On Principles of Egocentric Person Search in Social Networks

Sara Cohen
Dept. of Computer Science and Engineering
Hebrew University of Jerusalem
sara@cs.huji.ac.il

Benny Kimelfeld Georgia Koutrika Jan Vondrák
IBM Research–Almaden
San Jose, CA 95120, USA
{kimelfeld,gkoutri,jvondrak}@us.ibm.com

ABSTRACT

Person search is the problem of finding, by means of keyword search, relevant people in a social network. In *egocentric person search*, the search query is issued by a person s participating in the social network, and the goal is to find people that possess two qualities: relevancy to the query, and relevancy to s herself. This position paper considers the latter quality, and specifically, scoring functions that rank persons by their relevancy to s. In particular, the paper proposes general principles (i.e., properties) that should be held by such scoring functions. Several functions, which were proposed in the past for measuring node connectivity, are analyzed with respect to the proposed principles. It is shown that none of these functions sufficiently satisfy the principles. In contrast, the paper presents two additional functions that satisfy the principles in a strong sense.

1. INTRODUCTION

Online social networks have grown in popularity at an extraordinary pace over the last few years. In fact, social networks, such as Facebook, MySpace and Twitter, have become so widespread that they currently boast hundreds of millions of users. The graph structure defined by a social network encodes interesting and useful information about the social relations between users. Leveraging this data to effectively answer different types of queries is an interesting and challenging problem.

Abstractly, a social network is simply a graph of people. Directed edges indicate that one person (node) likes/trusts/recommends another. (We use a directed model, as in Twitter, to model possibly asymmetric relations.) In addition, each node is associated with textual data, such as personal information, posts, etc.

Social networks have been the focus of extensive research, studying metrics like centrality and cohesion, as well as phenomena like the small world property. See [5] for a history of the development of social network analysis. More recently, online social networks have been studied in the context of topics such as social search [3, 9, 10] and link prediction [8].

The focus of this paper is on *egocentric person search*. *Person search* is the problem of finding, by means of keyword search, relevant people in a social network. Person search is an important type of query over a social network, as it is an aid in finding people of interest. In *egocentric person search*, the search query is issued by a person s participating in the social network, and the goal is to find people that possess two qualities: relevancy to the query, and relevancy to s herself. This position paper considers the latter quality, and specifically, scoring functions that rank persons by their relevancy to a given node s. Results that are highly ranked by relevancy to the query poser s are people (transitively) trusted by s. Hence, s can be less wary of entering into a real-life relationship (social or otherwise) with these people.

Suppose, for example, that the searcher s is node Sally in the small fragment of a social network in Figure 1, and she poses the query "oral surgeon" (or "car mechanic", "immigration lawyer", "really nice guy"). Obviously, her goal is to find a person satisfying the query, who is also trusted or recommended by people who she trusts. Assuming that nodes Tim, Ted and Tony are relevant to the keywords, our goal is to measure their relevancy to s by taking into consideration the graph structure.

Egocentric person search highly differs from social (web) search [3, 9, 10]. The latter generally refers to the problem of *ranking web pages* while taking into consideration social relations. In contrast, the former problem is that of *ranking social network nodes*. The problem studied in this paper bears similarity to *expert search* [4]. However, the latter has not taken into consideration the egocentric aspect of this problem. Another different yet related problem is efficient search within a social network [1, 2, 13]; there, the focus is usually on efficiently finding nodes with given properties (and not on sophisticated scoring functions). Link prediction, which is the problem of predicting which social relations are likely to be added to a social network [8], is also highly related. (This relationship is discussed further in Section 4). Intuitively, egocentric person search differs from link prediction in that we must rank nodes t who are relevant to the search keywords, even if it would a priori seem unlikely for s and t to form a social relation.

As mentioned previously, when ranking results of an egocentric person search, one must measure relevancy to the query, and relevancy to s herself. The former can be quantified using standard information retrieval ranking functions. Thus, the focus of this work is on the latter. As an aid to developing and studying node scoring functions, we present

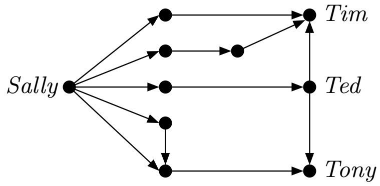


Figure 1: Small fragment of a social network

several properties that seem intuitively appealing, which we expect to hold in any natural node scoring function. These properties are given in the form of graph manipulations, and how they effect scoring of nodes. Intuitively, the underlying assumptions of our properties are, all else being equal, *(1)* a node closer to s should score higher than one farther from s, and *(2)* a node with multiple independent paths from s should score higher than one with fewer paths. For example, in Figure 1, it would seem that Tim is more relevant to Sally than Tony, who in turn is more relevant than Ted. All three nodes being at distance 2 from Sally, there are more independent paths from Sally to Tim than there are to Tony (and more to Tony than to Ted). Note that independent paths translate, in the real world, to independent opinions, and hence, are quite valuable.

After presenting our general principles (properties), we consider several node scoring functions that have been mainly used in the context of link prediction. We analyze the degree to which these functions satisfy our properties, and show that this degree is insufficient: each of these functions satisfies one or more properties in a trivial manner, or even violates them. We then introduce two additional node scoring functions, *expected distance* and *reliability*, and show that these functions satisfy all properties in a strong sense.

2. EGOCENTRIC PERSON SEARCH

A social network is a directed graph $G(V, E)$, where V is a set of nodes, called people, and E is a set of edges. We use a directed model to take into consideration asymmetric social relations, as in Twitter. Thus, an edge (u, v) indicates that u views v in a positive light, that is, u likes/trusts/recommends v. In addition, each person v is associated with textual content, such as personal information, posts and so forth. A small fragment of a social network appears in Figure 1. Note that most textual content has been omitted in this figure for simplicity in presentation (we only show the first names of the nodes).

In egocentric person search, a keyword search query is issued by a person s in the network, and the result is a ranked list $t_1, \dots, t_k$ of nodes. Abstractly, we denote a query as a pair (s, c), where s is the node initiating the query, and c is a string of keywords. For example, the query "tax consultant", issued by Sally in Figure 1, will result in a ranked list of nodes, relevant to the keywords, from the graph.

The main challenge in egocentric keyword search is to formulate an effective scoring function for results of a query (s, c). Clearly, the score of a node t should take into consideration two aspects. First, how relevant are the keywords c to t? Second, how relevant is t for s? For the former aspect, standard information retrieval scoring mechanisms can be used. Therefore, we focus on the latter aspect. Formally, we will consider functions $\text{score}(s, t, G)$ that measure the relevance of t for s in graph G. Relevance of t for s is important in a social setting; in particular, it measures how much s (her friends, friends of friends, and so forth) like/trust/recommend t. In the following section we present three simple properties that any $\text{score}(s, t, G)$ should satisfy.

3. SCORING FUNCTION PROPERTIES

In this section, we fix a graph G and three distinct nodes s, t, and v of G. We denote by $\pi^v_{s,t}(G)$ the graph that consists of all the simple paths from s to t through v. We are interested in three special types of nodes that are on paths from s to t: *connectors*, *mergers*, and *splitters*. Each of these plays a special part in G, and hence, manipulating such nodes, will yield a new graph, for which we will expect that s and t will be even more closely related.

We say that v is a *connector* if v lies on a path from s to t, v has a single incoming edge (u, v), a single outgoing edge (v, w), and there is no edge from u to w. Intuitively, v is a connecting link on a path from s to t, and has no additional interplay with the graph. We say that v is a *merger* if v has multiple incoming edges that are on simple paths from s to t, v has a single outgoing edge, and v separates s from t in the graph $\pi^v_{s,t}(G)$. Intuitively, merger nodes serve as a merging point of multiple paths from s to t. Finally, v is a *splitter* if v has a single incoming edge, multiple outgoing edges that are on simple paths from s to t, and v separates s from t in the graph $\pi^v_{s,t}(G)$. Thus, v can be thought of as splitting an incoming path from s into multiple diverging paths to t.

Our properties use three types of graph transformations, as depicted pictorially in Figure 2 and formally defined next. Let v be a node in G.

1. If v is a connector, with incoming edge (u, v) and outgoing edge (v, w), then $\mathsf{shorten}_v(G)$ denotes the graph that is obtained from G by removing v, and adding an the edge (u, w). (See Figure 2(a).)[1]

2. If v is a merger with outgoing edge (v, w) and incoming edges $(u_1, v), \dots, (u_n, v)$, then $\mathsf{unmerge}_v(G)$ denotes the graph that is obtained from G by removing v, and for all $i \in \{1, \dots, n\}$ adding a new node v_i and the edges (u_i, v_i) and (v_i, w). (See Figure 2(b).)

3. If v is a splitter with incoming edge (u, v) and outgoing edges $(v, w_1), \dots, (v, w_n)$, then $\mathsf{unsplit}_v(G)$ denotes the graph that is obtained from G by removing v, and for all $i \in \{1, \dots, n\}$ adding a new node v_i and the edges (u, v_i) and (v_i, w_i). (See Figure 2(c).)

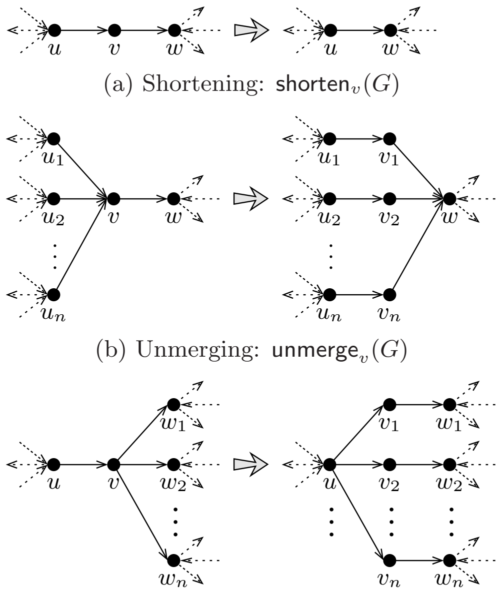
We now list the properties that we expect a scoring function $\text{score}(s, t, G)$ to satisfy.

- **Shortening property:** If node v is a connector, then it holds that $\text{score}(s, t, G) \leq \text{score}(s, t, \mathsf{shorten}_v(G))$.

- **Unmerge property:** If node v is a merger, then it holds that $\text{score}(s, t, G) \leq \text{score}(s, t, \mathsf{unmerge}_v(G))$.

- **Unsplit property:** If node v is a splitter, then it holds that $\text{score}(s, t, G) \leq \text{score}(s, t, \mathsf{unsplit}_v(G))$.

[1]Note that when a node v is *removed*, every edge that is incident to v is removed as well.



(a) Shortening: $\mathsf{shorten}_v(G)$

(b) Unmerging: $\mathsf{unmerge}_v(G)$

(c) Unsplitting: $\mathsf{unsplit}_v(G)$

Figure 2: Transformations

Intuitively, the shortening property should be satisfied, as shorter paths from s to t obviously reflect a closer relationship between the two. The unmerge and unsplit properties should be satisfied, as they intuitively state that introducing more independent opinions (i.e., disjoint edges), should only improve the ranking of t.

Example 3.1. To demonstrate these properties, let G be the graph of Figure 1, and let s be the node Sally. The node on the path from Sally to Ted is a connector. Hence, removing this node, and directly connecting Sally to Ted, should raise Ted's score (shortening property). Observe the merger node on the paths from Sally to Tony. Applying an unmerge transformation to this node would result in the graph structure currently existing between Sally and Tim, and hence, should raise Tony's score (unmerge property). □

Note that all properties use the rather weak "$\leq$" to indicate that the change in G should not lower score(s, t, G). Given a specific scoring function, the shortening property is *satisfied in the strong sense* if for all s, t, G and connector nodes v, $\text{score}(s, t, G) < \text{score}(s, t, \mathsf{shorten}_v(G))$. (Note the strict inequality.) Similarly, this property is *satisfied in the trivial sense* if $\text{score}(s, t, G) = \text{score}(s, t, \mathsf{shorten}_v(G))$ always holds. Finally, this property is *satisfied in the weak sense* if the inequality $\text{score}(s, t, G) \leq \text{score}(s, t, \mathsf{shorten}_v(G))$ is sometimes strict. We define satisfaction in the strong, trivial and weak sense similarly for the other two properties.

Remark 3.2. Due to the small-world property, often observed in social networks [12], there is a high likelihood that any two given nodes will be connected by a short path. Hence, it sometimes may make sense to only take a limited neighborhood of a node into account when computing the scoring function. We do not directly formulate this requirement as one of our properties. However, it can usually be taken into consideration by taking any scoring function, and applying it only to a neighborhood-bounded projection of nodes s and t. Due to space limitations, this is not discussed further. □

In the upcoming section we explore several scoring functions, and determine which of the properties they satisfy, and thus, whether they are appropriate for use in ranking results of egocentric person search.

4. SCORING FUNCTIONS

The goal of a scoring function is to quantify the relationship between s and t in a graph G, for the purpose of egocentric person search. Attempts to quantify the relationship between nodes have been made in the past, for different goals. For example, the *link prediction problem* is defined as follows: *Given a snapshot of a social network at time T, link prediction is to accurately predict the edges that will be added to the network during the interval from time T to a given future time T'.* Intuitively, a link is more likely to be added from s to t during this interval, if they are already well-related at time T. Hence, scoring functions for link prediction measure the relatedness of s and t.

Several link prediction measures have been studied in the past. We consider some of the more prominent functions:

- $\text{rdist}(s, t, G)$ is the reciprocal of the distance from s to t in G, i.e., $(\text{dist}(s, t, G))^{-1}$, where $\text{dist}(s, t, G)$ is the length of the shortest path from s to t.
- $\text{allPaths}(s, t, G)$ is the sum $\sum_{l=1}^{\infty} \beta^l |\text{paths}^l_{s,t,G}|$ where $\text{paths}^l_{s,t,G}$ is the set of all length-l paths from s to t in G [7, 8]. Thus, $\text{allPaths}(s, t, G)$ directly sums over all paths from s to t, exponentially dampening by length to count short paths more heavily.
- $\text{rootPR}(s, t, G)$ (i.e., rooted PageRank [8]) is the stationary probability of t in a random walk that returns to s with probability α at each step, moving to a random neighbor with probability $1 - \alpha$.

Table 1 shows which properties are satisfied (and to which degree) by the three functions.

Example 4.1. Consider again the graph in Figure 1. Suppose that s is the node Sally, and we are interested in ranking nodes Tim, Ted and Tony. The function rdist gives the same score to all three nodes, as all are at distance 2 from Sally. The function allPaths will score Tim and Tony equally (and above Ted), as they both have precisely one path of length 2, and two paths of length 3. Similarly, rootPR will score Tim and Tony equally (and above Tim). Rooted PageRank cannot differentiate between the graph structure relating Sally to Tim, and that to Tony. On the other hand, it would seem that Tim should be the highest scoring, as its paths are independent (while those to Tony are not). None of the three scoring functions achieve such a scoring. □

4.1 Expected Distance and Reliability

It is easy to see that with respect to our properties, none of the functions considered thus far is a good fit for node scoring. Therefore, we introduce two new functions (expected distance and reliability), which can be appropriate for node scoring. (To the best of our knowledge, these functions have not been considered in the past for related social network

	Shortening	Unmerge	Unsplit
rdist(s,t,G)	weak	trivial	trivial
allPaths(s,t,G)	strong	trivial	trivial
rootPR(s,t,G)	strong	trivial	fails

Table 1: Table of property satisfaction.

scoring problems.)

Expected Distance. We first consider the expected distance function. Intuitively, this function measures the expected distance from s to t, when each edge is removed with probability p. Note that for this value to be well defined, we will chose a number m, that is returned by the function, when no path from s to t exists.

We fix a parameter $m \in \mathbb{R}$, and a probability $p \in (0,1)$. We will implicitly assume that m is larger than the number of nodes in the graph G. The *m-bounded distance* from s to t, denoted $\hat{\delta}_G(s,t)$, is defined by

$$\hat{\delta}_G(s,t) \stackrel{\text{def}}{=} \min\{\text{dist}(s,t,G), m\} .$$

Thus, if G has no path from s to t, then $\hat{\delta}_G(s,t) = m$. Note that if $s \neq t$, then $\hat{\delta}_G(s,t)$ is always in the interval $[1,m]$.

We denote by G^{r} a random subgraph of G that is obtained by removing each edge of G, independently, with probability $1-p$. The *expected* m-bounded distance, denoted by $\overline{\delta_G}(s,t)$, is defined as follows.

$$\overline{\delta_G}(s,t) \stackrel{\text{def}}{=} \mathbb{E}\left[\hat{\delta}_{G^{\text{r}}}(s,t)\right] .$$

That is, $\overline{\delta_G}(s,t)$ is the expected m-bounded distance from s to t in a random subgraph G^{r} of G. Finally, our scoring function is the reciprocal of $\overline{\delta_G}(s,t)$, namely

$$expd(s,t,G) \stackrel{\text{def}}{=} (\overline{\delta_G}(s,t))^{-1} .$$

Reliability. Reliability is another function that strongly satisfies all properties. Intuitively, reliability measures the likelihood that a random subgraph G^{r} of G contains a path from s to t. Intuitively, reliability satisfies the shortening property, since longer paths are more likely to be disconnected when a random subgraph is chosen. Similarly, reliability satisfies the unmerge and unsplit properties, since they give preference to graphs with disjoint paths, which in turn, increase the likelihood of s and t being connected in G^{r}. Fixing a probability $p \in (0,1)$, we define

$$rel(s,t,G) = \Pr\left[G^{\text{r}} \text{ has a path from } s \text{ to } t\right] .$$

The following theorem shows that both functions strongly satisfy all three properties. The proof is nontrivial, and is omitted, due to space restrictions. In the case of expected distance, the proof is based on the notion of *stochastic ordering* [11].

THEOREM 4.2. *The functions $expd(s,t,G)$ and $rel(s,t,G)$ strongly satisfy all three properties.*

REMARK 4.3. Although *expd* and *rel* strongly satisfy all properties, they do not necessarily imply the same relative ranking for all nodes. To demonstrate that, we consider the ranking of the nodes t_1 and t_2 in Figure 3. Assume that there are sufficiently many disjoint paths from s to t_2 so that

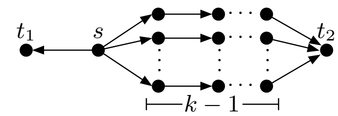


Figure 3: Expected distance vs. reliability

$rel(s,t_2,G) > p$. Since $rel(s,t_1,G) = p$, we get that *rel* ranks t_1 lower than t_2. However, the expected distance from s to t_1 is $p + (1-p)m$, and that from s to t_2 is at least k; hence, choosing the parameters to be such that $k > p + (1-p)m$ would make *expd* rank t_1 higher than t_2. □

5. CONCLUSION

This paper presents a first attempt at defining principles that should guide node relevance ranking in egocentric person search. Three properties, determining how graph manipulations should affect node scoring, were presented. Traditional node scoring functions were analyzed with respect to these properties, as well as two additional functions, which are shown to strongly satisfy the properties.

For future work we intend to experimentally test the effectiveness of various node scoring schemes and validate the given properties. We also intend to study additional models of social networks, such as the undirected model (e.g., to represent social relationships in Facebook), as well as social networks with weak and strong (or more generally, weighted) edges [6].

Acknowledgements

The research of Sara Cohen was partially supported by the ISF (Grant 143/09) and the Israeli Ministry of Science and Technology (Grant 3-6472).

6. REFERENCES

[1] L. Adamic and E. Adar. How to search a social network. *Social Networks*, 27(3):187–203, 2005.

[2] L. Adamic, R. Lukose, A. Puniyani, and B. Huberman. Search in power-law networks. *Phys. Rev.*, E 64(046135):187–203, 2001.

[3] P. Briggs and B. Smyth. Harnessing trust in social search. In *ECIR*, pages 525–532, Berlin, Heidelberg, 2007. Springer-Verlag.

[4] R. D'Amore. Expertise community detection. In *SIGIR*, 2004.

[5] L. C. Freeman. *The Development of Social Network Analysis: A Study in the Sociology of Science*. Empirical Press, 2004.

[6] S. Granovetter. The strength of weak ties. *American Journal of Sociology*, 78:1360–80, 1973.

[7] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[8] D. Liben-Nowell and J. M. Kleinberg. The link-prediction problem for social networks. *JASIST*, 58(7):1019–1031, 2007.

[9] K. McNally, M. P. O'Mahony, B. Smyth, M. Coyle, and P. Briggs. Towards a reputation-based model of social web search. In *IUI*, pages 179–188, New York, NY, USA, 2010. ACM.

[10] R. Schenkel, T. Crecelius, M. Kacimi, S. Michel, T. Neumann, J. X. Parreira, and G. Weikum. Efficient top-k querying over social-tagging networks. In *SIGIR*, pages 523–530, New York, NY, USA, 2008. ACM.

[11] M. Shaked and J. G. Shanthikumar. *Stochastic orders and their applications*. Probability and Mathematical Statistics. Academic Press Inc., Boston, MA, 1994.

[12] J. Travers and S. Milgram. An experimental study of the small world problem. *Sociometry*, 32(4):425–443, 1969.

[13] S. B. Yang and H. Garcia-Molina. Improving search in peer-to-peer networks. In *ICDCS*, pages 5–14, 2002.