Indexing and Querying Overlapping Structures

Faegheh Hasibi
Norwegian University of Science and Technology
Trondheim, Norway
faeghehh@idi.ntnu.no

ABSTRACT

Structural information retrieval is mostly based on hierarchy. However, in real life information is not purely hierarchical and structural elements may overlap each other. The most common example is a document with two distinct structural views, where the logical view is `section/ subsection/ paragraph` and the physical view is `page/ line`. Each single structural view of this document is a hierarchy and the components are either disjoint or nested inside each other. The overlapping issue arises when one structural element cannot be neatly nested into others. For instance, when a paragraph starts in one page and terminates in the next page. Similar situations can appear in videos and other multimedia contents, where temporal or spatial constituents of a media file may overlap each other.

Querying over overlapping structures is one of the challenges of large scale search engines. For instance, FSIS (FAST Search for Internet Sites) [1] is a Microsoft search platform, which encounters overlaps while analysing content of textual data. FSIS uses a pipeline process to extract structure and semantic information of documents. The pipeline contains several components, where each component writes annotations to the input data. These annotations consist of structural elements and some of them may overlap each other. Handling overlapping structures in search engines will add a novel capability of searching, where users can ask queries such as "Find all the words that overlap two lines" or "Find the music played during Intro scene of Avatar movie". There are also other use cases, where the user of the search engine is not a person, but is a specific program with complex, non-traditional information retrieval needs.

This research attempts to index overlapping structures and provide efficient query processing for large-scale search engines. The current research on overlapping structures revolves around encoding and modelling data, while indexing and query processing methods need investigations. Moreover, due to intrinsic complexity of overlaps, XML indexing and query processing techniques cannot be used for overlapping structures. Hence, my research on overlapping structures comprises three main parts: (1) an indexing method that supports both hierarchies and overlaps; (2) a query processing method based on the indexing technique and (3) a query language that is close to natural language and supports both full text and structural queries.

Our approach for indexing overlaps is to adapt the PrePost [3] XML indexing method to overlapping structures. This method labels each node with its start and end positions and requires modest storage space. However, PrePost indexing cannot be used for overlapping nodes. To overcome this issue, we need to define a data model for overlapping structures. Since hierarchies are not sufficient to describe overlapping components, several data structures have been introduced by scholars. One of the most interesting data models is GODDAG [2]. GODDAG is a tree-like graph, where nodes can have multiple parentage. This model can support overlaps as well as simple inheritance. Our proposed data model for indexing overlaps is such a tree-like structure, where we can define overlapping, parent-child and ancestor-descendant relationships.

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

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

Keywords

Overlapping Structures, Indexing, Query Processing, Semi-structured Data

1. REFERENCES

[1] M. Bennett, J. Fried, M. Kehoe, and N. Voskresenskaya. *Professional Microsoft Search: FAST Search, SharePoint Search, and Search Server.* Wrox, 2010.

[2] C. Sperberg-McQueen and C. Huitfeldt. Goddag: A data structure for overlapping hierarchies. In P. King and E. Munson, editors, *Digital Documents: Systems and Principles*, volume 2023 of *Lecture Notes in Computer Science*, pages 606–630. Springer Berlin-Heidelberg, 2004.

[3] C. Zhang, J. Naughton, D. DeWitt, Q. Luo, and G. Lohman. On supporting containment queries in relational database management systems. In *Proceedings of the 2001 ACM SIGMOD international conference on Management of data*, SIGMOD '01, pages 425–436, New York, NY, USA, 2001. ACM.