Finding Experts By Semantic Matching of User Profiles

Rajesh Thiagarajan, Geetha Manjunath, Markus Stumptner

HP Laboratories
HPL-2008-172

Keyword(s):
semantics, similarity matching, ontologies, user profile

Abstract:
Extracting interest profiles of users based on their personal documents is one of the key topics of IR research. However, when these extracted profiles are used in expert finding applications, only naive text-matching techniques are used to rank experts for a given requirement. In this paper, we address this gap and describe multiple techniques to match user profiles for better ranking of experts. We propose new metrics for computing semantic similarity of user profiles using spreading activation networks derived from ontologies. Our pilot evaluation shows that matching algorithms based on bipartite graphs over semantic user profiles provide the best results. We show that using these techniques, we can find an expert more accurately than other approaches, in particular within the top ranked results. In applications where a group of candidate users need to be short-listed (say, for a job interview), we get very good precision and recall as well.

External Posting Date: October 21, 2008 [Fulltext] Approved for External Publication
Internal Posting Date: October 21, 2008 [Fulltext]

To be presented at PICKME 2008, a ISWC Workshop, Karlruhe, Germany, to be published in CEUR Workshop proceedings in Oct 2009

Finding Experts By Semantic Matching of User Profiles

Rajesh Thiagarajan[1], Geetha Manjunath[2], and Markus Stumptner[1]

[1] Advanced Computing Research Centre, University of South Australia
{cisrkt,mst}@cs.unisa.edu.au
[2] Hewlett-Packard Laboratories, India
geetha.manjunath@hp.com

Abstract. Extracting interest profiles of users based on their personal documents is one of the key topics of IR research. However, when these extracted profiles are used in expert finding applications, only naive text-matching techniques are used to rank experts for a given requirement. In this paper, we address this gap and describe multiple techniques to match user profiles for better ranking of experts. We propose new metrics for computing semantic similarity of user profiles using spreading activation networks derived from ontologies. Our pilot evaluation shows that matching algorithms based on bipartite graphs over semantic user profiles provide the best results. We show that using these techniques, we can find an expert more accurately than other approaches, in particular within the top ranked results. In applications where a group of candidate users need to be short-listed (say, for a job interview), we get very good precision and recall as well.

1 Introduction

The problem of finding experts on a given set of topics is important for many lines of business e.g., consulting, recruitment, e-business. In these applications, one common way to model a user is with a *user profile* which is a set of topics with weights determining his level of interest. When used for personalization, these user profiles matched with a retrieved document (may be a search result) for checking its relevance to him. A similar matching technique can be used for expert finding as well - wherein we first formulate the requirement (query) as an expected profile of the expert who is sought after. Expert finding is then carried out by matching the query profile with the available/extracted expert user profiles.

In the above context, automatic extraction of topics of expertise (interest) of a person based on the documents authored (accessed) by the person through information extraction techniques is well known. However, when these extracted profiles are used for expert finding, the profile matching is often carried out by applying traditional content matching techniques which miss most potential candidates if the query is only an approximate description of the expert (as is usually the case). In this paper, we propose and evaluate multiple approaches for semantic matching of user profiles to enable better expert-finding in such cases.

Let us briefly look at the challenges in comparing user profiles. User profiles are generally represented in the *bag-of-words* (BOW) format - a set of weighted terms that describe the interest or the expertise of a user. The most commonly used content matching technique is cosine similarity - cosine between the BOW vector representing the user profile and that of the document to match. Although this simple matching technique suffices in a number of content matching applications, it is well known that considering just the words leads to problems due to lack of semantics in the representation. Problems due to polysemy (terms such as *apple, jaguar* having two different meanings) and synonymy (two words meaning almost the same thing such as *glad* and *happy*) can be solved if profiles are described using semantic concepts instead of words. Once again simple

matching techniques can be used on these *bags-of-concepts* (BOC). However, these approaches fail to determine the right matches when there is no direct overlap/intersection in the concepts. For example, do two users with *Yahoo* and *Google* in their respective profiles have nothing in common? There does seem to be an intersection in these users' interests for Web-based IT companies or web search tools! Such overlaps are missed as current approaches work under the assumption that the profile representations (BOW) contain all the information about the user. As a result, relationships that are not explicit in the representations are usually ignored. Furthermore, these mechanisms cannot handle user profiles that are at different levels of granularity or abstractions (e.g., *jazz* and *music*) as the implicit relationship between the concepts is ignored.

In this paper, we solve the above issues in user profile matching through effective use of ontologies. We define the notion of semantic similarity between two user profiles to consider inherent relationships between concepts/words appearing in their respective BOW representation. We use the process of *spreading* to include additional related terms to a user profile by referring to an ontology (Wordnet or Wikipedia) and experiment with multiple techniques to enable better profile matching. We propose simple metrics for computing similarity between two user profiles with ontology-based Spreading Activation Networks (SAN). We evaluate multiple mechanisms for extending user profiles (set and graph based spreading) and semantic matching (set intersection and bipartite graphs) of profiles. We show the effectiveness of our user profile matching techniques for accuracy in expert-ranking as well as candidate selection. From a given set of user profiles, our bipartite-graph based algorithms can accurately spot an expert just within its top three ranks. In applications where a group of candidate users need to be found (for a job interview), we get very good precision and recall as well.

The organization of the rest of this document is as follows. We describe different related research efforts for profile building and ontology-based semantic matching techniques in section 2 followed by a brief section giving some background and definitions needed to understand our solution. An overview of the our spreading process is presented in Section 4. We present our new similarity measures in Section 5. We describe our evaluation procedure for expert finding in Section 6 and share our improved results. We summarize our contributions and state possible future work in Section 7.

2 Related Work

Determining interest profiles of users based on their personal documents is an important research topic in information extraction and a number of techniques to achieve this have been proposed. Expert finding techniques that combine multiple sources of expertise evidence such as academic papers and social citation network have also bee proposed [1]. User profiles have been extracted using multiple types of corpora - utilizing knowledge about the expert in Wikipedia [2], analysing the expert's documents [3–5], and analysing openly accessible research contributions of the expert [6]. Use of Wikipedia corpus to generate semantic user profiles [7] have been seen. Pre-processing the profile terms by mapping terms to such ontology concepts prior to computing cosine similarity has been shown to yield better matching [3]. A number of traditional similarity measurement techniques such as the cosine similarity measure or term vector similarity [8, 9], Dice's coefficient [10] and Jaccard's index [11] are used in profile matching. For example, Jaccard's index is used in [2] to match expert profiles constructed using Wikipedia knowledge.This approach will not determine a semantic inexact match when there is no direct overlap in the concepts in the two user profiles. Use of knowledge obtained from

an ontology,in our solution, enables similarity checks when there are no direct overlaps between user profiles and, therefore, result in more accurate similarity measurements.

The problem of automated routing of conference papers to their reviewers is a somewhat related problem to that of expert finding. Most of the current approaches to that problem use a group of papers authored by reviewers to determine their user profile and perform routine content matching (similar to personalization) to determine whether a paper is fit to be reviewed by that user [12]. The expert finding task introduced by TREC 2005 [13] requires one to provide a ranked list of the candidate experts based on the web data provided. Our attempt is to handle the problem of choosing the best expert given a description of a hypothetical expert (set of topics with weights) and a set of user profiles of candidate experts.

Use of ontologies to derive new concepts that are likely to be of interest to the user through semantic spreading activation networks has been studied as well [14–17, 5]. Previous studies have shown that the spreading process improves accuracy and overcomes the challenges caused by inherent relationships and Polysemy in word sense disambiguation process [15, 16] and ontology mapping [17]. We use this spreading process to facilitate the semantic similarity computation. We build on the spreading process used in [5] to learn user preferences in order to drive a personalized multimedia search. The learning process utilizes ontologies as a means to comprehend user interests (in BOW format) and establishes the need to consider related concepts to improve search quality. While the results in [5] suggest that personalized search is of better quality in comparison to normal search, they do not show whether the consideration of related terms contributes to these improvements. On the other hand, we show that our spreading process indeed improves the accuracy of our new similarity measures and in the particular context of user profile matching.

In [18] the document descriptions are preprocessed into a Concept Forest (CF) representation by determining related synsets from Wordnet. The similarity between 2 entities is then the Jaccard coefficient between their respective CFs. However, since only *is-a* relationships from an ontology are used, similarities that are contributed by other relationships are not considered. We use a semantic network constructed with related terms where edges can be qualified to handle different relationships with their respective semantics.

A number of approaches have already been proposed to determine the similarity between two ontology concepts (or words). These determine similarity by: measuring the path distance between them [19], evaluating shared information between them [20], recursively matching sub-graphs [21], combining information from various sources [22], analysing structure of the ontology [23], and combining content analysis and web search [24]. A few other measures are evaluated in [25]. While all these approaches are only able to determine closeness between two concepts (or words), we compute similarity between two weighted sets of concepts (or words). One of our algorithms use the simple path measure described in [19] over a bipartite graph to determine such a set intersection.

A number of IR approaches that use SAN to improve search are presented in [26]. One of our similarity measures that processes the semantic network post spreading the BOWs builds on these earlier works. Our work differs from earlier works in the treatment of the results of the activation process. While the previous work utilizes the results of the activation to rank documents with respect to a query, our work maps an aggregate of the activation results to a similarity value. Knowledge from an ontology is used to extend the BOW with terms that share important relationships with original terms to improve document retrieval is presented in [4]. Our work on set spreading is somewhat similar

to this but we further explore the notion of computing similarity by optimal concept matching in bipartite graphs and using SAN.

3 Background

In this section, we formally define and explain some terms used in the rest of the document.

Definition 1 (User Profile). *An user profile, u is a set of binary tuples $\{\langle t_1, w_1\rangle, \ldots, \langle t_n, w_n\rangle\}$ where t_i are the terms that describes the user and w_i denotes the importance of t_i in describing the user. We use $terms(u)$ to denote the set of terms t_i in the profile u.*

Cosine Similarity: The BOW representation is typically used for computing cosine similarity between the user profiles. If the vector representation of a user profile u_j is $\overrightarrow{V}(u_j)$ and the *Euclidean length* ($|\overrightarrow{V}(u_j)|$) of an entity u_j is $\sqrt{\sum_{i=1}^{n} w_i^2}$, the similarity of the entities u_j and u_k is

$$sim_{cos}(u_j, u_k) = cos(\overrightarrow{V}(u_j), \overrightarrow{V}(u_k)) = \frac{\overrightarrow{V}(u_j) \cdot \overrightarrow{V}(u_k)}{|\overrightarrow{V}(u_j)||\overrightarrow{V}(u_k)|} \tag{1}$$

Spreading: Spreading is the process of including the terms that are related to the original terms in an user profile by referring to an ontology. Let us study the earlier mentioned simple example of two users having *google* and *yahoo* in their profile in detail to understand the spreading process better.

Example 1. Consider computing the similarity of the following users
- $u_1 = \{\langle google, 1.0\rangle\}$, and
- $u_2 = \{\langle yahoo, 2.0\rangle\}$.

A simple intersection check between the profiles result in an empty set (i.e. $u_1 \cap u_2 = \emptyset$) indicating their un-relatedness (cosine similarity is 0). However, if we were to manually judge the similarity of these two users we would give it a value greater than 0. This is because we judge the similarity not just by considering the two terms from the profiles but also by considering the relationships that might exist between them due to our prior knowledge. We are able to establish the fact that both $google$ and $yahoo$ are search engine providers.

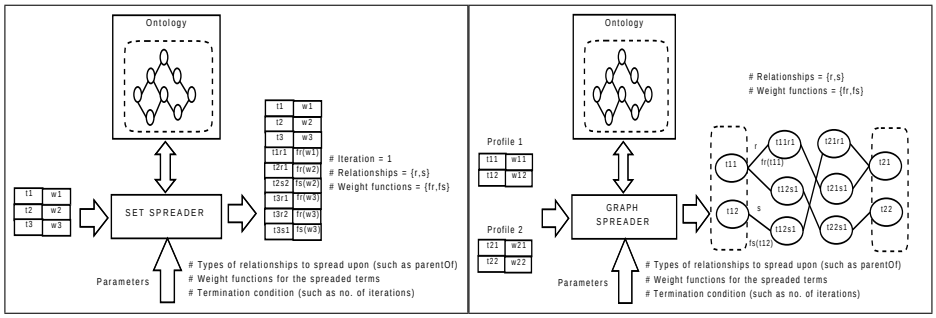
Now let us see the effectiveness of spreading in the similarity computation process in the same example. Spreading the profiles u_1 and u_2, by referring to Wikipedia parent category relationship, extends the profiles to
- $u_1' = \{\langle google, 1.0\rangle, \langle internet\ search\ engines, 0.5\rangle\}$, and
- $u_2' = \{\langle yahoo, 2.0\rangle, \langle internet\ search\ engines, 1.0\rangle\}$.

The simple intersection check results in a non-empty set (i.e. $u_1' \cap u_2' \neq \emptyset$) indicating their relatedness (cosine similarity is 0.2). The result of the spreading (i.e. the inclusion of the related term $internet\ search\ engines$) process makes sure that any relationship that exists between the profiles are taken into consideration.

4 Spreading to Create Extended User Profiles

In this section, we describe two techniques to compute and represent the extended user profiles (see example of section 3) using an ontology. An ontology $\mathcal{O}$ represents human

knowledge about a certain domain as concepts, attributes and relationships between concepts in a well-defined hierarchy. It is usually represented as a graph where nodes are the concepts and edges are the relationship labelled with the type of relationship. For the purpose of profile spreading we assume that all the terms t_i describing an entity are mappable to concepts in a reference ontology. For example, all the terms t_i in a BOW representation of a user profile maps to a concept in the Wordnet ontology. Given a term t_i, the spreading process utilizes $\mathcal{O}$ to determine the terms that are related to t_i (denoted as $related_{\mathcal{O}}(t_i)$). Although spreading the profiles with related terms allows for a comprehensive computation, uncontrolled addition of all the related terms leads to the dilution of the profiles with noise or unrelated terms. This dilution may have negative implications on the computation process where the similarity in the noise may contribute to the similarity values between entities. It is therefore desirable to have control over the types of relationships to be considered during this spreading process.



(a) Set Spreading (b) Graph Spreading

Fig. 1: Two Schemes for Profile Spreading

The weights of the new related terms are proportional to the weights of the original term as the weight w_i of a term t_i indicates the importance of the term within a user profile. However, during spreading the weights of the related terms should differ according to the semantics of the relationships on the edge. For example, spreading based on Wikipedia may be limited to only spreading along the parent categories. We therefore use a set of linear influence functions, one per relationship-type (role/property of an ontology), to control the spreading process. For example, a spreading process based on Wordnet limited to types synonym and antonym can have functions $t_{ij} = w_i \times 0.9$ and $t_{ij} = w_i \times -0.9$ respectively. We propose two schemes for representing the related terms post-spreading: extended set and semantic network.

4.1 Set Spreading

Depicted in Figure 1a, set spreading is a process of extending an user profile such that the related terms, which are determined with respect to an ontology, are just appended to the original set of terms. Set spreading is an iterative process. After each iteration, the related terms from the previous iterations are appended to the profile. The spreading process is terminated if there are no related terms to spread the profile with or after a fixed number of iterations.

4.2 Graph spreading

Shown in Figure 1b, graph spreading is the process where terms from two profiles and the related terms are build into a semantic network (SAN). Unlike set spreading, graph spreading preserves the relationship between a term in a profile and its related term in the form of a graph edge. This allows consideration of relationships based on their semantics on the same network. Graph spreading terminates like set spreading, or if there exists a path between every pair of the term nodes from the two profiles. This condition best suits the ontologies that have a top root element which subsumes the rest of the elements in the ontology. For example, Wordnet based spreading can be tuned to employ this termination condition when path from individual terms to the root suffices to terminate the spreading. In less rigorous ontologies such as the Wikipedia category graph may not be able to support this condition as there may not be a single root. In such a case, the spreading process is terminated if there exists at least one path from every node that belongs to the smallest of the two profiles to the nodes in the other profile. We describe the complete details of the two spreading algorithms in our technical report [27].

5 Similarity Computation

In this section, we describe the complete details of our variant metrics to compute semantic similarity using ontologies.

5.1 Set-based Measure

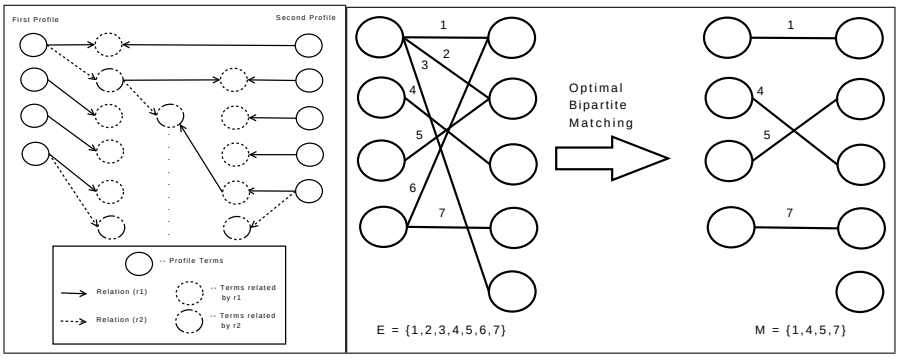
Set spreading process enriches the profiles by appending the related terms in order to capture all the relationships between the terms. For set spreading, the same cosine similarity technique defined in Equation 1 is applicable to compute similarity between the extended BOWs or BOCs. Set spreading-based similarity computation begins by measuring similarity of the original profiles, and proceeds by incrementally extending the profiles until termination while computing the similarity between profiles at every iteration.

5.2 SAN-based measure

This similarity computation metric is inspired by the abundant work that exists in the area of semantic search especially by techniques that process a SAN (e.g., [26, 15]). We focus on similarity computation techniques that use a SAN resulting from graph spreading process (see figure 2a for an overview of SAN structure). Following the construction of the semantic network the similarity values are computed either by reducing the graph to a bipartite graph or by activating the graph with an activation strategy. We have implemented both these techniques for evaluation. A brief introduction to the activation process is presented below. For a more detailed discussion the reader is pointed to [15].

The SAN activation process is iterative. Let $A_j(p)$ denotes the activation value of node j at iteration p. All the original term nodes corresponding to the tuples in a user profile t_j take their term weights w_j as their initial activation value $A_j(0) = w_j$. The activation value of all the other nodes are initalized to 0. In each iteration,

- Every node propagates its activation to its neighbours.
- The propagated value is a function of the nodes current activation value and weight of the edge (see [15]) that connects them (denoted as $O_j(p)$).



(a) SAN Building Process (b) Bipartite Graph Matching (Hungarian Algorithm)

Fig. 2: SAN-based Similarity Computations

After a certain number of iterations when the termination condition is reached, the highest activation value among the nodes that are associated with each of the original term node is retrieved into a set $ACT = \{act_1, act_2, ..., act_{n+m}\}$[3]. The aggregate of the these activation values can be mapped to the similarity between the profiles under the intuition that the nodes with higher activation values are typically the ones that have value contributions from both the profiles and hence should contribute more to similarity and vice versa. Therefore, the similarity value is the sum of the set ACT normalized to a value between 0 and 1. The SAN-based similarity between two profiles u_1 and u_2 where $max(ACT)$ is the highest activation value is

$$sim_{san}(u_1, u_2) = \frac{\sum_{\forall act_i \in ACT} act_i}{|ACT| \times max(ACT)} \tag{2}$$

5.3 Similarity Computation by Matching Bipartite Graph

A key insight here is that by omitting the intermediate related nodes and considering only the path length between the nodes representing the original profile terms, the semantic network can be converted to a bipartite graph (shown on the left side of Figure 2b). The nodes of the first profile and second profile are the two vertex sets of the bipartite graph where the edge denotes the length between the original term nodes as obtained from the semantic network. Once the bipartite graph is derived, we are able to apply standard algorithms for optimal matching of the bipartite graph. Our similarity measures based on optimal bipartite matching operates under the simple notion that the nodes with higher weights and that are closely located contribute more to the similarity of the entities and viceversa.

Each node v_i^u in the semantic network is a pair $\langle t_i, w_i \rangle$ where $u = 1$ or 2 denoting which user's profile term the node represents. The $path(v_i^1, v_j^2)$ denotes the set of edges between two nodes v_i^1 and v_j^2 in the semantic network. All the edges between any two nodes with different terms in the semantic network have uniform weights $\forall e \in path(v_i^1, v_j^2)$ set $wt(e) = 1$ where $wt(e)$ denotes the weight of the edge e. For any two vertices v_i^1 and v_j^2 the distance between the them is

$$len(v_i^1, v_j^2) = \begin{cases} 0, & \text{if } t_i = t_j \\ \sum_{\forall e_k \in path(v_i^1, v_j^2)} wt(e_k), & \text{otherwise} \end{cases}$$

[3] n and m are the number of terms in the first and second profile respectively.

Definition 2 (Bipartite Representation). *The bipartite graph representation G of the profiles u_1 and u_2 is a pair $G = \langle V, E \rangle$ where*

- *$V = V^1 \cup V^2$ where V^1 denotes the vertices from the first profile u_1 and V^2 denotes the vertices from the second profile u_2*
- *$V^1 = \{v_1^1, v_2^1, \ldots, v_n^1\}$ and $V^2 = \{v_1^2, v_2^2, \ldots, v_m^2\}$ where $n \leq m$ and $v_i^k = \langle t_i^k, w_i^k \rangle$ is a term.*
- *$E = \{e_{11}, e_{12}, \ldots, e_{ij}\}$ where $i = \{1, 2, \ldots, n\}$, $j = \{1, 2, \ldots, m\}$ and $len(v_i^1, v_j^2)$ denotes the path length between then vertices v_i^1 and v_j^2.*

Given the bipartite representation G, the optimal matching $E' \subseteq E$ between two vertex sets is computed using the Hungarian Algorithm [28]. The optimal bipartite graph (shown on the right side of Figure 2b) is $G' = \langle V, E' \rangle$ where $E' \subseteq E$ such that $\sum_{\forall e_{ij} \in E'} len(v_i^1, v_j^2)$ is optimal. Given the weights of vertices in the representation $W^{12} = \{w_1^1, \ldots, w_i^1, w_1^2, \ldots, w_j^2\}$, these are normalized (value [0-1]) to $W^{12'} = \{w_1^{1'}, \ldots, w_i^{1'}, w_1^{2'}, \ldots, w_j^{2'}\}$ where $\forall_{w_l^{k'} \in W^{12'}}$ is $w_l^{k'} = \frac{w_l^k}{\sum w_l^k}$.

Aggregate Path Distances: Abiding by our notion that the closer nodes with higher weights contribute more to the similarity value, we present three (slightly different) path length aggregation measures for empirical evaluation. The path distance of an edge e_{ij} in the optimal bipartite graph is defined as

$$path(e_{ij}) = \begin{cases} 1, & \text{if } len(v_i^1, v_j^2) \text{ is } 0 \\ 0, & \text{if } len(v_i^1, v_j^2) \text{ is } \infty \\ \frac{w_i^{1'} \times w_j^{2'}}{len(v_i^1, v_j^2)}, & \text{otherwise} \end{cases}$$

The Euler path distance of an edge e_{ij} in the optimal bipartite graph is defined as

$$eupath(e_{ij}) = \begin{cases} 1, & \text{if } len(v_i^1, v_j^2) \text{ is } 0 \\ 0, & \text{if } len(v_i^1, v_j^2) \text{ is } \infty \\ \frac{w_i^{1'} \times w_j^{2'}}{e^{len(v_i^1, v_j^2)}}, & \text{otherwise} \end{cases}$$

The Euler half path distance of an edge e_{ij} in the optimal bipartite graph is defined as

$$euhalf(e_{ij}) = \begin{cases} 1, & \text{if } len(v_i^1, v_j^2) \text{ is } 0 \\ 0, & \text{if } len(v_i^1, v_j^2) \text{ is } \infty \\ \frac{w_i^{1'} \times w_j^{2'}}{e^{\left(\frac{len(v_i^1, v_j^2)}{2}\right)}}, & \text{otherwise} \end{cases}$$

The aggregate distance of all the matching edges of the bipartite graph is given by the sum of their path distances.

Similarity Measures: Given two user profiles u_1 and u_2, the similarity between them using aggregate path distances in the optimal bipartite graph are defined as follows.

$$(3) \quad sim_{path}(u_1, u_2) = \frac{\sum_{\forall e_{ij} \in E'} path(e_{ij})}{min(size(terms(u_1)), size(terms(u_2))) \times max(path(e_{ij}))}$$

$$(4) \quad sim_{eupath}(u_1, u_2) = \frac{\sum_{\forall e_{ij} \in E'} eupath(e_{ij})}{min(size(terms(u_1)), size(terms(u_2))) \times max(eupath(e_{ij}))}$$

$$(5) \quad sim_{euhalf}(u_1, u_2) = \frac{\sum_{\forall e_{ij} \in E'} euhalf(e_{ij})}{min(size(terms(u_1)), size(terms(u_2))) \times max(euhalf(e_{ij}))}$$

5.4 Compound Similarity Measures

While the term vector similarity technique considers only intersecting terms while computing similarity, when two profiles actually intersect this measure is quite accurate. Therefore, we propose compound similarity measures where the similarity between intersecting profile terms are computed using cosine similarity (Equation 1), and the similarity between the remaining profile terms are computed using our bipartite graph approaches (Equations 3, 4, and 5). More details follow.

Given two user profiles u_1 and u_2, the intersecting profile parts are denoted as u'_1 and u'_2 such that $terms(u'_1) = terms(u'_2) = terms(u_1) \cap terms(u_2)$. The remaining non-overlapping profile parts are denoted as $\hat{u_1}$ and $\hat{u_2}$ such that $terms(\hat{u_1}) = terms(u_1) \setminus terms(u_2)$ and $terms(\hat{u_2}) = terms(u_2) \setminus terms(u_1)$. The combined size of the two profiles is denoted as $N = |terms(u_1)| + |terms(u_2)|$. The size of the intersecting profile parts is $N' = |terms(u'_1)| + |terms(u'_2)|$. The size of the non-overlapping profile parts is $\hat{N} = |terms(\hat{u_1})| + |terms(\hat{u_2})|$.

The compound similarity measure based on sim_{path} (Equation 3) is

$$sim^{C}_{path} = \frac{sim_{cos}(u'_1, u'_2) \times N' + sim_{path}(\hat{u_1}, \hat{u_2}) \times \hat{N}}{N} \tag{6}$$

The compound similarity measure based on sim_{eupath} (Equation 4) is

$$sim^{C}_{eupath} = \frac{sim_{cos}(u'_1, u'_2) \times N' + sim_{eupath}(\hat{u_1}, \hat{u_2}) \times \hat{N}}{N} \tag{7}$$

The compound similarity measure based on sim_{euhalf} (Equation 5) is

$$sim^{C}_{euhalf} = \frac{sim_{cos}(u'_1, u'_2) \times N' + sim_{euhalf}(\hat{u_1}, \hat{u_2}) \times \hat{N}}{N} \tag{8}$$

6 Evaluation and Results

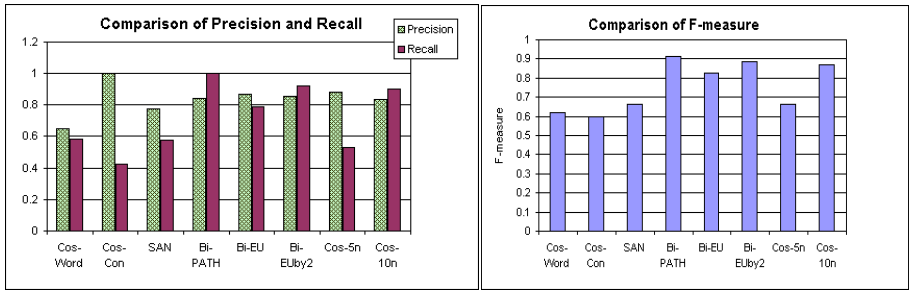
We evaluate the different algorithms described in the previous section in the context of expert finding. We use an inhouse-built software called Profile Builder to generate expert profiles using techniques described in [7] to create profiles by analysing the documents (such as web pages visited by the expert). Both the BOW (word profiles) and BOC (terms are Wikipedia concepts; Wiki profiles) representations of the experts are generated by the profile builder software. An expert finding query is correspondingly in the form of either a BOW or a BOC. For a given query profile, matching expert profiles are determined by computing similarity between the expert profile and the query profile.

Measure	Description
COS-Word	Cosine similarity measure between expert and query BOW profiles (Equation 1)
COS-Con	Cosine similarity measure between expert and query BOC profiles (Equation 1)
COS-5n	Mean cosine similarity between BOC profiles after 5 iterations of set spreading
COS-10n	Mean cosine similarity between BOC profiles after 10 iterations of set spreading
Bi-PATH	Compound similarity measure after graph spreading as defined in Equation 6
Bi-EU	Compound similarity measure after graph spreading as defined in Equation 7
Bi-EUby2	Compound similarity measure after graph spreading as defined in Equation 8
SAN	Similarity measure after graph spreading as defined in Equation 2

Table 1: Glossary of the Similarity Measures

A pilot study conducted as a part of the evaluation process interviewed 10 participants with expertise in different fields of computer science research. From each of

the participants, 5 to 10 documents that in the participant's opinion best describe their research were collected. Along with the documents, the participants were asked to give 5 keywords for each of their document that in their opinion best described the document. Since these keywords somewhat described the expertise of the participants, they were used by the participants to provide two similarity judgments. We believe this approach reduces the subjectivity in judging similarity and gives us more realistic values for comparison. Every participant was asked to judge the similarity between their profile and other profiles. Additionally, each of the participants judged the similarity between every pair of profiles (third person view). The mean of the subjective judgments provided by the participants were used as the base/reality values to evaluate our similarity measures. The comparison of the computed similarity value with the reality values were actually made across all user pairs. However, for evaluating the algorithms in the context of expert finding, we consider a user q to represent the query profile and evaluate similarity results of user pairs (q, x) where x is every other user (experts).



(a) Precision and Recall Observations (b) F-measure Observations

Fig. 3: Effectiveness of Similarity Measures for Expert Search (Threshold-based Match)

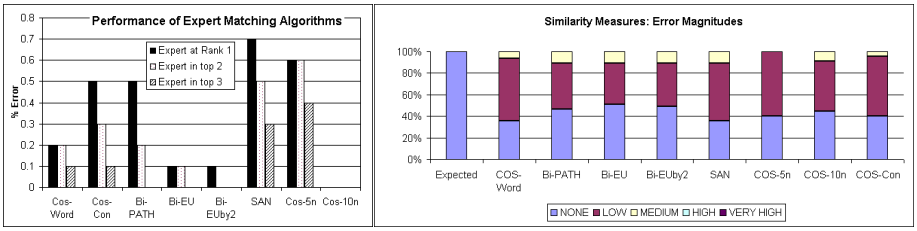
We first evaluate the effectiveness of our similarity measures in the context of short-listing a group of experts (eg: for recruitment interview). Here the selected expert profiles are those that exceed a pre-determined similarity threshold. We repeat the search for 10 query profiles over the derived expert profiles using all the approaches listed in Table 1. Figure 3 shows the results from our candidate search process where we measured precision, recall, and F-measure[4] of all the approaches. In short, precision represents the fraction of the correctly determined experts from those selected by our algorithms (based on how many of the matched results are the experts we wanted to get). Recall represents the effectiveness of the algorithm to find all experts (based on how many experts did we miss). F-measure is a compound measure, basically a harmonic mean of precision and recall. Higher values are better. From Figure 3, we are able to make the following observations.

- All the metrics based on bipartite graph mapping (Bi-*) work very well over the standard cosine similarity measurement techniques (Cos-Word and Cos-Con).
- The accuracy of set-based measures increases with the increase in the number of spreading iterations (Cos-10n performs much better than Cos-5n).
- The precision of all our approaches are almost equal while the recall varies.
- Our algorithms show significant improvements in recall when compared with the standard approaches. Our approaches Bi-* and Cos-10n exhibit upto 20% improvement.

[4] We use the standard definitions of Precision, Recall and F-measure as defined in [8]

– The recall of our Bi-PATH approach is 100% while Bi-EUby2 and Cos-10n approaches exhibit around 90% recall.
– Spreading with 5 iterations (Cos-5n) is almost equal performance to other path-based/reachability conditions for termination in a general semantic search approach (SAN). This may be suggestive of the maximum diameter of the relevant subgraph consisting of the user's concepts.
– The precision of the cosine similarity approach considering semantic concepts (Cos-Con) is 100% however it has the poorest recall of around 40%. It shows only right experts but may miss 60% of other experts.

We conclude that user profile matching through use of ontologies increases the accuracy of expert finding process and bipartite based compound measures Bi-PATH and Bi-EUby2 matches performs the best.



(a) Errors in Top 3 Selected Experts (b) Error Magnitudes over all User Pairs

Fig. 4: Accuracy of Expert Search using Different Algorithms

We next analyse the accuracy of our approaches in the context of determining an expert within the top three selections returned by our expert finding process. Here, we choose the top 3 experts based on reality values and compare those with the top 3 matches using our computed similarity metrics. The error percentage of all the approaches in this scenario is presented in Figure 4a - lower the better. As seen, our bipartite-graph based algorithms can accurately spot an expert just within its top three ranks. The Cos-Word approach has a 20% chance that the first expert returned is not the required expert. Among the top three ranks, Cos-Word still does not guarantee that a matching expert will be found because there is a 10% chance that the top three results are false positives.The set-based measures Cos-10n is the best among all the approaches with the high possibility that all the top three ranks are positive expert matches.

In order to check the effectiveness of the algorithms as a similarity measure for matching any two users, we show the magnitude of error across all the 100 user pairs. Analysis of the error magnitudes[5], as shown in Figure 4b, that our spreading based computations yield more accurate similarity judgements than the simple vector based counterparts as our bipartite approaches have the maximum number of *no errors* as a generic matching of two user profiles.

7 Conclusion

We presented a number of similarity computation measures that improve the expert finding process by accurately matching expert profiles for a query. Our approach utilises

[5] Difference in slabs, for example expected = VERY HIGH, observed = VERY LOW results in VERY HIGH error magnitude

spreading as a means to capture the semantics of the terms in user profiles. The evaluation of the similarity measures shows the improvements in accuracy that is achieved over existing traditional similarity computation methods. Our bipartite graph based measures out perform all other algorithms for the specific use case of expert finding. We plan to explore use of more sophisticated techniques [25] to measure similarity at single concept level and study their effects on the profile matching. Additionally, we would like to extend the approaches to automatically use other domain ontologies (not just Wordnet or Wikipedia) from a ontology repository like Swoogle.

References

1. Bogers, T., Kox, K., van den Bosch, A.: Using Citation Analysis for Finding Experts in Workgroups. In: Proc. DIR. (2008)
2. Demartini, G.: Finding Experts Using Wikipedia. In: FEWS. (2007)
3. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: Proc. IJCAI. (2007)
4. Nagarajan, M et al.: Altering document term vectors for classification: ontologies as expectations of co-occurrence. In: WWW. (2007)
5. Cantador, I et al.: A multi-purpose ontology-based approach for personalised content filtering and retrieval. In: Advances in Semantic Media Adaptation and Personalization. (2008)
6. Jung, H., Lee, M., Kang, I.S., Lee, S., Sung, W.K.: Finding topic-centric identified experts based on full text analysis. In: FEWS. (2007)
7. Banerjee, S., Ramanathan, K., Gupta, A.: Clustering short texts using Wikipedia. In: SIGIR. (2007)
8. Manning, C., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press (2008)
9. Dhyani, D., Ng, W.K., Bhowmick, S.S.: A survey of web metrics. ACM Comput. Surv. **34**(4) (2002)
10. van Rijsbergen, C.J.: Information Retrieval. Butterworth (1979)
11. Hamers, L et al.: Similarity measures in scientometric research: The Jaccard index versus Salton's cosine formula. Inf. Process. Manage. **25**(3) (1989)
12. Biswas, H., Hasan, M.: Using Publications and Domain Knowledge to Build Research Profile: An Application in Automatic Reviewer Assignment. In: Proc. ICICT. (2007)
13. : TREC Enterprise Track. http://www.ins.cwi.nl/projects/trec-ent/wiki/ (2005)
14. Castells, P., Fernández, M., Vallet, D., Mylonas, P., Avrithis, Y.S.: Self-tuning personalized information retrieval in an ontology-based framework. In: OTM Workshops. (2005)
15. Tsatsaronis, G., Vazirgiannis, M., Androutsopoulos, I.: Word sense disambiguation with spreading activation networks generated from thesauri. In: Proc. IJCAI. (2007)
16. Véronis, J., Ide, N.: Word Sense Disambiguation with Very Large Neural Networks Extracted from Machine Readable Dictionaries. In: In Proc. COLING. (1990)
17. Mao, M.: Ontology mapping: An information retrieval and interactive activation network based approach. In: Proc. ISWC. (2007)
18. Wang, J.Z., Taylor, W.: Concept forest: A new ontology-assisted text document similarity measurement method. In: Proc. of WI. (2007)
19. Pedersen, T., Patwardhan, S., Michelizzi, J.: WordNet:: Similarity - Measuring the Relatedness of Concepts. In: Proc. AAAI. (2004)
20. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: Proc. IJCAI. (1995)
21. Zhu, H., Zhong, J., Li, J., Yu, Y.: An Approach for Semantic Search by Matching RDF Graphs. In: FLAIRS. (2002)
22. Li, Y., Bandar, Z., McLean, D.: An approach for measuring semantic similarity between words using multiple information sources. IEEE Trans. Knowl. Data Eng. **15**(4) (2003)
23. Schickel-Zuber, V., Faltings, B.: Oss: A semantic similarity function based on hierarchical ontologies. In: Proc. IJCAI. (2007)

24. Iosif, E., Potamianos, A.: Unsupervised semantic similarity computation using web search engines. In: Proc. of WI. (2007)
25. Budanitsky, A., Hirst, G.: Evaluating wordnet-based measures of lexical semantic relatedness. Computational Linguistics **32**(1) (2006)
26. Crestani, F.: Application of Spreading Activation Techniques in Information Retrieval. Artif. Intell. Rev. **11**(6) (1997) 453–482
27. Thiagarajan, R., Manjunath, G., Stumptner, M.: Computing semantic similarity using ontologies. HP Labs Technical Report HPL-2008-87 (2008)
28. Kuhn, H.W.: The Hungarian Method for the Assignment Problem. Naval Research Logistic Quarterly **2** (1955) 83–97