InfoLand: Information Lay-of-Land for Session Search

Jiyun Luo, Dongyi Guan, Hui Yang
Department of Computer Science
Georgetown University
{jl1749,dg372}@georgetown.edu, huiyang@cs.georgetown.edu

ABSTRACT

Search result clustering (SRC) is a post-retrieval process that hierarchically organizes search results. The hierarchical structure offers overview for the search results and displays an "information lay-of-land" that intents to guide the users throughout a search session. However, SRC hierarchies are sensitive to query changes, which are common among queries in the same session. This instability may leave users seemly random overviews throughout the session. We present a new tool called InfoLand that integrates external knowledge from Wikipedia when building SRC hierarchies and increase their stability. Evaluation on TREC 2010-2011 Session Tracks shows that InfoLand produces more stable results organization than a commercial search engine.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Storage and Retrieval—*Information Search and Retrieval*

Keywords

Search Results Clustering; Session Search

1. INTRODUCTION

Search result clustering (SRC) [1, 4] is a post-retrieval process that hierarchically organizes search results. It is used in Meta search engines such as Yippy.com (previously known as Clusty). SRC hierarchies display an information "lay of land" for search and help users to quickly locate relevant documents from piles of search results.

Session search has recently attracted more attentions in Information Retrieval (IR) research. A session usually contains multiple queries. These queries are usually highly related to a main topic and to each other. Ideal SRC hierarchies generated for queries in the same session should be highly related too. However, the state-of-the-art SRC hierarchies are usually sensitive to query changes and hence demonstrate unstable hierarchies throughout the session. Figure 1 shows hierarchies generated by Yippy for queries 'diet' and 'low carb diet' (TREC 2010 session 123). Although many sessions only show slightly changes among queries[1], the hierarchies built for these queries' search results can be dramatically different from each other.

The reason that causes unstable hierarchies lies in the fact that that many hierarchy construction approaches are data-driven [1, 4]. A common approach, which is also used in Yippy, is to first group similar documents into clusters and then label the clusters. We observe that clustering-based ap-

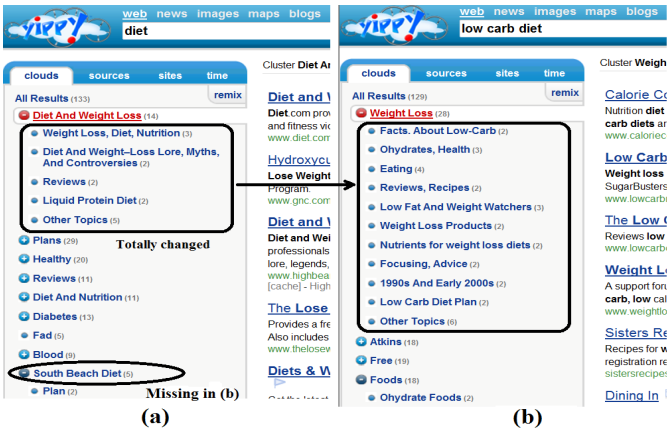

Figure 1: SRC hierarchies generated by Yippy.

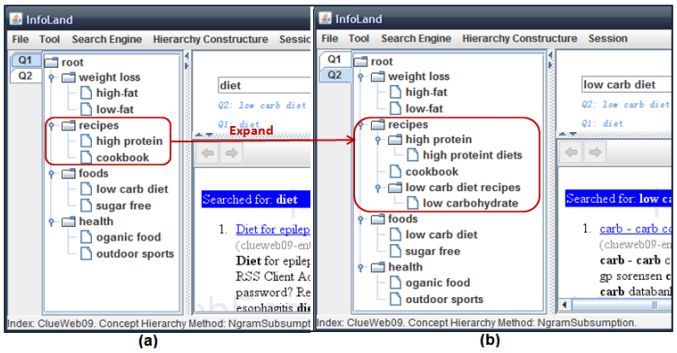

Figure 2: SRC hierarchies generated by InfoLand.

[1]For instance, about 38.6% adjacent queries in TREC 2010 and 2011 Session tracks [2] only show one word difference and 26.4% show two word difference.

SIGIR'13, July 28–August 1, 2013, Dublin, Ireland.
ACM 978-1-4503-2034-4/13/07.

proaches often produce mixed-initiative clusters and reduce hierarchy stability.

We propose a novel hierarchy construction tool, InfoLand, which injects world knowledge to an existing hierarchy to increase its stability. Evaluation over TREC 2010 and 2011 Session tracks shows that InfoLand produces significantly more stable stable SRC hierarchies than Yippy.

2. BUILD STABLE CONCEPT HIERARCHIES

We propose three major steps in building stable concept hierarchies: concept extraction, mapping to Wiki entries, and hierarchy construction. First, a single query q and its search results D are processed and a set of concepts C that best represents D are extracted by algorithms described in [5]. Concepts from the hierarchy built by Yippy are also included in C.

Next, for a concept $c \in C$, InfoLand maps it to its most relevant Wiki entry e, which is called a *reference Wiki entry*. We built a Lemur[2] index over the entire Wikipedia collection in ClueWeb09.[3] A concept c is sent as a query to the index and the top 10 returned Wiki pages are examined. The titles of these pages are considered as candidate Wiki entries for c and are denoted as $\{e_i\}, i = 1 \cdots 10$. Due to ambiguity in natural language, the top returned results may not be related to the current search session. We hence disambiguate Wiki entries by measuring the similarity between the entires and the topics mentioned in the search queries. The similarity is measured by by *mutual information* between an entry candidate e_i and all concepts C for query q:

$$\mathrm{MI}(e_i, C) = \sum_{c \in \mathcal{C}} \mathrm{PMI}(e_i, c|\mathrm{E}) \times \log(1 + ctf(c)) \cdot idf(c) \qquad (1)$$

where $\log(1+ctf(c)) \cdot idf(c)$ measures the importance of concept c in representing the main topic in D. Point-wise Mutual Information (PMI) measures the similarity between e_i and c w.r.t. a corpus E: $\mathrm{PMI}(e_i, c|\mathrm{E}) = \log \frac{\mathrm{df}(e_i, c; \mathrm{E}) \times |\mathrm{E}|}{\mathrm{df}(e_i; \mathrm{E}) \times \mathrm{df}(c; \mathrm{E})}$, where $\mathrm{df}(x; \mathrm{E})$ is the document frequency of term x in corpus E and $|\mathrm{E}|$ is the collection size.

The most relevant Wiki entry to the query is selected as the *reference Wiki entry*. We obtain reference Wiki entries e_x and e_y for concepts x and y and decide whether x subsumes y based on the following cases:

(a) e_x is a Wiki category: From e_y's Wiki page, we extract the Wiki categories that e_y belongs to. We call the list of Wiki categories for e_y *super categories* and denote them as $\mathcal{S}_y$. x subsumes y if $e_x \in \mathcal{S}_y$.

(b) Only e_y is a Wiki category: x does not subsumes y.

(c) Neither e_x nor e_y is a Wiki category: We form super category sets for both $\mathcal{S}_y$ and $\mathcal{S}_x$. For each $s_{yi} \in \mathcal{S}_y$, we extract its super categories and form a super-supercategory set $\mathcal{SS}_y$ for e_y. We then measure the normalized overlap between $\mathcal{SS}_y$ and $\mathcal{S}_x$: $\mathrm{Score}_{\mathrm{sub}}(x, y) = \frac{\mathrm{count}(s; s \in \mathcal{S}_x \text{ and } s \in \mathcal{SS}_y)}{\min(|\mathcal{S}_x|, |\mathcal{SS}_y|)}$, where $\mathrm{count}(s; s \in \mathcal{S}_x \text{ and } s \in \mathcal{SS}_y)$ denotes the number of categories that appear in both $\mathcal{S}_x$ and $\mathcal{SS}_y$. If $\mathrm{Score}_{\mathrm{sub}}(x, y)$ for a potential parent-child pair (x, y) is above 0.6, we consider x subsumes y.

Lastly, based on the subsumption relationship identified, we form SRC hierarchies as in [3].

[2]http://www.lemurproject.org.

[3]http://www.lemurproject.org/clueweb09.php/.

Table 1: Stability of SRC Hierarchies for TREC queries. ‡ indicates a significant improvement at $p < 0.005$.

2010	FBS	Node overlap	Parent-child precision
Yippy	0.463	0.415	0.144
InfoLand	0.603‡	0.529‡	0.450‡
2011	FBS	Node overlap	Parent-child precision
Yippy	0.440	0.327	0.115
InfoLand	0.504‡	0.420‡	0.247‡

3. EVALUATION

Data from TREC 2010 and 2011 Session tracks is used in the evaluation. For every query q, we retrieve the top 1000 documents from an index built over the ClueWeb09 CatB as its search results D. All TREC official ground truth documents are also merged into the results set. In total, our dataset contains 299,000 documents, 124 sessions, and 299 queries (on average 2.41 queries per sessions).

Given a session $\mathcal{S}$ with queries q_1, q_2, ... q_n, we measure the stability of SRC by averaging the hierarchy similarity among query pairs in $\mathcal{S}$. It is defined as: $Stability(\mathcal{S}) = \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \mathrm{Sim}_{hie}(H_i, H_j)$, where n is the number of queries in $\mathcal{S}$, H_i and H_j are hierarchies built for q_i and q_j, and $\mathrm{Sim}_{hie}(H_i, H_j)$ is the hierarchy similarity. Methods to calculate Sim_{hie} include fragment-based similarity (FBS), node overlap, and parent-child precision [5].

Table 1 compares the stability evaluation for hierarchies generated by InfoLand and by Yippy over the TREC 2010 and 2011 datasets. InfoLand significantly outperforms Yippy in stability in all metrics for both datasets.

Figure 2 shows the SRC hierarchies build by InfoLand for TREC 2010 session 123. Comparing to Figure 1, we observe a local expansion of concepts from the left hierarchy to the right. It coincides well with the fact that this session contains a specification from 'diet' to 'low carb diet'. Other parts of the two hierarchies remain almost the same; which demonstrates high hierarchy stability.

4. CONCLUSIONS

Search results hierarchies built for queries in the same session are usually sensitive to query changes. This partly diminishes the benefits that search result organization intents to offer. We present a new tool called infoLand that incorporates external knowledge to improve the stability of SRC hierarchies and enable them to better serve as information lay-of-land to guide session search. Evaluation over TREC 2010 and 2011 Session tracks demonstrates that InfoLand produces more stable hierarchies than Yippy.

5. ACKNOWLEDGMENTS

This research was supported by NSF grant CNS-1223825.

6. REFERENCES

[1] D. C. Anastasiu, B. J. Gao, and D. Buttler. A framework for personalized and collaborative clustering of search results. In *CIKM '11*.

[2] E. Kanoulas, B. Carterette, M. Hall, P. Clough, and M. Sanderson. Overview of the trec 2011 session track.

[3] M. Sanderson and B. Croft. Deriving concept hierarchies from text. In *SIGIR '99*.

[4] U. Scaiella, P. Ferragina, A. Marino, and M. Ciaramita. Topical clustering of search results. In *WSDM '12*.

[5] H. Yang. *Personalized Concept Hierarchy Construction*. PhD thesis, 2011.