Related Documents Search Using User Created Annotations

Jakub Ševcech*
Institute of Informatics and Software Engineering
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Ilkovičova, 842 16 Bratislava, Slovakia
sevo.jakub@gmail.com

Abstract

We often use various services for creating bookmarks, tags, highlights and other types of annotations while surfing the Internet or when reading electronic documents as well. These services allows us to create a number of types of annotation that we are commonly creating into printed documents. Annotations attached to electronic documents however can be used for other purposes such as navigation support, text summarization etc. We proposed a method for searching related documents to currently studied document using annotations created by the document reader as indicators of user's interest in particular parts of the document. The method is based on spreading activation in text transformed into graph. For evaluation we created a service called Annota, which allows users to insert various types of annotations into web pages and PDF documents displayed in the web browser. We analyzed properties of various types of annotations inserted by users of Annota into documents. Based on these we evaluated our method by simulation and we compared it against commonly used TF-IDF based method.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering;
I.7.5 [**Document and Text Processing**]: Document analysis

*Master degree study programme in field Software Engineering. Supervisor: Prof. Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies, STU in Bratislava. Work described in this paper was presented at the 9th Student Research Conference in Informatics and Information Technologies IIT.SRC 2013.

Ševcech, J. Related Documents Search Using User Created Annotations. Information Sciences and Technologies Bulletin of the ACM Slovakia, Special Section on Student Research in Informatics and Information Technologies, Vol. 5, No. 2 (2013) 44-47

Keywords

search, annotation, query by document

1. Introduction

Multiple services provides us functions for creating bookmarks, tags, highlights and other types of annotations while surfing the Internet or when reading electronic documents as well. We use these annotations as means to store our thoughts or to organise personal collections of documents using methods such as tag-cloud. There is active research in the field of utilization of annotation for example in support of navigation between documents [1, 12] . Great many applications use annotations as means of navigation between documents and for organizing content [4, 8] . User generated tags is one of the most commonly used methods for organizing content. Tags are used for organizing bookmarks in services such as Diigo[1], but they are also used to organize notes[2], in various blogs and many other applications. User created annotations can be used not only to support navigation, but there are many other possible applications. Annotations can play a great role for example in education systems such as Alef [8] where they are used to improve content quality by means of content error reports, to enrich course content using user generated comments and questions, to organize content using tags and even content summarization using highlights created by document readers [5].

One of possible application for annotations is in document search. There are two possible approaches for exploitation of annotations in search. One of them is to use annotations while indexing documents by expanding documents in a similar way anchor texts are used [12] or by ranking document quality using annotations as its quality indicators [10].

The second possible application of annotation in document search is in query construction. In an experiment performed by Golovchinsky et al. [2], they let users to create annotations into documents using a tablet. They used these annotations as queries in related document search and they compared search precision of these queries with relevance feedback expanded queries. They found, that queries derived from users annotations produced significantly better results than relevance feedback queries. Whereas query expansion requires users to create initial

[1]Diigo, http://www.diigo.com/
[2]Evernote, https://www.evernote.com/

query, query composition using annotations does not require additional activity of document readers, instead it reuses annotations created with another purposes such as better understanding of source document.

In the experiment presented in [2], they used only user generated annotations as source for query construction for document search. More often in search for related documents the content of source document is used to create queries. One of such approaches is presented in [11], where authors were extracting most important phrases from source document. This task is similar to document summarization, however they did not used these phrases to create summarization, instead they were using them as queries to retrieve related documents.

Similar approach for related document retrieval is presented in [7], where they are using similar document retrieval for plagiarism detection. The presented method consists of the most representative sentences extraction step and document retrieval using commonly used search engine and representative sentences from source document as queries for retrieval of candidate documents.

Described solutions use only document content in related document retrieval process. We believe that annotations attached to document can improve related document retrieval precision. In our work we combine the content of the document and user generated annotations to create queries to retrieve related documents. The proposed method uses annotations as interest indicators to determine parts of the document user is most interested in and it produces query in form of list of keywords.

2. Query construction

There exist multiple approaches for related document retrieval, where query for document search is whole document. Many of these approaches are using common search engines for related document retrieval [7, 11]. These search engines are commonly accepting queries only in form of the list of keywords. It is then necessary to extract queries from source document when searching for related documents. One of possible approaches to extract words to the keyword query from source documents is to use one of ATR algorithms or term frequency based metrics such as TF-IDF to extract most important words and use them as query. One of possible approaches to extract words to the keyword query is to use one of ATR algorithms such as TF-IDF to extract most important words and use them as query. This is the method used in commonly used search engines such as ElasticSearch[3] or Apache Solr[4]. They provide special type of query interface called "more like this" query, which processes source text and returns list of similar documents. Internally, the search engine extracts the most important words using TF-IDF algorithm from source text and it uses these most important words as a query to search for related documents. This method for query construction provides rather straightforward possibility to incorporate user created annotations: we can extend source text by content of created annotations with various weights for different types of annotations.

However, the TF-IDF algorithm takes into account only the number of occurrences of words in the source document and in the document collection. We believe that not only the number of word occurrences but also the structure of the source text is very important. Especially if we suppose that while reading the document, users are most interested in only a portion of the document, the portion where they attach annotations. We proposed a method based on spreading activation in text of studied document. This method uses annotations as interest indicators to extract parts of documents user is most interested in. Since spreading activation algorithm is a graph algorithm, the text is firstly preprocessed and transformed into graph. The proposed method is thus composed of two steps: text to graph transformation and query word extraction using spreading activation.

2.1 Text to graph transformation

The text to graph transformation is based on word neighbourhood in the text. The graph created from text using words neighbourhood conserves words importance in node degree but it also reflects the structure of the source text in the structure of edges [6]. Using various graph algorithms such as community detection, various node and edge weightings or spreading activation we can extract properties such as most important terms, topics etc. We use this graph to extract words that can form queries to retrieve similar documents using spreading activation algorithm. To transform text to graph, it is firstly preprocessed in several steps: segmentation, tokenization, stop-words removal and stemming. Then words from the text are transformed to nodes of the graph. The edges of the graph are created between two nodes if corresponding words in the text are in defined maximal distance. The algorithm for text to graph transformation is as follows:

```
1: words = text.removeStopwords.stem.split
2: nodes = words.uniq
3: edges = []
4: for i=0 to words.length do
5:    for j=i to min(i+maxDistance, words.length-1) do
6:       edges.add(words[i],words[j])
7:    end for
8: end for
9: Graph.new(nodes, edges)
```

2.2 Query word extraction

In the text transformed to graph we can use spreading activation algorithm to find the most important nodes/words. Using this algorithm the initial activation is propagating through the graph and we observe where this activation is concentrating. In our case the activation is inserted into the graph through annotations attached to document by its reader. We can divide attached annotations into two classes:

- that highlight parts of documents and
- that attach additional content into documents.

The proposed method takes into account both types. Those, which highlight parts of the document, contribute by activation to nodes representing words of highlighted part of the document. Annotations which enrich content of the document are extending the document graph by adding new nodes and edges and inserting activation to this extended part of the graph.

[3]ElasticSearch, http://www.elasticsearch.org/
[4]Apache Solr, http://lucene.apache.org/solr/

When initial activation is spreading through created graph, the nodes where activation is concentrating are considered words fit into the query. This algorithm is able to extract words, which are important for annotated part of the document, but it is also able to extract globally important words, that are important for document as a whole. The portion of locally and globally important words can be controlled by number of iteration of the algorithm. With increasing number of iteration the activation is spreading from activated part of the document to globally important words. When using this method it is important to determine the best number of iterations and the right amount of activation for various types of annotations to insert into the graph.

3. Evaluation

We have developed a service called Annota[5] [9], which allows users to attach annotations to web pages and to PDF documents displayed in a web browser. The user can create various types of annotations such as: tags, highlights, comments attached to text selections and notes attached to the document as a whole. The service is focused on supporting visitors of digital libraries in annotation of documents while reading them.

We analyzed behaviour of 82 users that created 1416 bookmarks and 399 annotations. We derived probabilistic distributions of annotation attributes such as note length, number of highlights per user and per document or probability of comment to be attached to text selection. The diagram on figure 1 represents an example of derived distribution of number of highlighted texts per document that follows logarithmic distribution. All observed parameters were following logarithmic or geometric distributions. Using these distributions and their parameters we created a simulation, to find optimal weights for various types of annotations and number of iterations of proposed method, where we optimized query construction for document search precision.

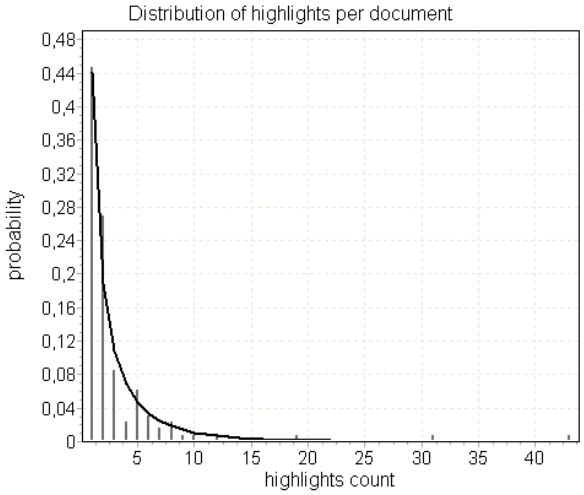

Figure 1: Logarithmic distribution of highlighted texts number per document.

We performed the simulation on dataset we created by extracting documents from Wikipedia. We created source documents with aim to create documents containing several similar sections and with different topics. These generated documents simulate documents, where the user is interested in only one fraction. To create such documents we used disambiguation pages on Wikipedia. The disambiguation page resolves multiple meanings of the same word and contains links to pages describing each of these meanings. We downloaded all disambiguation pages and we selected random subset of these pages for which we downloaded pages they are linking to. Along with these documents we downloaded all documents, which had common category with at least one of these documents. We used search engine ElasticSearch to create the index of all downloaded documents and to search within this index.

In the simulation we generated annotations in a way to correspond with probabilistic distributions extracted from annotations created by users of the service Annota. From every disambiguation page and pages it was linking to, we created one source document by combining abstracts of all pages in random order. For every source document we selected one of abstracts composing the source document. This abstract simulates one topic user is most interested in. Into this abstract we generated various types of annotations, both annotations highlighting parts of the document and annotations inserting additional content. Annotations highlighting parts of the document were randomly distributed along the whole abstract. To simulate content of annotations extending content of annotated document (note, comments) we used parts of the page annotated abstract was extracted from.

Generated annotations along with source document content were used to create query for related documents search in the index of all downloaded documents. We considered relevant document the document that was from the same category as the page of annotated abstract. We compared search precision for proposed method and for TF-IDF based method ("more like this" query) provided by ElasticSearch when searching for 10 most relevant documents. We performed this simulation with several combinations of parameters and we implemented hill climbing algorithm to optimize parameter combination for the highest precision for both compared methods. Single iteration of performed simulation is described by following pseudocode:

```
1: for all disambiguation in disambiguationPages do
2:     pages = disambiguation.pages
3:     abstracts = extractAbstracts(pages)
4:     for all abstract in abstracts do
5:         text = abstracts.shuffle.join(" ")
6:         graph = Graph.new(text)
7:         annotations = Annotation.generate(abstract)
8:         graph.activate(annotations, annotationWeights)
9:         graph.spreadActivation
10:        query = graph.topNodes
11:        results = ElasticSearch.search(query)
12:        rel = results.withCategory(abstract.page.cat)
13:    end for
14: end for
```

Along with simulation for parameter optimization and method comparison, we performed two experiments to determine retrieval precision with no annotations and when whole abstract of the source document was highlighted. These experiments aimed to determine precision of com-

[5]Annota, http://annota.fiit.stuba.sk/

pared methods when no annotations are available and when we have complete information about user's interest. Obtained results for methods with generated annotations along with simulation with no annotations and with whole document fragment annotated are summarized in table 1.

Table 1: Simulation results for proposed method and TF-IDF based method.

Method	Precision
TF-IDF based, no annotations	21,32%
Proposed, no annotations	21,96%
TF-IDF based, generated annotations	33,64%
Proposed, generated annotations	37,07%
TF-IDF based, whole fragment annotated	43,20%
Proposed, whole fragment annotated	53,34%

Proposed method obtained similar or better results to TF-IDF based method in all performed experiments. The results of experiments with no annotations, where only the content of the document was used to create query suggest that proposed method provide similar even better results for query word extraction. These results were achieved despite the fact that proposed method is using only information from the document content and not the information about other documents in the collection by contrast to TF-IDF based method. The proposed method can thus be used as an alternative to TF-IDF based method when creating query from document content.

The comparison of both methods using simulation proved that proposed method can create queries that can be used to retrieve similar documents with significantly higher precision than compared method. The experiments with whole document fragments annotated suggests, that with increasing number of annotations the precision of generated queries increases.

4. Conclusions and future work

We proposed a method for query construction from document content and attached annotations. In the process of query construction we considered document content and its structure by using text to graph transformation and query terms extraction using spreading activation in created graph. We used user created annotations to insert initial activation to document parts user is most interested in. The simulation based on probabilistic distributions of various parameters of annotations created by users of Annota proved, that the proposed method outperforms TF-IDF based method when creating query for related documents search from source document and attached annotations. The proposed method achieved slightly better results when no annotations were used and it outperformed compared method when document-attached annotations were used in query construction. The proposed method does not use information from other documents, only information from source document content and attached annotations. It is thus search engine independent and can be used to create queries for any search engine accepting queries in form of list of keywords.

In the future work we plan to use annotations attached not only by document reader but also by other users when creating query for related documents. We see potential in use of social relations such as group membership in weighting of annotations created by other users in query construction process. Moreover there are several possible enhancements of index creation process using annotations. We plan to use annotations to enrich document content while creating index of annotated documents and we will compare performance of search in annotation enriched index against related document search in index created using only document content.

We showed it is possible to use annotations as indicators of user's interest. We plan to use user's previous annotations for disambiguating users interests and we will use them for query expansion similarly to work presented in [3] where they used various user activity indicators and social context for disambiguating search and for user query expansion.

Acknowledgements. This work was partially supported by the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

[1] M. Agosti and N. Ferro. A formal model of annotations of digital content. *ACM Trans. Inf. Syst.*, 26(1), Nov. 2007.

[2] G. Golovchinsky, M. N. Price, and B. N. Schilit. From reading to retrieval: freeform ink annotations as queries. In *SIGCHI Bulletin*, pages 19–25. ACM Press, 1999.

[3] T. Kramár, M. Barla, and M. Bieliková. Disambiguating search by leveraging a social context based on the stream of userâĂŹs activity. In *User Modeling, Adaptation, and Personalization*, pages 387–392. Springer, 2010.

[4] D. Millen, M. Yang, S. Whittaker, J. Feinberg, L. Bannon, I. Wagner, C. Gutwin, R. Harper, and K. Schmidt. *Social bookmarking and exploratory search*. Springer London, London, 2007.

[5] R. Móro and M. Bieliková. Personalized text summarization based on important terms identification. In *Database and Expert Systems Applications (DEXA), 2012 23rd International Workshop on*, pages 131–135. IEEE, 2012.

[6] D. Paranyushkin. Visualization of Text's Polysingularity Using Network Analysis. *Prototype Letters*, 2(3):256–278, 2011.

[7] Á. R. Pereira and N. Ziviani. Retrieving similar documents from the web. *Journal of Web Engineering*, 2(4):247–261, 2003.

[8] M. Šimko, M. Barla, V. Mihál, M. Unčík, and M. Bieliková. Supporting collaborative web-based education via annotations. In *World Conference on Educational Multimedia, Hypermedia and Telecommunications*, volume 2011, pages 2576–2585, 2011.

[9] J. Ševcech, M. Bieliková, R. Burger, and M. Barla. Logging activity of researchers in digital library enhanced by annotations. *7th Workshop on Intelligent and Knowledge oriented Technologies*, pages 197–200, Nov. 2012, (in Slovak).

[10] Y. Yanbe, A. Jatowt, S. Nakamura, and K. Tanaka. Can social bookmarking enhance search in the web? In *Proceedings of the 2007 conference on Digital libraries - JCDL '07*, page 107, New York, New York, USA, June 2007. ACM Press.

[11] Y. Yang, N. Bansal, W. Dakka, P. Ipeirotis, N. Koudas, and D. Papadias. Query by document. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 34–43. ACM, 2009.

[12] X. Zhang, L. Yang, X. Wu, H. Guo, Z. Guo, S. Bao, Y. Yu, and Z. Su. sDoc : Exploring Social Wisdom for Document Enhancement in Web Mining. *Proceedings of the 18th ACM conference on Information and knowledge management*, 2009.