure and the outcomes in its rankings; and, the SVD based approach is not as effective.

Case 1. Network of 200 actors. $\tau(R_1, R_3) = 0.007352$ and $\tau(R_2, R_4) = 0.68160$

Case 2. Network of 400 actors. $\tau(R_1, R_3) = 0.087$ and $\tau(R_2, R_4) = 0.6345$

4.5 Experiments on Simulations of Service Delivery

This research was conducted for the application of ranking agents in a service delivery setting. Although we are not able to share results on the actual data, we present results on data obtained from a service delivery simulator that we developed. Our simulator captures various real-life, people-centric aspects of service delivery that we observed in practice: a. compatibility and incompatibilities between people within team settings and their effect on the outcomes b. transient slackness in performance of people due to repetitive nature of tasks in service delivery c. performance aspects such as ramp-up curve when people begin their work. Here, we briefly discuss our experiments with the simulator. A reference to the technical report on our simulator will be provided in the final version of the paper.

Our simulator created an environment of 500 agents. Associated with each agent was a list of compatible and incompatible agents, apart from individual level of competency. We generated a time-series of requests tagged by types. We used the types of the recent requests performed by an agent to measure his likelihood of slackening. For a given request, we picked agents so as to balance load (a common practice in service delivery). For each assignment of agents, we computed a likely-outcome that takes into account: their individuals levels of competency, their affinity for slackening, and basic compatibility index between them as a team. We repeated the experiments carried out in Section 4.4 on the simulated data. With the usual meanings of rankings R_1, R_2, R_3, R_4 , over multiple runs, we found that $\tau(R_1, R_3) \approx 0.30$ and $\tau(R_2, R_4) \approx 0.75$. Note that the Kendall measure between R_1 and R_3 in case of simulation is much higher than those obtained in Section 4.4. This could be because, the outcomes are computed by a method which combines the parameters of the simulation in a predictable manner whereas the outcomes of the movies do not necessarily follow any predictable model.

5. CONCLUSIONS

The main goal of our work is to highlight applications that require ranking that takes the structure and outcomes associated with the interaction networks. We developed a novel ranking technique and presented several mathematical properties of the technique. Our technique is sensitive to changes in both structural and outcome aspects of the interaction networks. There are several directions to pursue.

One of the drawbacks of our current empirical work is the comparison against a single baseline heuristic. There are other ways of modifying the interaction networks that take the outcomes into account. For example, one may rank the actors separately based on structure and outcomes and then use rank aggregation techniques to get a combined ranking. We are currently working on this approach. From a computational point of view, the need to invert a matrix is the most challenging task of this approach. To make this scale to massive graphs, we need to develop a heuristic which approximates the matrix computation in the same sense as power method does for eigenvector computation. Conducting our experiments on massive real-life datasets is another focus of our current work.

6. REFERENCES

- [1] S. Asur, S. Parthasarathy, and D. Ucar. An event-based framework for characterizing the evolutionary behavior of interaction graphs. In *KDD*, pages 913–921, 2007.
- [2] P. Bonacich and P. Lloyd. Eigenvector-like measures of centrality for asymmetric relations. *Social Networks*, 23:191–201, 2001.
- [3] K. P. Chitrapura and S. R. Kashyap. Node ranking in labeled directed graphs. In *CIKM*, pages 597–606, 2004.
- [4] C. DeLong, S. Mane, and J. Srivastava. Concept-aware ranking: Teaching an old graph new moves. In *ICDM Workshops*, pages 80–88, 2006.
- [5] T. H. Haveliwala. Topic-sensitive page rank. In *In the proceedings of the World Wide Web (WWW) Conference*, pages 517–526, 2002.
- [6] S. Kameshwaran, S. Mehta, V. Pandit, G. Parija, S. Singh, and N. Viswanadham. Analyses for service interaction networks with applications to service delivery. In *the Proceedings of SIAM International Conference on Data Mining (SDM)*, 2009.
- [7] M. G. Kendall. Further contributions to the theory of paired comparisons. *Biometrics*, 11:43, 1955.
- [8] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.
- [9] X. Liu, J. Bollen, M. L. Nelson, and H. V. de Sompel. Co-authorship networks in the digital library research community. *Information Processing and Management*, 41:1462–1480, 2005.
- [10] M. Newman. Scientific collaboration networks: I. Network construction and fundamental results. *Physical Review*, 64, 2001.
- [11] M. Newman. Scientific collaboration networks: II. Shortest paths, weighted networks, and centrality. *Physical Review*, 64, 2001.
- [12] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web, 1998.
- [13] T. H. Wei. *The algebraic foundations of ranking theory*. Cambridge University Press, London, 1952.
- [14] S. White and P. Smyth. Algorithms for estimating the relative importance in networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 266–275, 2003.