
Visualizing User Model in Exploratory Search Tasks

Kalle Ilves
University of Helsinki

Alan Medlar
University of Helsinki

Dorota Głowacka
University of Helsinki

Abstract

We present our ongoing work on an interactive exploratory information retrieval system designed to explain to the user the reasons for its predictions and help the user to build a mental model to predict how the system will behave. Users indicate which documents interest them and reinforcement learning is used to model the user by allowing the system to trade off between exploration and exploitation.

In most interactive systems, the user has a concept in mind that the system is trying to learn while the user interacts with the system, e.g. a particular genre of music or specific types of documents. Many of such systems rely on machine learning techniques to help the system to identify the concept that the user has in mind. In recent years, there has been a growing interest in applying various reinforcement learning (RL) techniques in exploratory information retrieval systems. RL allows the system to trade between exploitation (moving towards more specific subtopics) and exploration (going towards alternative topics), and has been shown to be helpful in information retrieval, in particular in exploratory search systems, where the final search goal maybe vague [2]. However, experimental results show that users of such information retrieval often feel that they are not fully in control of the system and sometimes they do not fully understand why a given set of results was returned to them by the information retrieval system [2]. In this paper, we address the issue of how to assist the user in understanding how the exploration – exploitation trade-off in a RL based information retrieval system affects their search results. We describe a prototype document retrieval system based on the exploration – exploitation trade-off to support exploratory search.

Appearing in Proceedings of the 4th Workshop on Machine Learning for Interactive Systems (MLIS) 2015, Lille, France. JMLR: W&CP volume 43. Copyright 2015 by the authors.

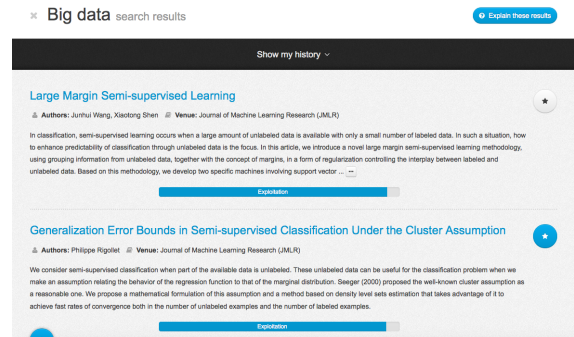


Figure 1: System interface with the query "big data".

Our ongoing evaluation of the system indicates that the additional visualization allows the user to better understand the workings of the algorithm used in the system and obtain better search results.

1 System Overview

The search starts with the user typing in a query, which results in a list of documents being displayed (Figure 1). The user can indicate which documents they are interested in by clicking on the ball next to it. The inbuilt RL techniques allow the system to form a model of user’s interests and suggest appropriate documents in the next search iteration. After clicking the “next” button at the bottom of the page, a new set of documents is displayed. An additional visualisation panel allows the user to understand how the displayed documents are related to one another and which words from each document were responsible for the document ranking. The system is particularly suited for exploratory search, where the user would like to learn more about a given topic, e.g. clustering, but may lack the knowledge to type in more specific queries about the topic, such as spectral clustering or k-means clustering. The current version of the system is based on over 1 million documents from the ArXiv repository.

The main idea behind the interactive interface is that instead of typing queries at each iteration, the user navigates through the contents by indicating which

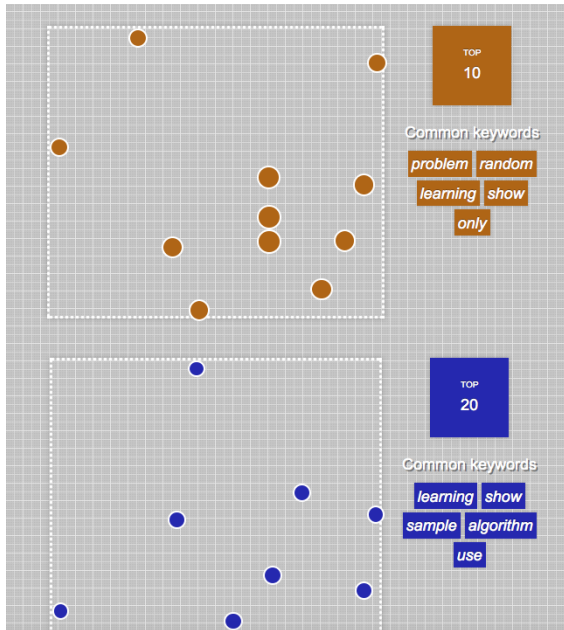


Figure 2: Visualizatiion of the top 20 documents.

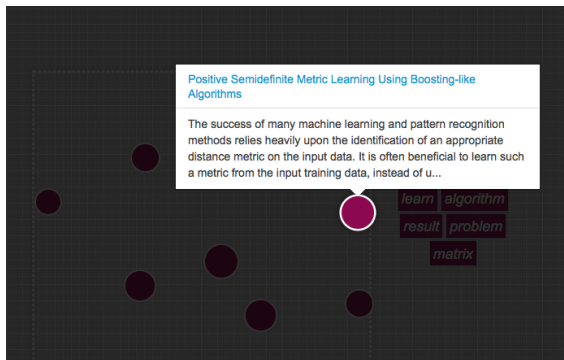


Figure 3: Visualizing title and abstract of document.

documents he is interested in. When the user clicks on the blue “Explain these results” button in the top right hand-side corner, a new tab opens (Figure 2). The balls represent the top 20 documents in sets of 10. Each set of 10 documents shows how dispersed or similar to each other these 10 documents are. The size of each ball indicates how “similar” a given document is to the original query. On the right side of each set of 10 documents a list of most common words in this set is presented. The user can keep on scrolling down to see more document sets. When the user hovers the mouse over one of the balls, the title and the abstract of the document it represents appear (Figure 3).

2 Retrieving and Ranking Documents

After typing in the first query, the initial set of documents is ranked based on the Okapi BM25 algorithm

[3]. Next, the user indicates which documents interest them to obtain more relevant documents. The user feedback is given by a relevance score $r \in [0, 1]$, where 1 indicates that the document is of high relevance. Documents that do not receive an explicit relevance feedback are assumed to receive relevance score of 0. In order to help the user to explore the document space, we use LinRel [1], an algorithm that has already been proven to work well in other interactive retrieval systems [2]. The algorithm maintains a representation of the estimate \hat{w} of the unknown weight vector w which maps document features to relevance scores. In each iteration, LinRel obtains an estimate \hat{w} by solving a linear regression problem. Suppose we have a matrix K , where each row k_i is a feature vector of documents presented so far. Let $r = (r_1, r_2 \dots r_p)^\top$ be the column vector of relevance scores received so far from the user, where p is a number of iterations. Thus, LinRel tries to estimate \hat{w} by solving the linear regression problem $r = K \cdot w$. Based on \hat{w} , LinRel calculates an estimated relevance score $\hat{r}_i = k_i \cdot \hat{w}$ for each document k_i . In order to deal with the exploration-exploitation trade-off, we present documents not with the highest score, but with the largest upper confidence bound for the relevance score. Thus, if σ_i is an upper bound on standard deviation of relevance estimate \hat{r}_i , the upper confidence bound of keyword i is calculated as $r_i + \gamma \sigma_i$, where $\gamma > 0$ is a constant used to adjust the confidence level of the upper confidence bound. In each iteration, LinRel calculates: $s_i = k_i \cdot (K^\top \cdot K + \lambda I)^{-1} K^\top \cdot r$, where λ is the regularization parameter. The documents that maximize $s_i \cdot r + \frac{\gamma}{2} \|s_i\|$ are selected for presentation. The first term $s_i \cdot r$ effectively ranks all the documents in the dataset based on user feedback so far. The second term $\frac{\gamma}{2} \|s_i\|$ is an exploration term which ensures that the user is presented with a more diverse set of results.

Acknowledgements: The project was supported by The Finnish Funding Agency for Innovation (under projects Re:Know and D2I) and by the Academy of Finland (under the COIN project).

References

- [1] P. Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3:397 – 422, 2002.
- [2] D. Głowacka, T. Ruotsalo, K. Konyushkova, K. Athukorala, S. Kaski, and G. Jacucci. Directing exploratory search: Reinforcement learning from user interactions with keywords. In *Proc. of IUI*, 2013.
- [3] K. Sparck Jones, S. Walker, and S. E. Robertson. A probabilistic model of information retrieval: Development and comparative experiments. *Information Processing and Management*, 36(6):779–840, 2000.