

POLIS: A Summarisation Logic for Structured Documents

Jan Frederik Forst

2nd year PhD

Information Retrieval

Thomas Rölleke and Tassos Tombros

The “internet revolution” significantly changed user’s behaviour seeking for information: it used to be the case that computer users could access only the information stored on their local computer (or potentially the information stored on other computers on a local network), and were thus able to maintain and organise data manually. With the vast amounts of data available on the internet, this is not possible anymore. Instead, locally organising data has been replaced by looking for relevant information using search engines, such as Google.

While search engines carry out fairly complex operations to find web pages that fulfil a searcher’s information needs, the actual implementation and the model the search engine uses to match a query against the collection of stored documents is entirely hidden from users, and is reduced to a text box awaiting searchers’ queries, or (for “advanced” searches) a textbox and a number of checkboxes. However, between the actual implementation of a search engine, and the interface that allows users to access it, exists another layer of abstraction, in which the model the search engine uses to retrieve documents is explained as a set of logical operations. This layer is known as the “logic” approach to information retrieval.

This “logic layer” cannot only be used to explain the operations of a search engine as a set of statements in some form of logic, it also allows the *definition* of search engine models (or *strategies*) as a series of logical statements, without consideration for the actual implementation of that model. While this logical layer of IR has been explored fairly extensively for the retrieval of webpages and structured and unstructured documents, other important information retrieval operations have not been described in terms of such a logical layer.

This is where my research starts. One of the less prominent information retrieval task, which has largely been overshadowed by the buzz created by web search engines, is that of document (or, more generally, *information*) summarisation. An informal description of what constitutes a “good” document summary is fairly simple: it should provide – in a cohesive and grammatically consistent way – the most informative content of a document. Over the past forty to fifty years, a number of approaches at summarising documents and other information have been proposed, and have been successfully applied to test collections.

However, these approaches were all very hard coded, and described at a very implementational level.

The aim of my research is to provide a logic that allows (expert) users to implement their own summarisation strategies, without having to worry about how these strategies are implemented. Instead, users enter a logical expression that tells the system how much documents should be compressed, what *parts* of documents should be considered for summarisation etc. This *summarisation logic* would be able to describe summarisation at the same level of abstraction that other logical IR layers use to describe search strategies of search engines.

Such a *summarisation logic* is characterised by two main properties: its syntax, and some semantics. While the syntax only provides a means for checking the correctness of expressions of the logic, the really important work is encompassed by the semantics, which details *how* summaries of documents will be generated. While users will not need to access the semantics of the logic to express their summarisation needs, it also provides a means for experts to compare and modify the behaviour of the logic.

So far, I defined an initial version of the semantics, and several (progressively more refined) versions of a syntax. This early version of a logic is currently being tested on a collection of structured documents, to evaluate the performance of the logic to other summarisation systems. Next steps will include a refinement of the summarisation model used, expressed as a semantics, and the implementation of other summarisation approaches via the abstraction layer provided by the logic.