SIGCHI Conference Proceedings Format

1st Author Name
Affiliation
Address
e-mail address
Optional phone number

2nd Author Name
Affiliation
Address
e-mail address
Optional phone number

3rd Author Name
Affiliation
Address
e-mail address
Optional phone number

ABSTRACT

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION
Over the past 20 years recommender systems have emerged as an important solution to assist users in finding relevant products or information in a large collection of items ranging from news, movies, music, academic articles, hotels to jobs or even online dating. Nowadays, recommender systems are an essential component of many online services such as Amazon, Netflix, LinkedIn, or Facebook. The use of recommender systems extends also into the sphere of education and scientific research. However, most of such systems focus on recommending scientific literature or conference papers. There are systems for scientific literature exploration [4, 5], recommending papers to reviewing committee members [3], recommending citations of papers [9], building introductory research lists [6] or conference talks [13, 10].

In this paper, we describe NAME, a system that allows students to identify potential BSc and MSc project supervisors best matching students' interests. Currently, in our institution there is no centralized list of potential topics for the final BSc and MSc dissertations. Instead, students are expected to identify a faculty member as a dissertation supervisor and negotiate the dissertation topic with him or her. When searching for potential supervisors, most students simply resort to browsing webpages of faculty members, however, apart from it being a tedious exercise, the webpages are often not up-to-date or the information they contain is very scant. The NAME system facilitates this process by providing an aggregate and up-to-date information about the research interests of all the faculty members as well as an intuitive and easy to use search interface.

SYSTEM OVERVIEW
The primary goal of the system is to assist students in exploring what research areas are currently covered by the faculty members with the aim of identifying potential dissertation supervisors. Reinforcement learning (RL) methods as well as visualization allow the user to assign relevance scores to the displayed keywords through which the user can direct the search according to their interest. The inbuilt RL mechanism helps the system to form a model of the users interests and suggest appropriate keywords and researchers' names in the next search iteration. In this section, we describe the system design, its interface and the algorithms incorporated into the system.

Interface Design
The main idea behind the interactive interface is that instead of typing queries at each iteration, the user navigates through the contents by assigning weights to the keywords on the display, which results in new keywords appearing on the screen as well as a new set of researchers most strongly associated with the displayed keywords being presented to the user.

The search starts with the user typing in a query or/and selecting one or more of the existing keywords as a query 1. We display a set of the most frequent keywords that currently exist in the system to help students who might have difficulties formulating their initial query. Following the query, a set of keywords is displayed on the left hand-side of the screen with a set of names of researchers being displayed on the right hand-side of the screen 3. The user can score the displayed keywords by moving the dial above each keyword. The score ranges from 0 (not relevant) to 1 (highly relevant). The user can score as many keywords as she likes. Additionally, the user can indicate that a given keyword should be removed from future iterations by clicking the red cross next to it. After each iteration, new keywords and a modified list researchers' names are displayed. The search continues until the user is satisfied with the results.

When hovering the mouse pointer over a given keyword, names of researchers that are strongly associated with this keyword are highlighted on the right hand-side of the screen. By clicking on a researcher's name, a strip appears at the bottom left hand-side of the screen appears containing the researcher's photograph, the name of the research group they belong to and their contact details. Additionally, a list of the titles of papers written by the researcher appears below the researcher's name. When the mouse hovers over a paper title, the paper abstract appears on the left hand-side of the screen at the bottom replacing the strip with the information about the researcher.

By clicking on one of the paper titles, the student opens a new tab where papers similar to the one the student clicked

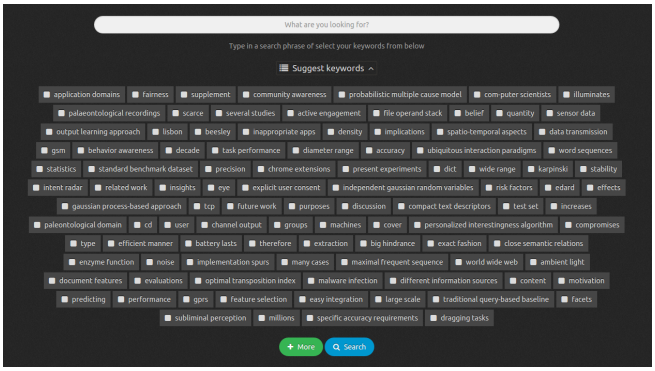

Figure 1. The initial query page of the system.

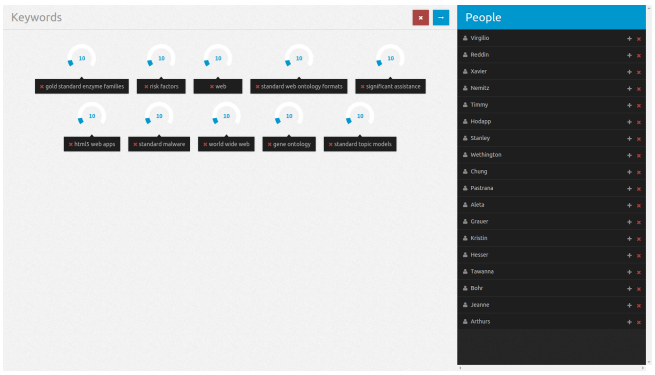

Figure 2. The interactive search interface.

on are displayed. The similarity is based on the cosine distance between the papers represented as vectors of length n, where n is the number of keywords in our database and each cell in the vector represents the number of times a given keyword appears in a given paper. This is beneficial to students who read a paper by a given researcher and would like to find papers on a similar topic written by other faculty members. Currently, we are working on a separate function that will allow the student to query the system by uploading a research paper that they found interesting (not necessarily written by our faculty staff member) and based on this paper, our system will suggest related keywords and researchers with research interests in line with the query paper.

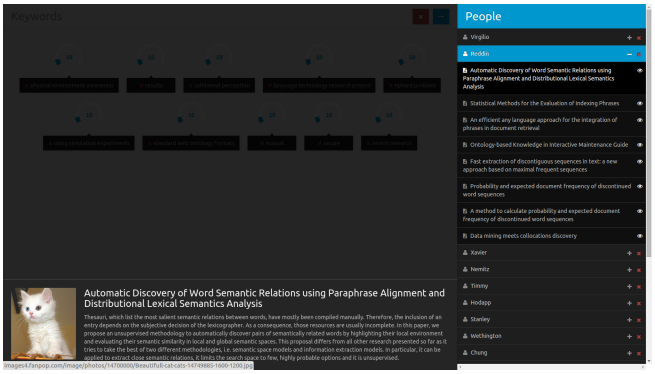

Figure 3. The interactive search interface.

System Design

The data flow from the systems perspective is illustrated in Figure 4. The system can be roughly divided into two parts: the pre-processing step and the online step, which can be further divided into two problems: exploration of the keyword space and ranking of the researchers. Below, we describe these three aspects in more detail.

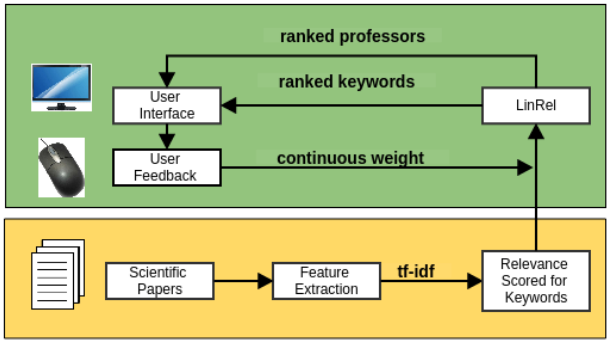

Figure 4. System Diagram.

The Preprocessing Step

In order to obtain keywords representative of each researcher, we collected scientific articles published by each researcher. To ensure that we have an up-to-date publication list of each researcher, we used our university publication repository storing the entire publication record of all the university employees.

We extract keywords from every article using an n-gram-based evaluation metrics for automatic keyphrase extraction [8]. In order to preserve the dependency between the keywords and the documents, initially we represent each keyword k as a binary vector of length n, where n is the number of documents under consideration. The vector indicates the presence or absence of a keyword in a given document. To account for the fact that certain keywords occur in many documents, while others are rare, we further normalize each vector using the inverse document frequency (idf) [11] measure:

$$\text{IDF}(k) = \log \frac{n}{|\{d \in D : k \in d\}|} \quad (1)$$

where $|\{d \in D : k \in d\}|$ indicates the number of documents where a keyword k appears.

Keyword Exploration

In order to help the user to explore the keyword space, we use LinRel [1], an algorithm that has already been proven to work well in other interactive retrieval systems [2, 7].

At each iteration at a time point t a subset of keywords is presented to the user and the user can provide feedback on them. After receiving the user feedback, a new estimate of the keyword weights is computed based on which new keywords are presented to the user and the user response.

Relevance Feedback. The relevance feedback is a vector of relevance scores $r_1 \dots r_t$ that the user assigns to the keywords $k_1 \dots k_t$ displayed to the user up to time t. The relevance

scores range from 0 to 1, where 1 indicates that a given intent was highly relevant to the user.

Intent Model. The users search intent is modeled as an intent vector v_t (of which a subset and corresponding information is visualized for the user) at time t. The vector v_t at time t consists of weights estimated for each keyword present in a data matrix K, where the rows of the matrix are the feature vectors $k_i \dots k_p$ of p keywords and the columns are the n articles in the collection described.

Intuitively, the estimation problem could be solved by obtaining feedback for every intent in K, however when dealing with large datasets it may not be possible in practice for the user to provide relevance scores to thousands or even millions of keywords. Thus, only a small subset of the information in the collection is presented to the user to examine at each time point t. The keywords seen by the user from the beginning of the search session until the present time t are saved in matrix K_t. The data contained in the matrix K_t provides a context for the user's current task and can be used to estimate what type of keywords might be useful for the user to continue the search.

Relevance and Uncertainty. At each iteration of the search, a new set of keywords is suggested to the user based on the feedback from the user obtained in previous iterations. When selecting the next set of keywords to display, the system might simply select the keywords with the highest estimate. But since the estimate v_t may be inaccurate, this exploitative choice might be suboptimal, i.e. the system will only suggest keywords similar to the ones presented to the user in previous iterations and so the user will get stuck in the local filter bubble. Alternatively, the system might exploratively select for presentation a keyword for which the user feedback improves the accuracy of the estimate v_t, enabling better keyword selections in subsequent iterations. In order to allow for a more exploratory selection of keywords for presentation to the user, we employ the LinRel algorithm which allows us to incorporate uncertainty into the system by trading off between exploration and exploitation. Thus, in each iteration t, we obtain an estimate v_t by solving the regularized linear regression problem:

$$A = (K \cdot (K_t^\top \cdot K_t + \lambda I)^{-1} K_t^\top) \tag{2}$$

where A is the solution to the regression problem with I being the identity matrix and $\lambda > 0$ the regularization parameter. The value of λ was set to 0.5, which was obtained through trial and error. The regression solution A is further adjusted by taking into consideration the user's feedback r_t obtained so far:

$$v_t = A \cdot r_t \tag{3}$$

However, if we only select keywords with the highest values of v_t for presentation to the user, then we risk displaying only intents that are similar to the ones shown in previous iterations. Thus, to deal with the exploration–exploitation tradeoff, LinRel selects for presentation not the keywords with largest estimated relevance score v_t, but the keywords with the largest upper confidence bound for the relevance score. The vector s_t with the upper confidence bounds for each keyword at time t is calculated as

$$s_t = A \cdot r_t + \frac{c}{2}\|A\| \tag{4}$$

where $\|A\|$ is the L_2 norm of the linear regression solution. The constant c is used to adjust the confidence level of the upper confidence bound. The optimal value of $c = 2$ was determined through trial and error.

To summarize, at each iteration of the search, based on the user response, the system suggests new keywords to the user. Instead of suggesting only keywords with the highest relevance score, the system suggests keywords with the highest confidence bound. In this way, the user is presented with a combination of keywords with a high relevance score as well as keywords about which the system is uncertain, which prevents the user from getting stuck in a small area of the search space.

Ranking of the Researchers

In our system, the researchers are represented by a combination of keywords obtained from their publications. After selecting a set of keywords to present to the user, the system uses the same set of keywords to rank the researchers. In order to rank the researchers, we apply a probabilistic model of information retrieval. We use a function based on the BM25 term-weighting and document scoring function [12]. We treat the set of keywords presented to the user at each iteration as a query q consisting of keywords $k_1 \dots k_j$ and we score all the researchers using the following scoring function:

$$\sum_{i=1}^{j} \text{IDF}(k_i) \cdot \left[\frac{f(k_i, R) \cdot (m+1)}{f(k_i, R) + m \cdot (1 - b + b \cdot |R|)} \right], \tag{5}$$

where $\text{IDF}(k_i)$ is the inverse document frequency of keyword k_i, R indicates the collection of documents representing a given researcher, $f(k_i, R)$ is the keyword frequency in document collection R and $|R|$ is the overall number of keywords in collection R. The terms m and b are free parameters, usually chosen, in absence of an advanced optimization, as $m = 1.2$ and $b = 0.75$.

CONCLUSION

REFERENCES

1. Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research 3* (2002), 397 – 422.

2. Auer, P., Hussain, Z., Kaski, S., A.Klami, J.Kujala, Laaksonen, J., Leung, A., Pasupa, K., and Shawe-Taylor, J. Pinview: Implicit feedback in content-based image retrieval. In *JMLR Workshop and Conference Proceedings: Workshop on Application of Pattern Analysis*, vol. 11 (2010), 51–57.

3. Basu, C., Cohen, W. W., Hirsh, H., and Nevill-Manning, C. Technical paper recommendation: A study in combining multiple information sources. *arXiv:1106.0248* (2011).

4. Chau, D. H., Kittur, A., Hong, J. I., and Faloutsos, C. Apolo: making sense of large network data by combining rich user interaction and machine learning. In *CHI* (2011), 167–176.

5. Dunne, C., Shneiderman, B., Gove, R., Klavans, J., and Dorr, B. Rapid understanding of scientific paper collections: integrating statistics, text analytics, and visualization. *JASIST: Journal of the American Society for Information Science and Technology* (2012).

6. Ekstrand, M. D., Kannan, P., Stemper, J. A., Butler, J. T., Konstan, J. A., and Riedl, J. T. Automatically building research reading lists. In *Proceedings of the fourth ACM conference on Recommender systems* (2010), 159–166.

7. Głowacka, D., Ruotsalo, T., Konyushkova, K., Athukorala, K., Kaski, S., and Jacucci, G. Directing exploratory search: Reinforcement learning from user interactions with keywords. In *Proc. of IUI* (2013).

8. Kim, S. N., Baldwin, T., and Kan, M.-Y. Evaluating n-gram based evaluation metrics for automatic keyphrase extraction. In *Proceedings of the 23rd international conference on computational linguistics*, Association for Computational Linguistics (2010), 572–580.

9. McNee, S. M., Albert, I., Cosley, D., Gopalkrishnan, P., Lam, S. K., Rashid, A. M., Konstan, J. A., and Riedl, J. On the recommending of citations for research papers. In *ACM conference on Computer supported cooperative work* (2002), 116–125.

10. Parra, D., Brusilovsky, P., and Trattner, C. See what you want to see: Visual user-driven approach for hybrid recommendation. In *Proceedings of the 19th International Conference on Intelligent User Interfaces*, IUI '14, ACM (New York, NY, USA, 2014), 235–240.

11. Spärck-Jones, K. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation 28*, 1 (1972), 11–21.

12. Spärck-Jones, K., Walker, S., and Robertson, S. A probabilistic model of information retrieval: Development and comparative experiments (parts 1 and 2). *Information Processing and Management 36* (2000), 779 – 840.

13. Verbert, K., Parra, D., Brusilovsky, P., and Duval, E. Visualizing recommendations to support exploration, transparency and controllability. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces*, IUI '13, ACM (New York, NY, USA, 2013), 351–362.