ConSTR: A Contextual Search Term Recommender

Thomas Krämer∗, Zeljko Carevic†, Dwaipayan Roy†, Claus-Peter Klas∗, Philipp Mayr∗
∗GESIS – Leibniz Institute for the Social Sciences, Cologne, Germany, †IISER, Kolkata, India
Email: *{thomas.kraemer, zeljko.carevic, philipp.mayr, claus-peter.klas}@gesis.org, †dwaipayan.ro...
in arXiv documents. Hence, we decided to enrich the arXiv index with keywords that we extracted using YAKE [8] - an unsupervised automatic keyword extraction method. We used the abstract of the arXiv documents to extract a maximum of 5 keywords per document with an n-gram size of two.

C. Search application

The search application is implemented using JHipster and Searchkit.

1) Interaction context: The interaction context consists of queries and keywords. Users can assess past search interactions and remove items from the interaction context with one click. The following interactions are stored in a user’s interaction context: 1) each issued query, 2) click on result (keywords of the document) and 3) click on a recommendation (query terms expanded by the recommendations).

2) Model settings: Users can choose the model (arXiv or Wikipedia+Gigaword5) used to determine the recommended terms. Users can also specify the similarity threshold (i.e. the minimum similarity score between a seed and a recommended term) to be applied when requesting for the nearest terms to the query, and the number of recommendation.

3) Recommendation pane: Recommendations are updated when a user issues a search and retrieved for both the current query and the interaction context. They are displayed on the upper (query based) or lower part (interaction context based) of the recommendation pane (see Figure 1). A click on a recommendation will expand the current query and results update automatically.

IV. FUTURE WORK

Currently, recommendations are retrieved for each term of a query. We intend to enable support for recommendations based on phrase queries as well as post-processing (e.g. stemming) before presenting recommendations. The interactions persisted in the interaction context will be extended and include other user actions, such as mouse position or the selection of search facets. Items in the interaction context are ranked and have a timestamp. As the interaction context grows over time, we will implement algorithms taking into account the interaction history and use only selected queries and keywords instead of the entire interaction context. As part of future work, ConSTR will be evaluated with real users.

V. DEMO ACCESS

The demo is available at constr.memotaxis.org. Further information can be found in the ConSTR demo video.

REFERENCES

[1] G. Marchionini, “Exploratory search: from finding to understanding,” Communications of the ACM, vol. 49, no. 4, pp. 41–46, 2006.
[2] C. C. Kuhlthau, “A principle of uncertainty for information seeking,” Journal of documentation, 1993.
[3] R. W. White and R. A. Roth, “Exploratory search: Beyond the query-response paradigm,” Synthesis lectures on information concepts, retrieval, and services, vol. 1, no. 1, pp. 1–98, 2009.
[4] P. Mutschke, P. Mayr, P. Schaer, and Y. Sure, “Science models as value-added services for scholarly information systems,” Scientometrics, vol. 89, no. 1, pp. 349–364, 2011.
[5] N. Fuhr, C.-P. Klas, A. Schaefer, and P. Mutschke, “Daffodil: An integrated desktop for supporting high-level search activities in federated digital libraries,” in Research and Advanced Technology for Digital Libraries, M. Agosti and C. Thanos, Eds., 2002, pp. 597–612.
[6] Z. Carevic, S. Schüller, P. Mayr, and N. Fuhr, “Contextualised browsing in a digital library’s living lab,” in Proceedings of the 18th ACM/IEEE Joint Conference on Digital Libraries, 2018, p. 89–98.
[7] D. Roy, D. Ganguly, S. Bhatia, S. Bedathur, and M. Mitra, “Using word embeddings for information retrieval: How collection and term normalization choices affect performance,” in Proc. of 27th ACM CIKM 2018, Italy, October 22-26, 2018. ACM, 2018, pp. 1835–1838.
[8] R. Campos, V. Mangaravite, A. Pasquali, A. M. Jorge, C. Nunes, and A. Jatowt, “Yake! collection-independent automatic keyword extractor,” in ECIR. Springer, 2018, pp. 806–810.