LinkedVis: Exploring Social and Semantic Career Recommendations

Svetlin Alex Bostandjiev, John O'Donovan, Tobias Höllerer
Department of Computer Science
University of California, Santa Barbara
{alex, jod, holl}@cs.ucsb.edu

ABSTRACT

This paper presents LinkedVis, an interactive visual recommender system that combines social and semantic knowledge to produce career recommendations based on the LinkedIn API. A collaborative (social) approach is employed to identify professionals with similar career paths and produce personalized recommendations of both companies and roles. To unify semantically identical but lexically distinct entities and arrive at better user models, we employ lightweight natural language processing and entity resolution using semantic information from a variety of end-points on the web. Elements from the underlying recommendation algorithm are exposed through an interactive interface that allows users to manipulate different aspects of the algorithm and the data it operates on, allowing users to explore a variety of "what-if" scenarios around their current profile. We evaluate LinkedVis through leave-one-out accuracy and diversity experiments on a data corpus collected from 47 users and their LinkedIn connections, as well as through a supervised study of 27 users exploring their own profile and recommendations interactively. Results show that our approach outperforms a benchmark recommendation algorithm without semantic resolution in terms of accuracy and diversity, and that the ability to tweak recommendations interactively by adjusting profile item and social connection weights further improves predictive accuracy. Questionnaires on the user experience with the explanatory and interactive aspects of the application reveal very high user acceptance and satisfaction.

Author Keywords

User Interfaces; Visual Knowledge Representation; Hybrid Recommender Systems; Data Integration; Social Web

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation: User Interfaces–*graphical user interfaces (GUI), user-centered design*

IUI'13, March 19–22, 2013, Santa Monica, CA, USA.

INTRODUCTION

Many of today's job search websites recommend jobs by simple matching of text descriptions to the job seeker's professional profile. The higher the correlation between the job description requirements and the job seeker's qualifications the more likely it is for a job to get recommended. This content-based matching method works to some degree, but it is narrow and ignores a wealth of social connections and other metadata that can potentially yield more interesting and diverse recommendations. For example, by looking at the jobs followed by other professionals with similar qualifications to some target user. We argue that it is valuable to take this so-called "social" information into account. We propose that a typical collaborative filtering approach can be used to identify professionals with similar backgrounds and reveal personalized job opportunities that are otherwise hidden. However, we find that in many cases there is insufficient overlap to generate useful predictions, so we look to the semantic web to bootstrap and augment the traditional approach. For example, a person graduating college might have a skill set very similar to someone who already has years of working experience after school, but these skill sets may be represented very differently in the two user-defined profiles. The jobs that the professional has held are of potential interest to the new graduate, and better techniques are needed to align these profiles as potential recommendation partners. Using these techniques, the *LinkedVis* interface helps users discover professionals with similar backgrounds who can in turn highlight potentially interesting companies and specific roles. For a job-seeker exploring possible career paths, it is typically useful to explore what-if scenarios. For example, one might want to answer questions such as: "What new jobs could I apply for if I became proficient in C++ programming?". LinkedVis enables users to answer such questions by leveraging underlying social and semantic information in a transparent manner via an interactive user interface. A video demo of the user interface can be viewed at [1]. To assess the performance of *LinkedVis*, a supervised laboratory study was performed using participants' real LinkedIn profiles and network connections. The study focused on aspects of recommendation accuracy, diversity and user satisfaction in three different interaction and control conditions within the *LinkedVis* system.

Contributions

The list below highlights the three core contributions in this paper. For each contribution, a discussion on design, implementation, evaluation and results are provided later.

[1] `http://www.youtube.com/watch?v=w7Q4JoYyEF4&hd=1`

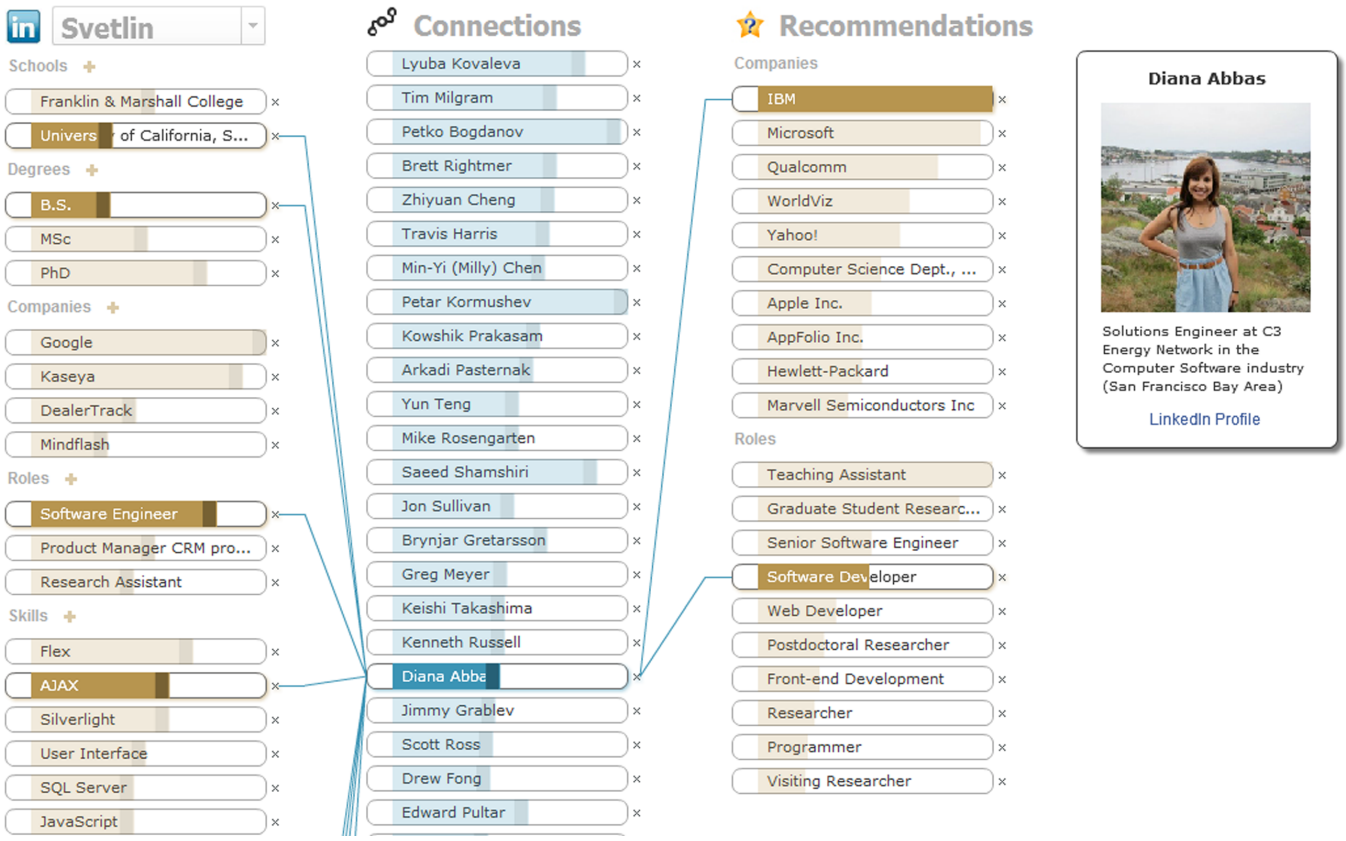

Figure 1. Screenshot of the LinkedVis Recommender System.

- *A novel approach that uses NLP and entity resolution steps to improve traditional collaborative filtering for job recommendations.* An evaluation of our hybrid system is performed against purely social approaches in terms of recommendation accuracy, diversity, and computational costs. Offline experiments and a user study show that the hybrid approaches produce recommendations with higher accuracy and greater diversity at a higher computational cost that, for the purpose of our real-time user evaluations were absorbed in a one-time pre-processing step per user.

- *Real-time visualizations of the relationships between profile items, social connections, and the resulting recommendations, supporting explanation of the recommendation process.* A supervised user study of LinkedVis shows that recommending career opportunities through a user interface increases user satisfaction and provides explanation and transparency in the recommendation process.

- *An interactive control mechanism for the recommendation process at the data level and the algorithmic level.* An interaction method was designed to provide control over system settings that are generally not modifiable in modern career search websites. Users were able to modify the weights of their professional qualifications and connections. The interface was also designed to support what-if scenarios and let users see how adding new qualifications to their profile affects their suitability for certain companies and roles – this feature received the highest user satisfaction in the post-study questionnaire. Overall, our evaluation shows that interaction with the system in real time can improve recommendation accuracy and user experience.

RELATED WORK

This section provides a discussion of related research in the following two areas. 1) Recommender system algorithms and their hybrids, and 2) the role of interaction and visualization for recommendation systems.

Recommender Systems

Much research has focused on ways to automatically filter and personalize content for web users [24]. So called recommender systems aim to tailor a user's information by predicting the right item at the right time. There are a wide variety of approaches to recommendation, such as those algorithms we use through Amazon, YouTube, Netflix, Pandora and many other popular online applications that personalize content for users. Core techniques include content-based recommendation [16] [9] which is a rudimentary approach that simply matches text-based descriptions of a candidate item to those in a target users profile. These methods tend to suffer from problems such as narrowness, since they recommend items that are textually similar to those already in a user's profile. Content-based approaches also fall short on non-machine analyzable data such as music for example, where detailed text

descriptions of content are generally not available. Collaborative filtering (CF) approaches [7][14][19][20][22], attempt to model the normal social process of asking a friend for a recommendation. In brief, CF algorithms generate recommendations based on the opinions of a neighborhood of users who have rated similarly in the past. CF methods perform well in rich data environments where there is plenty of overlap in user preference. They are capable of making good predictions on non-machine analyzable data such as music, since they rely on human evaluation of the candidate items. For the same reason, these methods tend to be less narrow than content-based approaches, and can produce serendipitous recommendations [20]. Another relevant class of recommender system algorithms use knowledge from domain specialists to produce recommendations. These are commonly known as knowledge-based or "expert" recommender systems [21].

Hybrid Approaches

Traditional recommender system techniques such as collaborative, content-based, and knowledge-based filtering, each have unique sets of strengths and limitations. For example, CF suffers from sparsity and early rater problems [20], while content-based approaches suffer from narrowness and require text descriptions. However, a hybrid approach can use one approach to make predictions where the other fails, resulting in a more robust recommender system [16]. Researchers have attempted to augment the traditional process of collaborative filtering with additional information such as trust models [18] to improve accuracy and robustness of the recommendations, while more recent work in [13, 17] refine the trust models used in [18]. Burke [6] proposes a taxonomy of different classes of hybrid systems and hybridization designs. For example recommendation algorithms can work in parallel before combining their results, may be pipelined such that the output of one algorithm is the input of the next, or may be combined into one monolithic algorithm. *LinkedVis* falls into the parallelized design class, since our approach firstly generates predictions from social sources, followed by semantic sources, and then applies a hybridization strategy afterwards.

Interfaces for Recommender Systems

Recommendations are now being delivered on a variety of devices and platforms, from small hand-held devices to 60 inch TV screens showing Netflix predictions on Roku or AppleTV devices for example. Accordingly. researchers have begun to place more focus on the role of the user interface interface in the recommendation process. Furthermore, evaluation mechanisms for recommender systems no longer focus solely on improving recommendation accuracy, but consider effects of interface elements in the overall evaluation process. For example, allowing for inspectability and control [26, 15] through interface elements improves overall acceptance of a recommendation [15, 4, 25]. However, there is some disagreement in the community as to what the appropriate levels of inspectability and control should be. [8] promotes the simpler UI, as do Netflix, while others [4, 15] argue the benefits of having greater transparency and control. The authors believe that it is important to have control available, and that the simple knowledge that control is available can have a bearing on a user's satisfaction with a recommender system. In the context of our LinkedVis system (Fig. 1), the interface can potentially be refined to just the predictions shown in the right column, with the remaining inspecability and control elements only shown on-demand, for those who wish to explore the system in more detail.

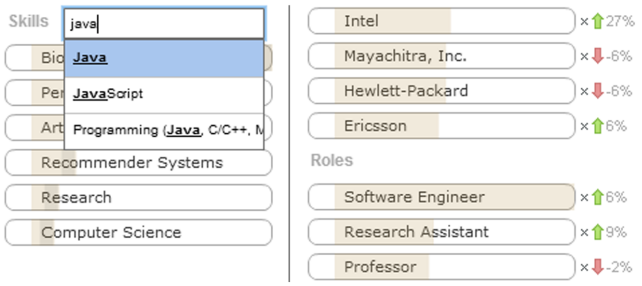

Figure 2. Illustration of the "what-if" scenario feature of *LinkedVis*. Users are able to add new hypothetical entities to their profile (left) and see how that affects the recommendations (right), i.e. whether the new entity brings them closer to their dream jobs.

DESIGN

The *LinkedVis* user interface provides a simple explanation of our collaborative filtering approach for discovering relevant entities: personal qualifications, connected professionals, companies, and roles (Figure 1).

Visualization

The interface is split into three distinct columns:

- *Profile*: The left column presents a semantically classified summary of the user's profile, such as educational background, previous and current jobs, and professional skills.
- *Connections*: The middle column consists of a list of connected professionals that have similar backgrounds to the user.
- *Recommendations*: The right column reveals grouped recommendations the user might be interested in, such as companies and roles.

An additional panel (right of the Recommendations column) provides detailed information about the last clicked entity.

Interaction

Each entity is weighted by our algorithms on a scale from 0 to 1 and is visually represented by a draggable slider. Via these sliders users are able to change the input weight of entities in their profile suggesting that for example their Ph.D. is more important than a part-time job they held at the dining commons. Weights of connected people can be changed to express how similar the user perceives those professionals to be. As the user drags a slider in his profile the weights of connected people change in real-time, and when weights of connected professionals are changed that also affects the recommendations in real-time. Users are able to explore what-if scenarios by adding qualifications to their profile and seeing how that affects their connections and recommendations. This is done via the "+" icon next to each profile group. Figure 2 illustrates this feature.

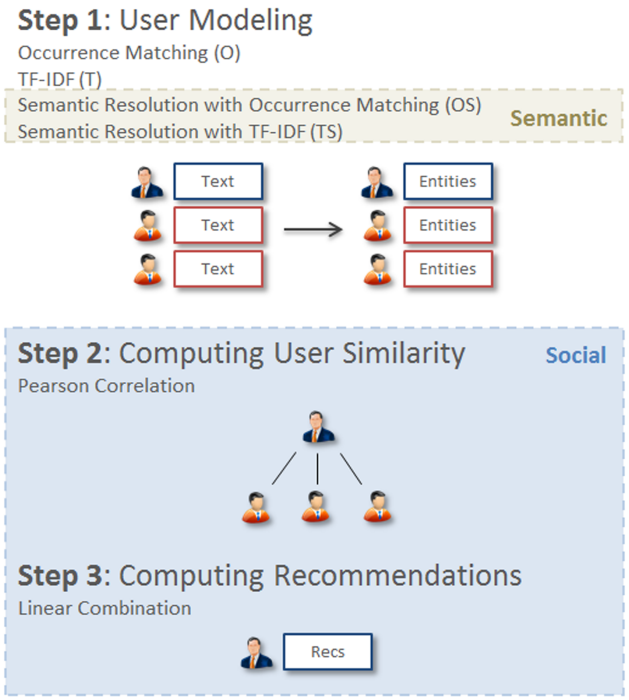

Figure 3. Three-step approach of *LinkedVis*.

APPROACH

As in most collaborative filtering methods, we need to model user profiles, design a metric for user similarity, and compute recommendations (Figure 3). These three steps are described in the following subsections.

User Modeling

`LinkedIn.com` is currently the dominant professional social network with more than 150 million users registered in more than 200 countries and territories[2]. We use the LinkedIn API to identify user's first connections. We then crawl the LinkedIn HTML pages related to the user and their connections. We parse the text out of each HTML file so that each user/connection is associated with a flat text file. These text files serve as the input for the algorithms described below. We treat each user/connection as a bag of weighted entities that describe it. For example, user X can be described by "MIT" and "software engineering" with respective weights of 0.8 and 0.6. We have designed four different approaches to model users and come up with these entities and their weights based on the user-associated text files:

- *Occurrence Matching (O)* : This simple model assigns a weight of 1 to each term (word) related to each user.

- *TF-IDF (T)* : This model uses "Term Frequency - Inverse Document Frequency" (TF-IDF)[3] to assign weights to each term. The input for the traditional TF-IDF approach is the corpus of text files describing the user and his connections.

- *Semantic Resolution with Occurrence Matching (OS)* : This model and the next involve alignment of semantic metadata, more specifically"semantic entity resolution". The idea is that we want to extract semantically meaningful entities for each user as opposed to just words. For example, the entity "Doctor of Philosophy" is more meaningful and descriptive than the terms "Doctor", "of", and "Philosophy". Our approach involves using the concept of "noun phrases" which are good topic markers in paragraph-long text descriptions [1, 12]. For each user, we identify noun phrases using a maximum entropy part-of speech tagger[4]. Then we resolve the noun phrases to real life semantic entities. Wikipedia presents the most evolved semantic graph in terms of completeness and non-redundancy [10], and therefore it is a good resource for entity-resolution. In order to map noun phrases to Wikipedia articles we rely on external callouts to a search engine. Previous work has been successful in turning search engines into knowledge bases [2, 3]. We identify the first article within the English Wikipedia that the search engine returns. This resolution approach helps us map phrases to semantically meaningful entities, i.e. "PhD", "Ph.D.", and "Doctor of Philosophy" will all map to the Wiki article `http://en.wikipedia.org/wiki/Doctor_of_Philosophy`. After we model each user in terms of semantic entities we assign a weight of 1 to each entity similarly to the *Occurrence Matching (O)* model.

- *Semantic Resolution with TF-IDF (TS)* : This model first uses the semantic entity resolution described in the previous model and then TF-IDF to assign weights to the semantic entities.

Computing User Similarity

After we model the user and his connections we want to rank the connection by how similar they are to the user in terms of their professional background. In order to compute user similarity we use a social collaborative filtering approach generally used in the field of recommender systems. We have adapted Pearson's correlation coefficient formula to account for the fact that entities in users' profiles are binary and do not contain scaled ratings. The similarity of each LinkedIn connection to the active user is given by:

$$W_{con_i} = \frac{TWCE_{user,con_i}}{\sqrt{TWE^2_{user} \cdot TWE^2_{con_i}}} \quad (1)$$

where $TWCE_{x,y}$ is the total weight of the entities x and y have common, and TWE_x is the total weight of user x's entities.

Computing Recommendations

The last step is to compute what companies and roles can be recommended based on the user similarities computed above. We compute the weight of each recommendation as the sum of the weights of all connections that contain the recommendation, i.e. the weighted sum.

[2] `http://en.wikipedia.org/wiki/LinkedIn`
[3] `http://en.wikipedia.org/wiki/Tf*idf`
[4] `http://www-nlp.stanford.edu/software`

EVALUATION

We carried out an evaluation of the system and its user interface via offline experiments and a user study. The evaluation focused on whether the system provides the contributions described in the beginning of this paper. We were interested in finding whether applying semantic entity resolution prior to performing social collaborative filtering improves the quality of the resulting recommended entities. We also measured the effectiveness of the visualization and interaction techniques provided by the user interface.

All evaluation was conducted on a Windows 7 64-bit computer (Intel Core i7-2670QM CPU at 2.2Ghz with 12GB of RAM) and the supervised user study used a 24-inch display with 1920x1080 pixel resolution. The web application and database were hosted locally.

EVALUATION VIA OFFLINE EXPERIMENTS

First, a set of offline experiments was performed over LinkedIn data collected for 47 users (27 male and 20 female, ages between 20 and 38) and their connections. 25 users were students and the rest were employed in industry. Users had 114.3 connections on average (SD=97.33). LinkedIn user connections data is not public and is only available through the LinkedIn API. In order to get a user's connections data the user had to approve a LinkedIn application that we created.

Leave-One-Out

We performed a leave-one-out cross-validation to compare the accuracy of the user models listed above. For each user in the collected data we extracted the companies and roles they have held. The 47 users had each worked for 3.82 companies on average, and served 3.86 different roles. We ran a leave-one-out analysis on that data using each user model. We measured how often the left-out entity appeared in the top N recommended entities, for N equals 1, 3, 5, 10, and 20. The analysis on companies and roles was done separately. The medians are presented in Figure 4.

In order to evaluate whether the semantic augmentation contributed to more accurate recommendations we performed paired tests. The leave-one-out accuracy data was not normally distributed so we chose a non-parametric counterpart of to the paired-samples t-test, the Wilcoxon Signed-Rank test. The analysis summarized in Table 1 was performed for the top 20 recommendations.

The Wilcoxon Signed-Rank tests revealed significant improvement in accuracy both on companies and roles. In terms of companies, for the *O* (Mdn=0.125) and *OS* (Mdn=0.25) models: Z=-2.973, p=0.001, and for the *T* (Mdn=0.33) and *TS* (Mdn=0.365) models: Z=-2.712, p=0.003. Moreover, in terms of roles, for the *O* (Mdn=0) and *OS* (Mdn=0.14) models: Z=-2.662, p=0.003, and for the *T* (Mdn=0.33) and *TS* (Mdn=0.33) models: Z=-2.122, p=0.014. Values ranged from 0 to 1 for all models.

The results from the analysis for the top 1, 3, 5, and 10 companies are summarized in Table 2. The semantic augmentation contributed to more accurate recommendations ($p<0.05$) in all cases except two: Z=-1.104, p=0.180 for top 10 companies and Z=-1.610, p=0.094 for top 1 roles for O-OS.

	Companies	Roles
Model	Mdn	Mdn
O	0.125	0
T	0.33	0.33
OS	0.25	0.14
TS	0.365	0.33

		Companies		Roles	
Model 1	Model 2	Z	P Val	Z	P Val
O	OS	-2.973	0.001	-2.662	0.003
T	TS	-2.712	0.003	-2.122	0.014

Table 1. Results from Wilcoxon Signed-Rank tests comparing the accuracy between the *O* and *OS* models and between the *T* and *TS* models over top 20 recommendations (df=46). Medians for the four models are presented in the top table and statistical significance values are described in the bottom table.

			Companies		Roles	
Model 1	Model 2	Top N	Z	P Val	Z	P Val
O	OS	20	-2.973	0.001	-2.662	0.003
		10	-1.104	0.180	-2.733	0.002
		5	-2.805	0.002	-2.733	0.002
		3	-2.088	0.022	-2.763	0.002
		1	-2.177	0.011	-1.610	0.094
T	TS	20	-2.712	0.003	-2.122	0.014
		10	-2.244	0.016	-3.131	<0.001
		5	-2.270	0.013	-3.237	<0.001
		3	-2.343	0.009	-4.077	<0.001
		1	-2.095	0.021	-3.114	0.001

Table 2. Results from Wilcoxon Signed-Rank tests comparing the accuracy between the *O* and *OS* models and between the *T* and *TS* models over top 1, 3, 5, 10, and 20 recommendations (df=46).

Diversity

As discussed in recommender system literature [11], predictive accuracy is not a sufficient measure of the performance of a recommender system. The "popular item" effect is a common problem in recommender systems, where predictions tend to include many popular items and not enough esoteric items. However, it is the correct prediction of these esoteric items that can have the biggest positive influence on user satisfaction. For example, it is not desirable for our system to recommend only big established companies but also startups.

We performed a diversity analysis on the different user models. For each company and role entity in the entire corpus of data we assigned a "popularity" score, i.e. a frequency score based on the number of the times the entity occurs in the whole corpus of data (Figure 5 left). The corpus contained 47 main users plus their connections, which accumulated to 5372 LinkedIn users total. The most popular roles were "Intern" and "Software Engineer" occurring 455 and 337 times respectively, and the most popular company was 'UC Santa Barbara" with 372 occurrences, which reflects the fact that the greater portion of our users were UC Santa Barbara students.

After we calculated frequencies for all companies and roles in the corpus, for each user we measured the total frequency of the top 20 recommended entities produced by each user model. The averages are presented in Figure 5 (right).

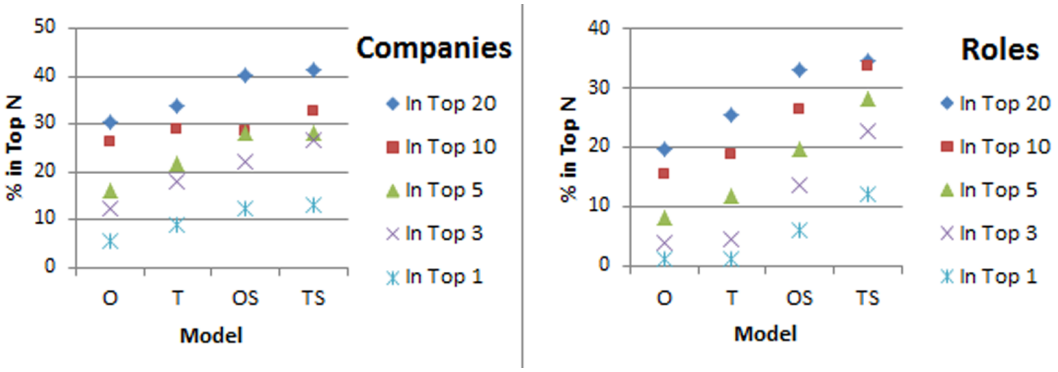

Figure 4. Leave-One-Out analysis results showing medians of how often the left-out entity appeared in the top N recommendations.

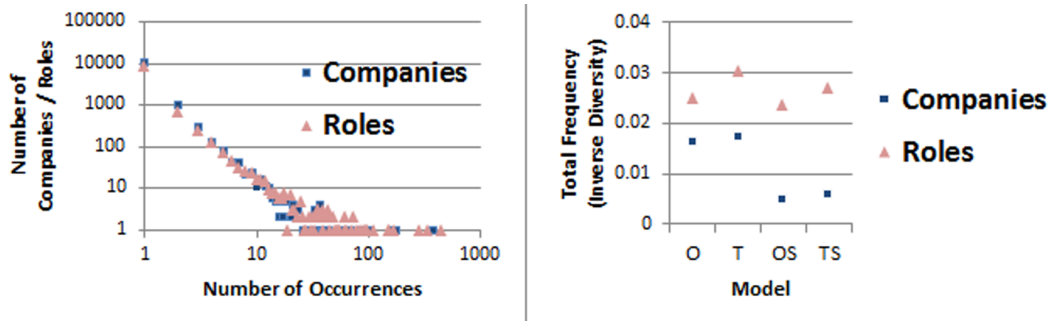

Figure 5. Results from the Diversity analysis. On the left, we show a histogram of the number of occurrences of companies and roles in the whole corpus of data (5372 LinkedIn users). The majority of the companies and roles, 85.8% and 85.3% respectively, occurred only once (10665 out of 12425 companies total and 8582 out of 10066 roles total). On the right, for each model we calculated the average frequency per user of his top 20 recommended companies / roles.

	Companies		Roles	
Model	Mean	StDev	Mean	StDev
O	0.016	0.010	0.025	0.012
T	0.017	0.010	0.030	0.013
OS	0.005	0.004	0.024	0.011
TS	0.006	0.004	0.027	0.010

		Companies		Roles	
Model 1	Model 2	t	P Val	t	P Val
O	OS	10.662	<0.001	1.744	0.044
T	TS	11.249	<0.001	3.771	<0.001

Table 3. Results from paired-samples t-tests comparing the frequency (inverse diversity) between the *O* and *OS* models and between the *T* and *TS* models (df=46).

Paired-samples t-tests were conducted to compare frequency (inverse diversity) between the *T* and *TS*, and between *O* and *OS* for both companies and roles. The results are summarized in Table 3.

The t-tests revealed significant differences in diversity both on companies and roles. In terms of companies, for the *O* (M=0.016, SD=0.010) and *OS* (M=0.005, SD=0.004) models: t(46)=10.662, p<0.001, and for the *T* (M=0.017, SD=0.10) and *TS* (M=0.006, SD=0.004) models: t(46)=11.249, p<0.001). Moreover, in terms of roles, for the *O* (M=0.025, SD=0.012) and *OS* (M=0.024, SD=0.011) models: t(46)=1.744, p=0.044, and for the *T* (M=0.030, SD=0.013) and *TS* (M=0.027, SD=0.010) models: t(46)=3.771, p<0.001).

These results suggest that the semantic augmentation produced more diverse recommendations.

EVALUATION VIA A USER STUDY

We conducted a supervised user study to obtain more insights on our main contributions in recommendation accuracy, explanation, and interaction. We gathered data implicitly and explicitly via pre- and post-study questionnaire. In the post-study questionnaire we asked questions to judge the perceived transparency and explanation of the system, and the quality of interaction.

Another goal of the study was to compare how different user models perform relative to one another in terms of recommendation accuracy. We compared all methods described in Section , plus an interactive method that let users change entity weights and improve the data. We looked for statistical significance of whether semantic computing can add value to standard user models, and whether the interaction method outperforms the best hybrid. As a base for the interaction method we chose the best performer out of the four non-interactive methods. In a pilot study involving six users we determined that *Semantic Resolution with TF-IDF (TS)* produces the best recommendations, and hence we named our interactive method *Interaction after Semantic Resolution with TF-IDF (ITS)*.

Setup

Each of the five methods produced a ranked list of recommended entities (companies and roles). We asked users to

evaluate the relevance of these entities on a Likert scale (1 to 5 stars), which produces a "rating" for each entity. Users rated the top 10 results for each of the five methods, compiled into one randomized list, at the end of the study. We measured the "utility" of each ranked list of entities using Breeze's *R-Score* "utility" metric [5, 23] to determine a utility score for the list. The metric assumes that the value of recommendations decline exponentially based on position in the recommended list. The utility of a given recommendation list for user u is given by:

$$R_u = \sum_j \frac{max(r_{ui_j} - d, 0)}{2^{\frac{j-1}{\alpha-1}}} \quad (2)$$

where i_j is the item in the jth position, r_{ui} is user u's rating of item i, (i.e. 1 to 5 stars), d is Breese's "don't care" threshold (experimentally chosen as 2 stars in our setting), and α is the half-life parameter, which we set to 1.5, controlling the exponential decline of the value of positions in the ranked list.

We performed one-way repeated measures ANOVA analyses for both companies and roles, with the independent variable being the user model and the dependent variable being the utility of the produced list of entities.

Note that users might have been biased to inflate relevance ratings of entities with higher value, for example giving a higher rating to "Google" than "LocalSoft" even though the user is better suited for the latter. This is due to the fact that bigger companies might be more desirable to work for. Our diversity analysis described in Section measures the level of entity popularity produced by each model. The analysis shows that the semantic models produce more diverse recommendations.

Participants

A total of 27 users participated in the user study, which lasted 41 minutes on average. The analyses used data from 20 participants as data for 7 participants was not logged correctly. We used the SONA[5] human subjects pool provided by the Department of Psychology at University of California, Santa Barbara to recruit participants. Out of the 20 participants 9 were male and 11 were female. Also, 15 were students: 5 pursuing a bachelors degree, 3 masters, and 7 doctorate, spanning 9 different majors. The other 5 participants were employed and working in the fields of the arts, sciences, and finance. The reported usage of LinkedIn was: 2 daily, 10 weekly, and 8 monthly. The way participants discover new career opportunities was mainly through personal connections, mailing lists, online job websites, and search.

Procedure

The study had four parts: a pre-computational step, a pre-study questionnaire, tasks, and a post-study questionnaire. It was required for each participant to have a LinkedIn account with at least 30 connections. Each participant was asked to approve a LinkedIn application through which we gathered their profile and connections data. Once the data was available we pre-computed all data necessary for the visualization. This step took no more than an hour for each user. The pre-study questionnaire was split into demographics and assessment of users' familiarity with and level of activity on LinkedIn and recommender systems. The following tasks were performed between the pre- and post-study questionnaires:

- *Preliminary Task*: In this task we explained what the system is about and asked users to familiarize themselves with the interface by clicking on a few entities and exploring how entities are connected.
- *Refinement Task*: Here we asked users to refine the visualization by changing some of the pre-computed weights for their profile items and connections. Users were able to see how the changes affected the weights of other entities in real-time.
- *What-if Scenario Task*: This task was designed for users to explore what-if scenarios. We asked users to add qualifications to their profiles that could potentially get them better jobs, such as new skills (i.e. Excel) and educational degrees (i.e. PhD), etc.
- *Rating Task*: Lastly, users were asked to rate the top 10 recommendations produced by each of the five approaches on a 1-to-5-star rating given in random order.

In the post-study questionnaire we asked questions related to transparency and explanation, and also interaction with the system.

		Companies				Roles			
Model 1	Model 2	Diff	Lower	Upper	P Val	Diff	Lower	Upper	P Val
OS	O	0.765	0.066	1.465	0.024	0.922	0.246	1.597	0.002
TS	T	0.774	0.075	1.474	0.022	0.772	0.097	1.448	0.016
ITS	TS	1.100	0.401	1.799	<0.001	1.306	0.631	1.981	<0.001

Table 4. Results from a Tukey post-hoc analysis of the user models: multiple comparisons of means with 95% family-wise confidence level

Recommendation Accuracy

We performed one-way repeated measures ANOVA analyses for both companies and roles, in which the independent variable was the user model and the dependent variable was the utility of the produced list of entities. Figure 6 shows the means of the five models over utility with 95% confidence intervals for companies and roles separately. The *Interaction after Semantic Resolution with TF-IDF (ITS)* model achieved the highest utility for both companies and roles, whereas, the simple *Occurrence Matching (O)* model reported the lowest utility. Mauchly's test did not show violation of sphericity against Model for companies (W(4)=0.39, p=0.06), and for roles (W(4)=0.43, p=0.10). The ANOVAs revealed a significant effect of the model variable on utility for both companies (F(4, 76)=205.59, p<0.001) and roles (F(4, 76)=215.08, p<0.001). To assess the statistical significance of pair-wise differences within our models, a Tukey post-hoc analysis was performed and the more interesting results were recorded in Table 4.

[5] `http://ucsb.sona-systems.com`

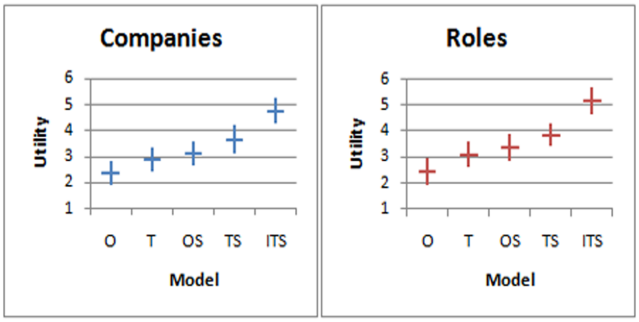

Figure 6. Plot of means of user models over utility with 95% confidence intervals.

Semantic vs. Non-semantic Results

In this section, we focus on the analysis whether our semantic models (*Semantic Resolution with Occurrence Matching (OS)* and *Semantic Resolution with TF-IDF (TS)*) perform better than the non-semantic models (*Occurrence Matching (O)* and *TF-IDF (T)*). The Tukey pair-wise test (Table 4) revealed that when both the *O* and *T* models were augmented with semantic computing, i.e. models *OS* and *TS*, there was an improvement in perceived recommendation accuracy for both companies and roles ((p=0.024, p=0.022) and (p=0.002, p=0.016) respectively). This result indicates that semantic computing can be successfully used to improve the results of a purely social computing approach. We did not find significant difference between the two non-semantic models, *O* and *T*, and between the two semantic models, *OS* and *TS*.

Interaction Results

In terms of interaction, the Tukey pair-wise test showed that the interactive model, *Interaction after Semantic Resolution with TF-IDF (ITS)*, performed better than the *TS* model itself ($p<0.001$ for companies and roles). We tested interaction only over our best performing model in order to simplify the study. Testing over all models would require a between-subjects study with a significantly larger participant pool. The result indicates that interaction with the visualization helps users get better recommendation. This is a somewhat intuitive result because users can see the recommended entities.

User Experience

Figure 7 presents the results from the post-study questionnaire. The goal of the post-study was to examine the perceived quality of explanation within the system and the level to which users believed interaction was useful. All positively-phrased questions received scores higher than 4 on a 1-5 scale. The highest score of 4.6 was received for the more general questions "Was the system easy to use?" and "Was the system fun to use?".

In terms of explanation and transparency, users perceived that the system helped them understand the basis for their recommendations (4.3) and they felt like they found connections with similar backgrounds to their own (4.3).

The interaction features of *LinkedVis* were also positively perceived in the study. The highest score (4.6) was achieved by

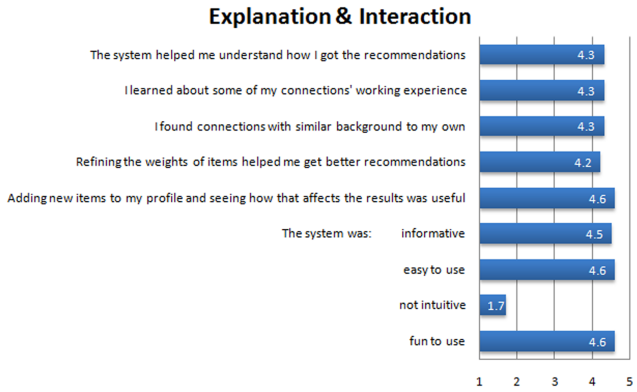

Figure 7. Post-study questionnaire results.

the what-if scenario feature of the user interface, i.e. the option to hypothetically add new entities to your profile and see how that affects the recommendations (Figure 2).

CONCLUSIONS AND FUTURE WORK

In this paper, we presented *LinkedVis*, a user interface for discovery of new career opportunities based on your professional social network. *LinkedVis* employs a collaborative filtering approach for finding similar connections in a target user's network, which is then augmented via semantic knowledge for user modeling and entity resolution. The interactive visual interface serves as an explanatory mechanism, and further, as an input modality for refining a user's profile and social connections and to exploring what-if scenarios in the career world.

We furthermore explored potential synergies that exist between social connections and associated semantic concept resolution for collaborative recommendation of jobs via the LinkedIn API.

A series of offline experiments and a user study were performed to evaluate the system in terms of integrating social network connections, semantic metadata, explanation and interaction into the recommendation process. The study results indicate that:

- Semantic entity resolution significantly increases user-rated recommendation accuracy and recommendation diversity in the job domain.
- Indication recommendation provenance and providing real-time illustration of the recommendation algorithm within a hybrid recommender system through a visual user interface resulted in high user satisfaction.
- Interaction with the social and semantic data relationships within a hybrid recommender system (allowing the adjustment of weightings for profile items and social connections, as well as enabling what-if analyses) can significantly improve recommendation accuracy and user experience.

The benefits of providing better explanation and control in recommender systems come at a cost. Every user interface

has a learning curve as users need to spend time to understand the system and be willing to tweak it in the first place. The learning curve can be alleviated through better user interface designs that appeal to large audiences. In this regard, the qualitative findings from our post-study user questionnaire are highly encouraging, in that they indicated high levels of user enthusiasm and engagement with the system.

However, no matter how intuitive a system's user interface is, many users will often be satisfied with being given just a final recommendation. To accommodate groups of users with and without an interest in parameter control, we are interested in user interfaces that scale fluidly between basic and advanced user modes. Future work in this area may focus on how control and explanation can be provided by minimal intrusion on the user, evaluated by large-scale user studies on variants of popular recommender systems augmented with advanced views that provide different degrees explanation and interaction on demand.

ACKNOWLEDGEMENTS

The authors would like to thank Cha Lee and Sibel Adali for their input on analysis methods and credibility indicators for microblogs, respectively. This work was partially supported by the U.S. Army Research Laboratory under Cooperative Agreement No. W911NF-09-2-0053; by NSF grant IIS-1058132; by the Intelligence Advanced Research Projects Activity (IARPA) via Air Force Research Laboratory contract number FA8650-10-C-7060; and by the U.S. Army Research Laboratory under MURI grant No. W911NF-09-1-0553; The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of ARL, AFRL, IARPA, NSF, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

REFERENCES

1. Bendersky, M., and Croft, W. B. Discovering key concepts in verbose queries. In *SIGIR*, S.-H. Myaeng, D. W. Oard, F. Sebastiani, T.-S. Chua, and M.-K. Leong, Eds., ACM (2008), 491–498.
2. Bernstein, M. S., Suh, B., Hong, L., Chen, J., Kairam, S., and Chi, E. H. Eddi: interactive topic-based browsing of social status streams. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, UIST '10, ACM (New York, NY, USA, 2010), 303–312.
3. Bostandjiev, S., O'Donovan, J., Hall, C., Gretarsson, B., and Höllerer, T. Wigipedia: A tool for improving structured data in wikipedia. In *Proceedings of the 2011 IEEE Fifth International Conference on Semantic Computing*, ICSC '11, IEEE Computer Society (Washington, DC, USA, 2011), 328–335.
4. Bostandjiev, S., O'Donovan, J., and Höllerer, T. Tasteweights: A visual interactive hybrid recommender. In *Proceedings of ACM RecSys '12 Conference on Recommender Systems* (Dublin, Ireland, 2012).
5. Breese, J. S., Heckerman, D., and Kadie, C. Empirical analysis of predictive algorithms for collaborative filtering. Morgan Kaufmann (1998), 43–52.
6. Burke, R. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction 12*, 4 (2002), 331–370.
7. Burke, R., Mobasher, B., Zabicki, R., and Bhaumik, R. Identifying attack models for secure recommendation. In *Beyond Personalisation Workshop at the International Conferece on Intelligent User Interfaces*, ACM Press (San Deigo, USA., 2005), 347–361.
8. Cremonesi, P., Garzotto, F., Negro, S., Papadopoulos, A., and Turrin, R. Comparative evaluation of recommender system quality. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, ACM (New York, NY, USA, 2011), 1927–1932.
9. Goldberg, K., Roeder, T., Gupta, D., and Perkins, C. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval 4*, 2 (2001), 133–151.
10. Halavais, A., and Lackaff, D. An analysis of topical coverage of wikipedia. *Journal of Computer-Mediated Communication 13*, 2 (2008), 429–440.
11. Herlocker, J. L., Konstan, J. A., Terveen, L. G., John, and Riedl, T. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems 22* (2004), 5–53.
12. Hulth, A. Improved automatic keyword extraction given more linguistic knowledge. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 03)* (2003), 216–223.
13. Kang, B., O'Donovan, J., and Höllerer, T. Modeling topic specific credibility on twitter. In *IUI* (2012), 179–188.
14. Karypis, G. Evaluation of item-based top-n recommendation algorithms. In *Proceedings of the 2001 ACM CIKM International Conference on Information and Knowledge Management, Atlanta, Georgia, USA* (2001), 247–254.
15. Knijenburg, B., Bostandjiev, S., O'Donovan, J., and Kobsa, A. Inspectability and control in social recommenders. In *Proceedings of ACM RecSys '12 Conference on Recommender Systems* (Dublin, Ireland, 2012).
16. Melville, P., Mooney, R. J., and Nagarajan, R. Content-boosted collaborative filtering for improved recommendations. In *Eighteenth national conference on Artificial intelligence*, American Association for Artificial Intelligence (Menlo Park, CA, USA, 2002), 187–192.
17. O'Donovan, J., Kang, B., Meyer, G., Hllerer, T., and Adali, S. Credibility in context: An analysis of feature distributions in twitter. In *In proceedings at the 2012 ASE/IEEE International Conference on Social Computing, SocialCom, Amsterdam, The Netherlands.*, IEEE (2012), 167–174.
18. O'Donovan, J., and Smyth, B. Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interfaces*, ACM (2005), 167–174.
19. O'Sullivan, D., Wilson, D. C., and Smyth, B. Improving case-based recommendation: A collaborative filtering approach. In *Proceedings of the Sixth European Conference on Case Based Reasoning.* (2002), LNAI 2416, p. 278 ff.
20. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of ACM CSCW'94 Conference on Computer-Supported Cooperative Work* (1994), 175–186.
21. Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B., Eds. *Recommender Systems Handbook*. Springer, 2011.
22. Sarwar, B. M., Karypis, G., Konstan, J. A., and Reidl, J. Item-based collaborative filtering recommendation algorithms. In *World Wide Web* (2001), 285–295.
23. Shani, G., and Gunawardana, A. Evaluating recommendation systems. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer, 2011, 257–297.
24. Shardanand, U., and Maes, P. Social information filtering: Algorithms for automating "word of mouth". In *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, vol. 1 of *Papers: Using the Information of Others* (1995), 210–217.
25. Tintarev, N., and Masthoff, J. Effective explanations of recommendations: user-centered design. In *Proceedings of the 2007 ACM conference on Recommender systems*, RecSys '07, ACM (New York, NY, USA, 2007), 153–156.
26. Tintarev, N., and Masthoff, J. Designing and evaluating explanations for recommender systems. *Recommender Systems Handbook* (2011), 479–510.