Personalizing Forum Search Using Multidimensional Random Walks

Gayatree Ganu
Computer Science
Rutgers University
gganu@cs.rutgers.edu

Amélie Marian
Computer Science
Rutgers University
amelie@cs.rutgers.edu

Abstract

Online forums are a vital resource for users to ask questions and to participate in discussions. Yet, the search functionality on such forum sites is very primitive; posts containing the searched keywords are retrieved in the order of their creation date. In these interactive and social web forum sites, users frequently make connections with other users due to shared interests, same information needs or similar profiles. A critical challenge then, is to score and rank the forum posts while taking into account these relations between users. In this paper, we present a personalized search over forums that leverages user similarities developed via multiple relations linking users. We build a novel multidimensional random walk model that uniformly incorporates the heterogeneous user relations to find similar forum participants. We then use this multi-relational user similarity to predict future interactions by personalizing answer search. Furthermore, we extend our methods to enhance keyword search for forum readers, by using expertise scores for all existing forum participants. Our results show that by leveraging the author dimension we can retrieve more relevant results than the traditional IR scoring alone.

1 Introduction

Forums are increasingly popular for seeking answers to technical problems, providing opinions on specific products or services, and sharing experiences. A participant in an online forum can initiate a thread by posting a question or invoking a discussion, and then waiting for responses from other forum users. Alternatively, users can search through existing content for answers to their questions. The collective knowledge of the forum user community in the form of archived discussions is a valuable resource for all Web users.

Despite their popularity, the search functionality available on the forum sites is often very primitive. Usually the results retrieved in response to a user query are posts containing the query keywords, ordered chronologically. There is little or no ranking of results based on the content in the posts. Moreover, isolated posts are not always the right focus level [Ganu and Marian, 2013]. A search system can leverage the inherent social interactions between forum participants to enhance user experience. Forum participant interactions provide vital clues about their information needs, interests and their preferred other users to answer questions. Some users are prolific and knowledgeable, and participate in many different discussions on varying topics. Such users are likely to contribute high quality information and their content should have higher ranking scores. Alternately, some users are similar to each other. For instance, patients of a particular cancer stage (Stage I through IV) are more likely to interact with others with the same progression of the disease [Jha and Elhadad, 2010]. Finding such similar users and weighting their content strongly will enhance personalized search. Unfortunately, existing forums do not provide such personalization.

Finding similarities in forum participants can enable a search system to retrieve more useful results authored by like-minded users. However, users interact with each other for a variety of reasons. Forums often allow users to make explicit friendships. Additionally, there exist several implicit cues of user affinity like participating in the same threads or discussing the same topics. Yet, two users having similar information needs at different times might never participate in the same discussions. For instance, in a forum for mothers, several participants will have similar questions about feeding, teething, and sleep patterns. However, some mothers with older children will never participate in newer threads related to infants. Alternately, for a location-based business search forum participants in the query location are likely to provide answers despite largely varying profiles or interest. Thus, it is a challenging problem to uniformly capture similarities in online users while incorporating multiple signals like profiles, interests or information needs.

Our approach to address the problem of finding like-minded forum participants is to use a multidimensional random walk that dynamically learns importance of the various inter-user relations. Random walk (RW) on graphs correctly captures many notions of node similarity. However, existing RW algorithms assume that the underlying graph is homogeneous comprising of nodes and edges of a single type each. Our work extends the RW algorithm to a multidimensional scenario, where each dimension represents a different relation semantic connecting the nodes. Moreover, our algorithm dynamically learns the importance of the various interpersonal relations w.r.t a user and finds the top-k most similar other users across heterogeneous relations.

In particular, we make the following contributions:

- We design several implicit signals of user affinity and build these relations over forum participants (Section 2).
- We propose a novel multidimensional random walk algorithm over a heterogeneous graph of user interactions (Section 3), to find the most similar nodes to a user. Our main contribution is the method to learn the egocentric importance of various user relations.
- We leverage the multidimensional similarity computation to make predictions on forum participants who are most likely to answer a question asked by a particular user (Section 4). Predicting forum participation is useful in making recommendations of users and threads to follow.
- Lastly, we enhance keyword search by re-ranking results using the importance of content contributors (Section 5) and show improvements purely IR-based text scoring.

The rest of the paper is structured as follows. We describe our forum dataset and the design of several implicit user affinity signals in Section 2. In Section 3, we present our multidimensional random walk (MRW) model for dynamically learning the importance of the heterogeneous relations between users. We demonstrate the utility of our multidimensional similarity computation for enhancing personalized search by predicting future forum interactions (Section 4). Next, in Section 5 we re-rank the results retrieved by *tf*idf* scoring using the learned importance of the authors, thus enhancing non-personalized keyword search. We present related work in Section 6 and conclude in Section 7.

2 Forum Dataset and Implicit User Relations

We build several implicit connections amongst participants in a breast cancer patient forum dataset. The data was collected from the publicly available posts and discussions on the online site `breastcancer.org`. The forum data contains threads on a variety of topics useful to breast cancer patients as well as for health professionals. The search offered by the web site over its forum data is very basic. Posts are presented chronologically filtered by keywords, with little scoring and ranking and no personalization.

The forum corpus is a large collection of 31,452 threads comprising of 300,951 posts. The posts in the corpus are written by 15K authors for whom we have unique usernames and an optional signature containing information like location, stage of the disease, date of cancer detection and current treatment plan. We prune infrequent mis-spellings and word formulations and retain 46K keywords occurring at least five times in the entire corpus. The corpus does not contain a reply structure or any explicit social network like friendships over the users. We now describe the different implicit relationships linking the forum participants.

2.1 Thread Co-participation

Our corpus contains 31,452 threads with an average of 9.7 and a median of 7 posts in each thread. Participants ask questions or invoke discussions through the first post in a thread, and other participants provide answers or opinions on the topic in the thread. When participants often post in the same threads, it indicates their shared interests or expertise in the topics covered in the threads, or their shared information needs. Therefore, we build a thread co-participation relation C between the forum participants. In C a directed edge exists from a user i to a user j if i posts in a thread after a posting by j; this directed edge is C_{ij} and its edge weight is $ew_C(i,j)$. $ew_C(i,j)$ represents the frequency of user interaction and is equal to the number of unique threads in which i posts after j. A higher edge weight indicates a stronger relation between the two users. C often contains edges in both directions having different weights, i.e. $ew_C(i,j)$ is usually not equal to $ew_C(j,i)$ due to the asymmetric ordering of posts of users i and j within threads.

2.2 Proximity of Thread Interaction

Threads in forums often span several posts, and frequently the theme of discussion changes as participants digress. Users who post in a thread are more likely to read contributions close to their posts, and are more likely to interact with such users in the future. We build a post separation relation matrix D where a directed link from forum participant i to j exists if i posts in a thread after j, and the edge weight $ew_D(i,j)$ is computed as the inverse distance between the posts of user j and user i averaged across all commonly participated threads. The minimum distance separating two posts is 1 (consecutive posts). As the distance between the posts of i and j increases, the relation is weaker and the edge weight decreases, i.e. $max(ew_D(i,j)) = 1$. The edges in this relation D are the same as those in the matrix C from Section 2.1, but the definition of the edge weight computation captures a different semantic of user interaction: D captures the closeness of user interactions within threads.

2.3 Topical Similarity from Text

The relations described above are built on common thread participation which is often constrained by temporal factors. We now build a relation between forum participants using the similarity in the text in all the posts contributed by them.

To build an implicit topical similarity relation that takes into account word synonyms in finding similarities, we implement a Latent Dirichlet Allocation (LDA) topic model using the Stanford topic modeling toolbox (www.nlp.stanford.edu/software/tmt) over the text in posts contributed by each user. LDA enables us to derive a probability score representing user contribution for each topic; we implemented LDA with 100 topics. Users who often write about a topic, even with slightly different words in their language model, have similar topical probabilities. Users are now represented with only 100 topic features. We then build a text similarity relation T with a directed link between user i and user j as the cosine similarity [Manning, Raghavan, and Schtze, 2008] of their topical feature vectors. Note that, all links in T are symmetric, i.e., $ew_T(i,j) = ew_T(j,i)$.

2.4 Profile Similarity

Finally, we capture the profile information from the optional signatures of authors. 71.3% posts in our corpus contain a free-form text format signature. A large majority of users write about their disease stage, treatment options, first diagnosed date and other highly relevant information which we leverage to build the signatures relation S. We first find all unique signatures in our corpus. We then tokenized these to find unigrams, bigrams and trigrams and we retain 10% of the most frequent phrases of each length, resulting in 11K unique features. Some examples of commonly occurring unigrams were *HER2-, Stage, Grades, 2cm, bilateral, mastectomy* showing the different cancer tumor characteristics and treatment directions. Bigrams included *Stage I* and all other stages of the disease, grade and tumor size details. Trigrams contained phrases like *mastectomy without reconstruction*. Therefore, signatures are useful for finding user similarity based on their disease progression and treatment. We then build the pairwise relation S as the cosine similarity between n-gram frequencies of terms in user signatures.

In the following section, we design a novel multidimensional random walk algorithm that finds similarity between forum participants through a uniform combination of the four similarity indicators C, D, T and S described above.

3 Random Walks for User Similarity

Random walks on graphs are a popular technique to find the important or influential nodes in a graph. Perhaps, the most popular random walk application is the PageRank algorithm [Page et al., 1999]. We now describe the preliminaries of the Power Iteration method, as it is defined on homogeneous networks in Section 3.1. The RW computation can be transformed into an egocentric similarity computation using a fixed root node as described in Section 3.2. We illustrate the ability of random walks to capture many different notions of node similarity. We then describe our novel multidimensional random walk (MRW) algorithm in Section 3.3, which can dynamically learn the importance of the various relations and combine these in a weighted transition matrix.

3.1 Random Walks on Social Graphs

The PageRank algorithm [Page et al., 1999] was developed to determine the importance of a web page in the homogeneous Internet graph where nodes represent the web pages and the edges represent directed hyperlinks.

Preliminaries: Let $G = (V, E)$ be a homogeneous network with vertices V representing entities of the same type and edges E representing a single relation between the vertices. G contains a directed edge G_{ij} if node i links to node j and carries a weight $ew_G(i, j)$ representing the strength of the directed link. Let the nodes in the network be numbered from $1, \ldots, n$ and the PageRank of the web pages be represented by the vector P, i.e., $p_1, \ldots, p_n$ are the PageRank scores of the n vertices. The PageRank p_i of a node i is a number in $(0, 1)$ and represents the stationary probability that a random walk reaches this node i.

Iterative PageRank Computation: Let A be a $n \times n$ matrix representing the link structure of the graph G. A_{ij} is defined to be zero if node j does not link to node i, and $ew_A(i, j)/\sum_k ew_A(k, j) \forall i, j, k \in V$ if node j links to node i. The value A_{ij} represents the probability that the random walk from node j will take the next step to node i.

The PageRank vector P is computed as $P = A \times P$. The PageRank vector P is the eigen vector of the adjacency matrix A. In experiments, P in each iteration is computed by iteratively multiplying A: $P^{t+1} = A \times P^t$.

This computation is repeated till there is no significant change in P, i.e., $\left\|P^{t+1}\right\|_1 - \|P^t\|_1 < \epsilon$. At convergence we arrive at the PageRank scores for every node in the network.

In practice the relation matrix A is replaced by the transition matrix M which includes adjustment for dangling nodes as well as random teleportations: $M = \alpha(A+D)+(1-\alpha)E$. D is a $n \times n$ matrix representing the transition from a dangling node. For a dangling node j having no out-links, the j-th column of the matrix D has all entries $1/n$ assigning uniform probability of picking any node in the graph. The matrix E represents the teleportation step, i.e., instead of following out-links from nodes the walk randomly jumps to any other node in the graph. E has all entries set to $1/n$. With a non-zero probability α the random walk proceeds along the out-links of nodes, and with a probability $(1 - \alpha)$ there is a jump to a random node in the graph. Usually, α is set to 0.85. Thus, the PageRank is computed over this modified transition matrix as $P^{t+1} = M \times P^t$.

3.2 Rooted Random Walks

PageRank assigns a score to each node which represents the node's relative importance. For personalized search, on the other hand, we are interested in finding similarities between users to find top-k closest neighbors. We now describe a modification of the RW computation that captures egocentric node similarity, i.e., similarity w.r.t. a fixed node.

Algorithm: The rooted random walk (rooted-RW) [Liben-Nowell and Kleinberg, 2003] is computed on a modified teleportation matrix E. We fix a node r as the root node of the random walk. The matrix E is modified such that every entry in the r-th row is set to 1 and all other entries are 0. During the teleportation step the random surfer can jump only to the root node r with a probability proportional to $(1 - \alpha)$; the random walk originating from the root node r periodically resets and returns to r. Hence, we are less likely to traverse to distant nodes from r, which is desired since these distant nodes are less likely to be similar to the root node. The rooted-RW score of a node j w.r.t. the root node r is $Score(j)_r$, defined as follows:

$$Score(j)_r \quad = \quad \text{Stationary weight of } j \text{ under the RW: } \\ \text{move to random neighbor with } \alpha \\ \text{return to } r \text{ with } (1-\alpha) \qquad (1)$$

$Score(j)_r$ represents the probability of a random walk originating at r and reaching j following the links in the graph. $Score(j)_r$ represents the similarity of j w.r.t the root node r. In the next section, we describe some of the desirable properties of random walks and how they closely capture many notions of node similarity in social networks of users.

Interpreting Node Similarity: In a network with entities represented by nodes, the definition of similarity closely depends on the definition of the edges connecting these entities. When the edges represent strength of connection or

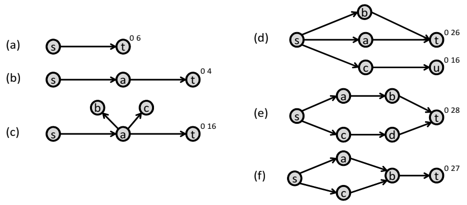

Figure 1: Node similarity scores captured by rooted-RW.

association, node similarity can be captured using random walks along the edges of a network.

Consider the example graphs in Figure 1 having the root node s and edges with unit weights. We compute the scores of target nodes t and u w.r.t s using Equation 1. Rooted-RW correctly capture the following notions of similarity:

- Nodes closer in the network are more similar.
- Paths via large out-degree intermediate nodes contributes lower similarity than paths via low out-degree nodes.
- A node connected to the root node through multiple paths is more similar to the root node.
- For two nodes connected via multiple paths, independent paths indicate stronger similarity than overlapping paths.

As shown in Figure 1(a), the proximity of t w.r.t. the root node s is higher (0.6) with a shorter path connecting the nodes s and t, than in Figure 1(b) with a $s-t$ path length of two. In comparison, if the intermediate node a has a large out-degree (Figure 1(c)), then t is weakly similar to s with a score 0.16, which is much lower than in Figure 1(b). In Figure 1(d) node t has a higher similarity score than node u because the root node s is connected to t though more paths. Lastly, t in Figure 1(e) has a higher similarity score than that in Figure 1(f) due to the two independent $s-t$ paths in Figure 1(e). The illustrative examples show that the rooted random walks are a suitable measure to capture egocentric node similarity. Yet, these walks are defined on homogeneous networks with nodes and edges of a single type each.

Next, we describe our novel MRW algorithm for uniformly capturing similarity over heterogeneous networks.

3.3 Multidimensional Random Walks

Entities are often linked through multiple relations. For instance, people can become friends due to their shared interests, location proximity, same age or gender or having similar experiences at the same time. The semantics of these different relations are distinct and merging these to create a homogeneous connections graph will result in obfuscating important characteristics. There is a need to distinguish between the reasons for user similarity, and for dynamically choosing the importance of the relations for each user. We now present our MRW algorithm that uniformly leverages heterogeneous relations for finding node similarities.

Random Walks on Heterogeneous Graphs: We first define heterogeneous graphs:

Definition 1 *A* ***heterogeneous graph*** $G = (V_N, E_R)$ *is a graph with a node mapping* ϕ *and an edge mapping* ψ *where each node* $v \in V_N$ *is mapped to one node type* $\phi(v) \rightarrow N$ *and each edge* $e \in E_R$ *is mapped to a link type* $\psi(e) \rightarrow R$. *There are* N *types of nodes and* R *types of links or relations. When* $|N| > 1$ *or* $|R| > 1$, *the graph is called a heterogeneous graph [Sun et al., 2011].*

If $|N| = 1$ and $|R| = 1$, then the graph is said to be homogeneous. A homogeneous graph comprising of a single node type and a single link type can be represented as an $n \times n$ adjacency matrix A, where A_{ij} represents a link between node i and j with a value $ew_A(i,j)$ proportional to the strength of the connection. In our multi-relational scenario, we have several such matrices $A_1, A_2, \ldots, A_k$ where there exist $k = |R|$ different relations linking nodes. A multidimensional random walk is then defined as follows:

Definition 2 *Let* $G = (V_1, E_R)$ *be a heterogeneous graph with* V_1 *nodes and* $|R|$ *types of links. Let* A_1, A_2, $\ldots$, A_k *each represent a single relation semantic linking the nodes in* V_1. *A* ***multidimensional random walk*** *is a random walk on the composite adjacency matrix* $A = \theta_1 * A_1 + \theta_2 * A_2 + ... + \theta_k * A_k$ *where* $\sum_i \theta_i = 1$ *and all* $\theta_i \geq 0$.

The composite matrix A is a convex combination of the matrices representing the different semantic relations connecting the nodes. In other words, the MRW can be interpreted as follows: when the RW arrives at a node, first a relation i is chosen with probability θ_i and then we jump to an adjoining node according to the matrix A_i.

Thus, we have a unified algorithm for combining the different user relations. We now describe our technique to build these relation weights θ_i in an egocentric manner.

Egocentric Weights Computation: A critical part of the MRW algorithm described above is the computation of the relation weights θ_i. We define the weights in an egocentric manner w.r.t the root node. If the root node has a higher edge weight for links of a particular relation, then this relation should be more significant in finding similarities w.r.t. this root node. Therefore, personal preferences should be taken into consideration while determining weights for the multiple dimensions of user relations.

For a root node r, the relation weight θ_i for the i-th relation amongst the $k = |R|$ user relations is computed as:

$$\theta_i(r) = \frac{\sum_m ew_{A_i}(r,m)}{\sum_k \sum_j ew_{A_k}(r,j)} \quad \ldots \forall m \in A_i, \forall j \in A_k \quad (2)$$

In the above equation, $ew_{A_i}(x,y)$ represents the edge weight or strength of relation between node x and node y in the graph representing the relation i. The egocentric weights $\theta_i(r)$ to be associated with each relation are developed as the relative weights of edges of relation i originating from the root node r to the total weights of the edges from r. The weight $\theta_i(r)$ is high if relation i is more important w.r.t the root node r compared to the other relations. Note that, the weights $\theta_i(r)$ are computed only w.r.t. the root node r and are not updated at each step of the random walk. The weights are not updated so that we correctly capture the importance of relations w.r.t. the designated root node. For instance, if

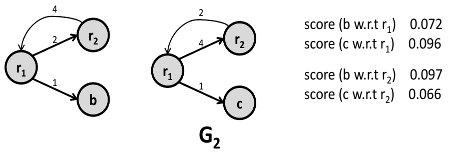

Figure 2: Node similarity using multidimensional RW.

topical similarity is important to a root node r for making connections, then the friends' friends of r who are similar to r's friends due to other reasons are less significant to node r. Thus, we compute relation weights egocentrically taking into account the particular preferences of the root node.

Interpreting Multidimensional Similarity: We demonstrate the utility of our MRW algorithm with egocentric weights in capturing node similarity. Figure 2 shows two different node relationships represented by the graphs G_1 and G_2. We assume that there are two root nodes r_1 and r_2 and the edges in G_1 and G_2 represent the only connections between the nodes in each relation. The scores are computed using the composite matrix as in Equation 1. When the root node is r_1, the relation represented by G_1 has a weight proportional to $\theta_1 = 3/8$ and G_2 has a weight proportional to $\theta_2 = 5/8$. We expect that the relation in G_2 is more important w.r.t. r_1 because r_1 makes stronger connections in G_2 as indicated by the higher edge weights of outlinks. The relation weights correctly capture this bias. As a result, the node similarity scores computed using Equation 1 over the combination of two relations, assign the node c with a higher score of 0.096 than the node b (0.072), even though the edge weights $ew_{G_1}(r_1, b)$ and $ew_{G_2}(r_1, c)$ have equal unit weight. In contrast, when we compute scores w.r.t. the root node r_2, the relation in graph G_1 is more important. As a result, b has a higher similarity score w.r.t. r_2 than c.

Therefore, Figure 2 shows how the MRW correctly captures notions of egocentric similarity. The weights to be associated with each relation are chosen dynamically, allowing us to capture the varying importance of the relations w.r.t the root node in an egocentric manner.

Complexity: Algorithms for finding PageRank broadly use two approaches. The Power Iteration method [Page et al., 1999] as described in Section 3.1 uses linear algebraic techniques. The time complexity for computing rooted-RW using this method, for one root node is $O(Knd)$ where K is the number of iterations till convergence, n is the number of nodes in a graph and d represents the average neighborhood size. Extending the RW framework to the multidimensional scenario requires computing the composite transition matrix one time for each query root node, as described in Definition 2. The time complexity for computing the composite matrix is $O(nd)$, and we can see that our multidimensional framework does not add a significant overhead to the rooted RW score computation. The second approach to compute PageRank is based on Monte Carlo approximation and is very efficient and highly scalable [Avrachenkov et al., 2007]. In the future, we aim to implement the fast distributed mapreduce based algorithm in [Bahmani, Chowdhury, and Goel, 2010], which computes approximate rooted-RWs from each node in the graph in a highly efficient manner.

Next, we leverage the MRW algorithm in predicting future interactions between forum participants using a uniform combination of the four similarity indicators C, D, T and S.

4 Personalized Answer Search

When searching for information on online forums, users often pose a question by starting a new thread. Other interested participants then choose to participate in the discussion to help answer the question. An online forum will benefit largely if the likelihood of a user's participation in a thread is known. This will enable users to find and contribute to the best threads, as well as provide the search users with the most useful other users with whom they could interact, become friends and develop meaningful communications.

In this section, we first describe our experimental setting for predicting user participation in threads in Section 4.1. We then use our MRW algorithm to find the top-k most similar users to the searcher, and predict that these similar users will answer the question posted in the thread (Section 4.2). Furthermore, in Section 4.3 we combine these user similarity scores with the user expertise on the particular question in the thread, to improve predictions on participation.

4.1 Evaluation Setting

We predict which forum participants are likely to answer a new question posted in a thread. For evaluating our methods, we divide the forum data into a training set comprising of 90% of the threads which were initiated before the remaining threads. These remaining 10% threads are used as a test set. We have about 2.1K threads in the test set and 28K threads in the training set. Leveraging the information in the training data, we build the different adjacency matrices C, D, T and S representing the various relations between the users. We also learn user preferences towards each relation from the training data to build weights for our MRW framework. The text in the initial posts and the users initiating the test threads are used to predict which other forum participants are most likely to participate in the given discussion. Thus, we design a new prediction task for forum participation which can be used to predict threads or other users which are most meaningful to follow.

4.2 Leveraging User Similarity

As described above, we make predictions on the forum participants who are most likely to answer a question posed by the user in the test thread. We do this in the following manner. We first compute the similarity w.r.t the user posing the question, called the test user, with all other forum participants. This similarity is developed using Equation 1 over each of the four interpersonal relations C, D, T and S separately using the rooted random walks as described in Section 3.2. Therefore, we first compute user similarities using single homogeneous signals of user affinity. We then develop a combined similarity using our MRW model which

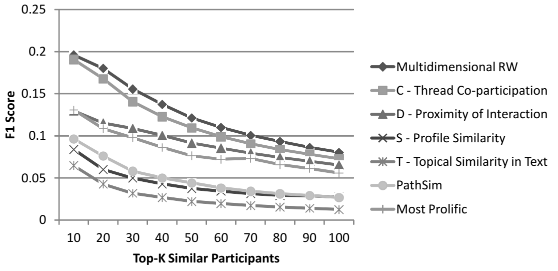

Figure 3: F1 score for forum participation prediction.

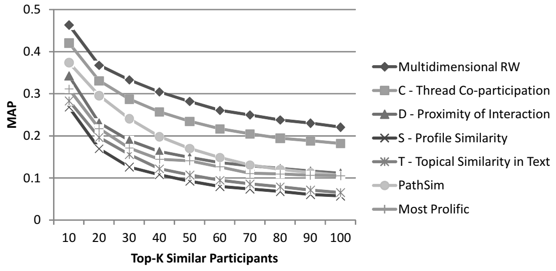

Figure 4: MAP for forum participation prediction.

incorporates the four user relations in a unified manner, with the computation of egocentric weights assigning varying importance to each relation. As an additional baseline, we also naively predict that the most prolific users in our training corpus are most likely to participate in the test threads.

For comparison we also compute the user similarity using the *PathSim* similarity metric defined in [Sun et al., 2011]. To the best of our knowledge, *PathSim* is the only user similarity metric defined on heterogeneous networks. However, *PathSim* has three key differences from our MRW model. First, *PathSim* defines a fixed path over the relations for finding node similarities using a fixed ordered product of matrices representing the individual relations. For instance to find users U having similar topical interests T, a path UTU is defined. It is not clear how to choose the best paths or how to combine the similarity computed using different paths. Second, due to the predefined paths, similarity of users separated by a distance longer than the length of the path cannot be computed. Lastly and most importantly, the *PathSim* metric does not allow for computing egocentric importance to be associated with the different inter-user relations: a key advantage of our MRW algorithm.

Once we generate the similarity of all users w.r.t. the test user, we rank these users to find top-k most similar users. We predict that these top-k users are most likely to participate in the discussion initiated by the test user. Recall from Section 2 that a thread in our corpus has very few posts on average. Hence, we make predictions using small values of k, i.e., $k = 10, 20, \ldots, 100$ most similar users.

Figure 3 shows the performance of the different similarity computation methods for predicting forum participation. As shown, our multidimensional RW algorithm has the highest prediction F1 score amongst all the methods. We see high precision at low values of k neighbors. Across the 2.1K test threads precision@10 of our multidimensional RW algorithm is 0.24 which is higher than any of the alternate similarity computation methods in Figure 3. Note that making accurate predictions in this scenario is a notably hard task: we have 15K authors in our corpus who may or may not participate in a thread for a variety of reasons. As k increases precision decreases but recall increases from recall@10 at 0.22 to recall@100 at 0.41, as expected. The forum participation prediction using single relations has much lower F1 score. The thread co-participation relation C as developed in Section 2.1 is the strongest single indicator of future interactions between users. Yet, our multi-relation approach significantly improves in the prediction F1 score ($p - value < 0.01$) over the thread co-participation relation by 3% at top-10 neighbors and 10% at top 100 neighbors. The naive approach of making predictions of the most prolific users has a significantly worse performance than several similarity-based measures. Therefore, incorporating the different heterogeneous relations in computing user similarity is beneficial in predicting forum participation.

The *PathSim* baseline computation on fixed length paths performs significantly worse than our MRW method. Figure 3 shows the *PathSim* average prediction performance across all metapaths of length four involving each of the similarity relation once. *PathSim* does not allow for computing egocentric weights, and does not uniformly capture node similarity across the entire graph of relations connecting users. The multidimensional RW method makes more accurate predictions than the *PathSim* method with an improvement of 103% at $k = 10$ and 199% at $k = 100$ neighbors, and these improvements are statistically significant ($p - value < 0.01$).

The F1 Score is a set-based measure that does not take into account the relative ordering of the predictions. We now compare the alternate prediction methods using Mean Average Precision (MAP). MAP computation takes into account that the correct predictions of forum participation should be the predictions with the most confidence, i.e., the highest similarity with the test user. Figure 4 shows the MAP values for predictions using top-k most similar users. When evaluating MAP, the multidimensional random walk method has a significantly higher prediction MAP than any of the alternate methods. Our multidimensional RW approach improves over the single thread co-participation relation by 10% for $k = 10$ neighbors and 21% for $k = 100$, demonstrating the utility of incorporating multiple relations while computing user similarities. Our method again shows statistically significant ($p - value < 0.01$) improvements over *PathSim* with a 24% improvement at $k = 10$ and a large 108% improvement at $k = 100$.

4.3 Leveraging Topical Expertise

In so far, we have generated the similarity between the test user and other forum participants using their relations discovered in the training threads. In addition, we expect that certain users have useful knowledge in certain topics, as can

Neighbors	$\beta = 0$	$\beta = 0.1$	$\beta = 0.2$	$\beta = 1$
Top 5	0.52	0.64 (8%)	0.61 (4%)	0.59
Top 10	0.31	0.50 (8%)	0.49 (5%)	0.46
Top 15	0.24	0.43 (8%)	0.42 (6%)	0.40
Top 20	0.20	0.39 (6%)	0.39 (7%)	0.37

Table 1: Prediction MAP when combining *MRWScore* and *EScore* with trade-off parameter β.

be learned from their posts in the training data. We can improve prediction accuracy by utilizing the topical information in the text of the thread initiating post to find expert forum participants who have a prior knowledge in the area.

In this section, we combine the user similarity scores developed in Section 4.2 using our MRW algorithm, with the expertise score of the forum participants w.r.t. the topics in the first post of the test thread. To find the expertise score, we represent each user in our corpus by a 46K word vector containing the frequencies of words used in the posts authored by the user. We then use the cosine similarity [Manning, Raghavan, and Schtze, 2008] between the content words in the thread-initiating post and each forum participant. If a user has strong knowledge on the topics of the thread initiating post, he will have used a similar vocabulary in the past. This similarity score allows us to find an expertise score for each user in the topics of the thread-initiating post. Thus, we combine the multidimensional user similarity score *MRWScore* w.r.t the test user with the topical expertise score *EScore* w.r.t. the test post to generate the final score of a user as follows:

$$UScore = \beta \times MRWScore + (1 - \beta) \times EScore \quad (3)$$

The trade-off parameter β controls the effect of the two components of the score of a user. As $\beta \rightarrow 1$, *MRWScore* dominates the scoring function and we get the same top-k closest neighbors as in Section 4.2. Table 1 shows the prediction MAP for varying top-k users when combining the two user scores using Equation 3. Utilizing only the *EScore* at $\beta = 0$ or solely the *MRWScore* at $\beta = 1$ gives lower prediction MAP than the combined method of Equation 3, demonstrating the need for such a combined method. *EScore* alone has a worse performance than our method built on multiple user relations (*MRWScore*). When predicting that top-10 most similar users will participate in the forum threads, our *MRWScore* ($\beta = 1$) shows a 51% improvement in MAP over predictions using *EScore* alone ($\beta = 0$).

As shown in Table 1, a combined $UScore$ shows better prediction MAP than each of the two individual scores. We see noticeable improvements when the user expertise *EScore* has a high impact on the overall *UScore*, as seen at low values of β. For $k = 10$ most similar users, for $\beta = 0.1, 0.2, 0.3$ the percentage improvement over the pure *MRWScore* predictions is 8%, 5% and 2% respectively as shown in the parentheses in Table 1. Hence, incorporating the topical expertise of a user has a significant impact in improving prediction accuracy of forum participation.

Therefore, we demonstrate the utility of our MRW algorithm for computing user similarity. We enable a personalized search that takes into account a users past behavior and interactions to find other similar users and their preferred answers. Next, we utilize our multidimensional similarity model to enhance the non-personalized keyword search for a general user of the forum.

5 Re-ranking Results using Author Score

Users often visit online forums and search using the functionality provided on these web sites. Keyword search refers to such search behavior demonstrated by a random visitor to the forum site, who may or may not have participated in the forum discussions in the past. We cannot assume any information about the searcher, and cannot provide a personalized search for this user [1]. Yet, we can leverage the multidimensional relations between forum participants to find the most influential users in our corpus who are more knowledgeable, prolific and write better answers. Posts written by such users should have a higher rank in the results retrieved for a keyword search. In this section, we discuss our method to generate authority scores for users and utilize these for improving keyword search.

5.1 IR Scoring of Posts

The *tf*idf* scoring increases proportional to the frequency of a term in the document, but is offset by the number of documents containing the term to account for commonly occurring words. A common form of the *tf*idf* function [Manning, Raghavan, and Schtze, 2008] is shown below:

$$tf * idf = (1 + \log(tf_{t,d})) \times \log(\frac{N}{df_t}) \times \frac{1}{CL(d)^{\lambda}} \quad (4)$$

where the search term is t, the document to be scored is d, N is the total number of documents, $tf_{t,d}$ is the frequency of the term t in d and df_t is the number of documents containing the term. The scoring is inversely proportional to the character length of the textual object CL. This weighting is controlled by a parameter $\lambda, \lambda < 1$. We use this *tf*idf* scoring to retrieve posts in response to a keyword query, and refer to this score of a post as its $IRScore_{\lambda}$.

5.2 Authority Score of Users

Forum participants demonstrate varying behaviors; some users are more knowledgeable, prolific and write many different posts on a wide variety of topics. These users tend to participate in many different threads and interact with many other participants. Posts written by such users are likely to be of higher quality, containing more useful information. To test this hypothesis, we now find the most influential users in our forum data by developing an authority score for forum participants over our multidimensional user graph.

For our user authority score computation, we build a random walk over the multidimensional heterogeneous graph of user similarities, taking into account the four interpersonal

[1] Users could be logged in the forum site before issuing a search query. We can then leverage personalized information to improve keyword search. However, our corpus does not contain session information or query logs. In the future, we wish to combine personalized search with results re-ranking as described in this section.

relations C, D, T and S from Section 2. The composite adjacency matrix from Definition 2 is generated by assigning equal weights θ_i to each user relation A_i. Note that, the different relations have different overall importance in computing the authority scores of users, proportional to the number of edges and edge weights in the different relations. Assigning equal weights θ_i to each relation matrix allows the random walk to take into account the varying importance of relations, learned automatically from past user interactions in the corpus. We build a random walk over the heterogeneous multidimensional composite matrix in a non-rooted manner, to find the overall importance or influence of the users in our forum corpus, referred to as the $AuthorityScore$ for the users in our corpus. In comparison to [Balmin, Hristidis, and Papakonstantinou, 2004] where random walks are used on a document semantic similarity graph, our work uses the authorship information to enhance keyword search.

5.3 Qualitative Relevance Evaluation

We now evaluate the perceived quality of our results through crowd-sourced user studies. We first built a test-set of queries and then conduct user studies to compare the relevance of the returned result, as described below.

Representative Queries: A critical challenge in studying forum search is the lack of a test set. We evaluate the *tf*idf* scoring of posts using a set of 14 representative queries. These queries were chosen from different areas of interest for a breast cancer patient from side effects of a particular medicine, alternate treatment options, to food and ingredients beneficial to patients. The queries contain 1 to 3 keywords with an average of 1.7 keywords per query.

Evaluating the relevance of all answers to a keyword query is very expensive. Typically users are interested only in the top-k results where k is usually small. We assess the relevance of top-20 results retrieved using the *tf*idf* scoring function for each of the 14 test queries.

Graded Relevance Scale: It is common practice in earlier works to use a graded relevance scale [Kekäläinen and Järvelin, 2002]. Search results retrieving posts often suffer from the lack of context. For evaluating our ranked list of results, we adapt the relevance scale in [Kekäläinen and Järvelin, 2002; Pehcevski, 2006] designed specifically for assessing relevance at multiple focus levels, taking into account too much or too little context. Therefore, we ask judges to annotate search results with one of the following:

- *Exactly relevant*: Document contains highly relevant information at the exact level.
- *Relevant but too broad*: Document contains relevant information, but also other irrelevant information.
- *Relevant but too narrow*: Relevant information accompanied with little context.
- *Partial answer*: Partially relevant information.
- *Not Relevant*: No relevant information.

This scale captures user assessment towards varying granularity levels and the usefulness of the search results.

Gathering Relevance Assessments: We conducted relevance assessment on the Amazon Mechanical Turk crowdsourcing website (https://www.mturk.com/). Workers were given five results to a query at a time and were asked to mark the relevance according to the proposed scale. Workers were also provided with examples of search results belonging to each relevance grade. Our tasks were answered by high-quality workers with a 95% or higher acceptance rate. We evaluated batches of tasks to find spammers based on abnormal submissions, for instance when time taken was very low, and blocked these workers. As an additional quality check, each task answered by the workers had an unmarked honeypot question used to assess worker quality. The honey-pot questions were drawn from a pool of questions evaluated by us and had the least ambiguity (we often picked irrelevant text to remove the granularity subjectivity). The honey-pot questions were answered correctly by workers who understood the instructions and who were not spammers. After these quality filtering steps, we retained 71% of the relevance annotations, resulting in 7.6 individual assessments for each search result on average. The relevance assessments were completed by 175 workers, with 114 s required to complete each task on average. For computing the final relevance grade of a result, we used the expectation maximization (EM) algorithm proposed by Dawid and Skene [Dawid and Skene, 1979] that takes into account the quality of a worker in weighting his vote. Gathering multiple votes and these cleaning and pruning methods reduces the error in relevance judgements ensuring that the annotations obtained are highly reflective of a general user's perception.

5.4 Re-ranking results

As described in the previous section, we obtain relevance estimates on a graded scale for the top-20 results for our test queries. We now re-rank the posts retrieved by the *tf*idf* scoring using a trade-off parameter ω to compute a modified post score as shown below:

$$\begin{aligned} Score_{Post} &= \omega \times IRScore_{\lambda} + \\ &\quad (1-\omega) \times AuthorityScore \end{aligned} \tag{5}$$

We compare the relevance of the pure IR scoring with the re-ranked list of results leveraging the user *AuthorityScore*. We evaluate the ranked lists of results using mean average precision (MAP) [Manning, Raghavan, and Schtze, 2008]. Computing MAP requires binary relevance assessment. For our experiments we assume that if the users annotate a search result as Exactly relevant, Relevant but too broad or too narrow, then the result is relevant. Figure 5 shows the MAP of the top-10 ranked results for different values of the trade-off parameter ω. As described earlier, the IR scoring returns a different ranked list for each size parameter λ, and we show the MAP for two values, $\lambda = 0.1, 0.2$. As shown in the figure, we get a higher overall MAP when the results are re-ranked using the user $AuthorityScore$ generated by our multidimensional RW over the various implicit user relations. The MAP value peaks in the range of $\omega = 0.7$ to 0.9. Setting $\omega = 0.9$ is a suitable choice (larger focus on IR score) and at this value the combined post score from Equation 5 achieves a 5% and 4% improvement over the IR score

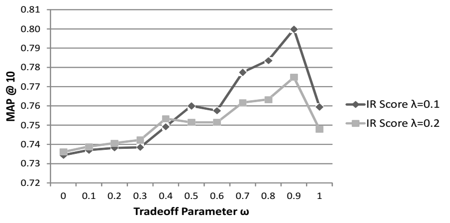

Figure 5: MAP of top-10 retrieved results.

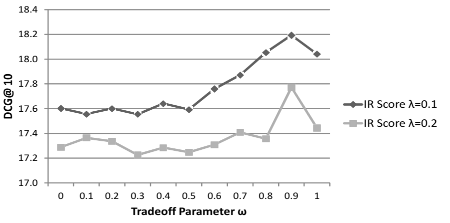

Figure 6: DCG of top-10 retrieved results.

ranking for $\lambda = 0.1, 0.2$ respectively. Hence, utilizing the authority score of users can have a noticeable impact on the perceived relevance within as few as the top-10 results.

We further investigate the quality of the re-ranked results by taking the gradation of the relevance assessments into account using the discounted cumulative gain (DCG) [Croft, Metzler, and Strohman, 2009]. DCG accounts for the decrease in importance of results as rank increases.

We translate the five relevance grades from Section 5.3 as follows: *Exactly relevant* has a score of 5, *Relevant but too broad* and *Relevant but too narrow* has a score of 4 and 3 respectively (incomplete information is worse than having to read extra text), *Partially relevant* has a score of 2, and *Not relevant* has a score of 1. Using these relevance scores we generated the DCG for each result list.

Figure 6 shows the comparison of DCG values for the different ranked lists controlled by the trade-off parameter ω. Again we see that the DCG of the re-ranked result set at $\omega = 0.9$ is higher than that of the pure IR scoring; our MRW method for computing *AuthorityScore* for forum participants assist in enhancing keyword search result relevance.

Thus, we build several implicit relations between online forum participants and demonstrate how these relations can be leveraged in a unified manner to enhance both personalized and keyword search.

6 Related Work

Many studies have discussed the different relations between online users. In [Adamic and Adar, 2005] the authors study the connections between users in two social networks using relations ranging from physical proximity, organizational hierarchy and profile information like gender or age. More recently, the authors in [Carmel et al., 2009] studied user similarity through explicit friendships or relations, through implicit co-participation and engagement with tags and comments, and a topic-based similarity. These studies indicate that there are many explicit and implicit reasons for user interactions in online communities, and there is a need for a unified framework for combining these diverse signals.

The PageRank citation ranking [Page et al., 1999] and several extensions like the topic-sensitive PageRank computation [Haveliwala, 2002] and personalized PageRank in ER graphs [Jeh and Widom, 2003], use the random walk methodology for finding authority nodes in a graph. These earlier works are built on homogeneous networks and fail to capture the notion of heterogeneous signals of node similarities. While PageRank computed the authority scores or influence scores over nodes, the rooted-RW method [Liben-Nowell and Kleinberg, 2003] is a commonly used metric for node similarity computation with respect to a fixed node. In our work, we extend the authority computation of the PageRank algorithm, and the node similarity computation of the rooted-RW method to a multidimensional relation space. In the future, we aim to extend our work by implementing approximate rooted-RW efficiently using the map-reduce framework in [Bahmani, Chowdhury, and Goel, 2010], and also extend our work to evolving social graphs [Bahmani et al., 2012] of forum participants.

The edges in our multidimensional graph represent similarity between nodes. In [Liben-Nowell and Kleinberg, 2003], the authors compare the effectiveness of about fifteen different similarity measures including the rooted-RW measure for predicting links in a co-authorship network. The studies in [Faloutsos, McCurley, and Tomkins, 2004; Koren, North, and Volinsky, 2007] find subgraphs that represent the connection between any two nodes in the graph efficiently, and use these subgraphs to compute node proximity. However, these studies do not incorporate multiple user relations. In our work, we define edge weights using cosine similarity or frequency counts of common user behavior, and develop similarity scores across the entire social graph. In the future, we aim to infer the reply structure in posts [Seo, Bruce Croft, and Smith, 2011] and study different edge weights and similarity measures.

Recently, the PathSim algorithm [Sun et al., 2011] was built on heterogeneous graphs and learns node similarity using fixed length predefined paths. Another approach in [Lao and Cohen, 2010] finds answer nodes to a typed query by assigning weights to constrained paths along the random walks. These predefined paths fail to find relations between distant nodes and do not allow for a dynamic selection of relations or paths w.r.t. a fixed user for similarity computation. Our work focuses on finding node similarities in heterogeneous relation graphs, with egocentric relation weights.

Topical analysis of the content posted by users along with the social network of interactions has been successfully used to predict the cancer stage of patients [Jha and Elhadad, 2010]. Similarities between posts and the reply structure in threads has been used for topic detection and opinion leader discovery [Zhu, Wu, and Wang, 2010]. Textual content [Xu and Ma, 2006] or common forum participants [Chen, Zhang, and Wang, 2008] have been successfully used to introduce links between different threads in user forums. The study in [Chen, Zhang, and Wang, 2008] combines the IR scoring of the text in posts with the importance of the thread learned via links. Yet, very little research has focused on improving search over forums using the interpersonal relationships of forum participants. In our work, we re-rank keyword search results using the authority score of users and show that the new ranking yields higher overall relevance as perceived by crowd-sourced judges. As shown in [Alonso, Schenkel, and Theobald, 2010], crowd-sourced relevance assessment of XML search obtained via Mechanical Turk had comparable quality to INEX specialized judges.

Finding similar users in online data has significant social and economical applications like targeted advertising and marketing, online dating, and networking. Predicting links in social networks using similarity measures and user behavior across networks is useful in understanding and addressing future user needs. In our work, we learn the interpersonal relationships amongst forum participants to predict users who are likely to answer questions posed in a thread.

7 Conclusions

Online users interact with each other due to a variety of reasons ranging from shared interests, similar profiles, or same information need at the same time. In this paper, we describe a multidimensional similarity framework that builds a random walk using heterogeneous relations between users, enabling us to capture user similarity across a variety of reasons in a unified manner. Our heterogeneous framework captures egocentric similarities for a user in our data, and we leverage these similarities to make highly precise predictions on future interactions between users. Finding which users are likely to provide answers to questions posted on a forum improves user search experience in a personalized manner. In addition, we conducted user studies to assess the relevance of search results generated in response to keyword queries. We then enhance keyword search by reranking results retrieved by traditional IR scoring by learning the authority of users contributing to the forums. Our results demonstrate an improvement in overall search result relevance within as few as top-10 results, as perceived by crowd sourced judges. Thus, we uniformly capture multidimensional relations between users to enhance search and access over online forums.

8 Acknowledgments

This work was supported by NSF Grants IIS-0844935 and BCS-1027801, and a Google Research Award.

References

Adamic, L. A., and Adar, E. 2005. How to search a social network. *Social Networks* 27.

Alonso, O.; Schenkel, R.; and Theobald, M. 2010. Crowdsourcing assessments for xml ranked retrieval. In *ECIR*, 602–606.

Avrachenkov, K.; Litvak, N.; Nemirovsky, D.; and Osipova, N. 2007. Monte carlo methods in pagerank computation: When one iteration is sufficient. *SIAM J. Numer. Anal.* 45(2):890–904.

Bahmani, B.; Kumar, R.; Mahdian, M.; and Upfal, E. 2012. Pagerank on an evolving graph. In *KDD*, 24–32.

Bahmani, B.; Chowdhury, A.; and Goel, A. 2010. Fast incremental and personalized pagerank. *VLDB* 4(3):173–184.

Balmin, A.; Hristidis, V.; and Papakonstantinou, Y. 2004. Objectrank: Authority-based keyword search in databases. In *VLDB*, 564–575.

Carmel, D.; Zwerdling, N.; Guy, I.; Ofek-Koifman, S.; Har'el, N.; Ronen, I.; Uziel, E.; Yogev, S.; and Chernov, S. 2009. Personalized social search based on the user's social network. In *CIKM*, 1227–1236.

Chen, Z.; Zhang, L.; and Wang, W. 2008. Postingrank: bringing order to web forum postings. In *AIRS*, 377–384.

Croft, B. W.; Metzler, D.; and Strohman, T. 2009. *Search engines: Information retrieval in practice*.

Dawid, A. P., and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society* 28(1):20–28.

Faloutsos, C.; McCurley, K. S.; and Tomkins, A. 2004. Fast discovery of connection subgraphs. In *KDD*, 118–127.

Ganu, G., and Marian, A. 2013. One size does not fit all: Multi-granularity search of web forums. In *CIKM*.

Haveliwala, T. H. 2002. Topic-sensitive pagerank. In *WWW*, 517–526.

Jeh, G., and Widom, J. 2003. Scaling personalized web search. In *WWW*, 271–279.

Jha, M., and Elhadad, N. 2010. Cancer stage prediction based on patient online discourse. In *BioNLP*, 64–71.

Kekäläinen, J., and Järvelin, K. 2002. Using graded relevance assessments in ir evaluation. *JASIST* 1120–1129.

Koren, Y.; North, S. C.; and Volinsky, C. 2007. Measuring and extracting proximity graphs in networks. *TKDD*.

Lao, N., and Cohen, W. W. 2010. Relational retrieval using a combination of path-constrained random walks. *Mach. Learn.* 81(1):53–67.

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

Manning, C. D.; Raghavan, P.; and Schtze, H. 2008. *Introduction to Information Retrieval*.

Page, L.; Brin, S.; Motwani, R.; and Winograd, T. 1999. The pagerank citation ranking: Bringing order to the web.

Pehcevski, J. 2006. Relevance in xml retrieval: The user perspective. In *SIGIR*.

Seo, J.; Bruce Croft, W.; and Smith, D. A. 2011. Online community search using conversational structures. *Information Retrieval* 14(6):547–571.

Sun, Y.; Han, J.; Yan, X.; Yu, P. S.; and Wu, T. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *VLDB* 992–1003.

Xu, G., and Ma, W.-Y. 2006. Building implicit links from content for forum search. In *SIGIR*, 300–307.

Zhu, T.; Wu, B.; and Wang, B. 2010. Extracting relational network from the online forums: Methods and applications. In *ICEMMS*, 424–427.