Tag Cloud Control by Latent Semantic Analysis

submitted by
Angelina Velinska

supervised by
Prof. Dr. Ralf Möller
Dipl. Ing. Sylvia Melzer

Software Systems Institute (STS)
Technical University of Hamburg-Harburg

Dr. Michael Fritsch
CoreMedia AG
Hamburg

Contents

List of Figures

Chapter 1

Introduction

This work investigates the implementation of Latent Semantic Analysis (LSA) for discovering the main concepts in texts, in order to present a content overview of a text, or a collection of texts in the form of a tag cloud.

1. introductory words, why is this work being written
2. mention information retrieval, lsa, tag clouds - generally
3. mention cms ? document collections ? content ?
4. mention the work of david mugo

During the last decade there have been constant optimizations in information retrieval effectiveness, making web search the preferred source of finding information. A substantial part of information retrieval deals with providing access to unstructured information in various domains. Information retrieval (IR) refers to finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers) [?]. Many people today use methods from the field of IR when they use a search engine online, or search through their emails. In this context "unstructured data" refers to data which does not have a clear structure. As an example, unstructured data is the opposite of the structured data stored in a relational database.

Information retrieval systems vary by the scale at which they operate. Three main types of IR systems can be distinguished - for web search, for personal IR, and systems performing domain-specific search. In web search IR systems, IR is done over billions of documents, stored on millions of distributed computers. Personal IR is performed at the level of consumer operating systems. And in the case of domain-specific or enterprise search, IR tasks are executed on collections of documents stored on centralized file systems, while search over the collection is provided by a handful of dedicated server machines.

IR technologies find wide application - in search engines, for browsing or filtering document collections, for further processing a set of retrieved documents. Before retrieval the documents are indexed, otherwise at each search, they would have to be scanned through for each query. Generally said, the index maps the words or terms back to the documents where they occur. An interesting technique for document indexing and retrieval, which will be applied in this work, is called latent semantic analysis (LSA). It indexes the document collection by representing it as a reduced matrix of words and documents. LSA representation improves IR performance with respect to a basic problem of word-matching search - synonymy, or the case when more than one term describe the same concept.

While IR deals with retrieval of documents, other systems manage content, such as documents. Content management includes a set of technologies and processes that support the creation, management and publication of content in any form or medium. Content may be documents, multi-media files, or any other file types that follow content lifecycle and require management. Content management systems (CMS) vary depending on their purpose and target environments - there are CMS for the web, for enterprise, for mobile devices, as well as CMS for management of document collections, also called document management systems (DMS).

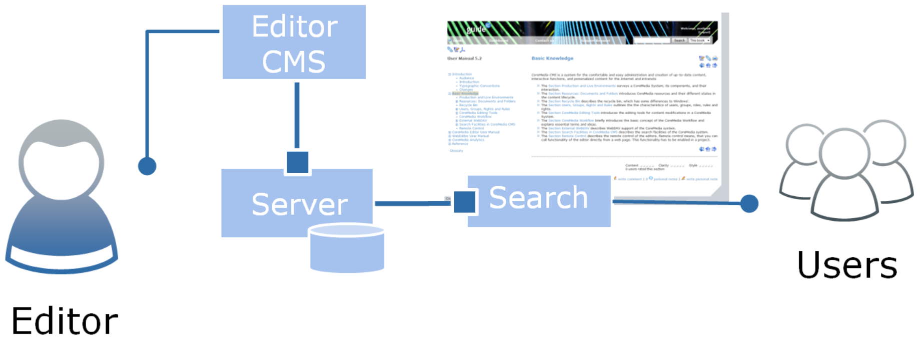

Figure 1.1: Online Document Management System

Figure **??** shows a simplified document management system, offering online access to its content. Using the editor CMS, editors can create, edit or delete content, which is managed and stored by a Content Management Server. Content is presented to the end users, who can access the document collection and search through it according to their information need. CMS usually consists of an environment for content management, and an environment for content delivery or presentation. However, in the introduction we only outline the basic concepts of the system.

1.1 Motivation and objective

A drawback of the classical LSA implementation as an information retrieval method is the low precision of the returned results. A previous work by David Mugo [?] has investigated the improvement of LSA precision performance by annotating the document collection and including the anotations used in LSA. In his work, Mugo constructs a concept-document matrix from the annotations used, and concatenates it with the word-document matrix normally generated in LSA process. The proposed solution, however, results in a slow speed of LSA, and has left Mugo's hypothesis open.

Taking into consideration the results from Mugo's work, the current project has several objectives to reach. It will investigate the implementation of LSA method for improving information retrieval in a domain-specific document management system with respect to context-based search. A further investigation will be made on improving the precision performance of LSA method by using semantic annotations, and on finding an adequate way to present the results of LSA as a tag cloud. And finally, it will be investigated how to use the tag cloud as a form of a relevance feedback to control LSA method.

In the context of the stated objectives, semantic annotations are meta data annotations used to add information to unstructured data, or to the document collection. Semantic annotations are based on an ontology in our case, specifically developed for the domain of interest CoreMedia CMS. Ontologies are used to capture some knowledge about a certain domain, by describing the concepts of the domain and the relationships between them. To further clarify the objectives, relevance feedback is an IR technique, used to influence the retrieved results based on the user's preference. It allows the user to modify the initial tag cloud by selecting the most relevant words. The tag cloud is then re-generated from LSA results with the relevance feedback posted as a query.

1.2 Outline

The reminder of this work is organized as follows. Chapter **??** describes in more detail what a document management system is, and provides an overview of the general structure of DocMachine 2.0, the DMS deployed at CoreMedia AG. Chapter **??** presents the basic concepts of ontologies and document annotations based on ontologies. In Chapter **??** an overview of latent semantic analysis method is given, as well as an approach for improving LSA's precision by including semantic annotations in the method. Chapter **??** presents the prototype implementation and makes an evaluation of the results achieved in this work. And finally, conclusions are drawn in Chapter **??**, along with some limitations of the current study and outlook for a future research.

Chapter 2

Latent Semantic Analysis

Summary. *This chapter presents Latent Semantic Analysis as a method for text processing, and defining the main concepts in texts.*

Latent Semantic Analysis (LSA) uses a Singular Value Decomposition (SVD) to construct a..

at the end- improvements of lsa with the basics explained. why am i using lsa instead of lda for example?

2.1 plan

1. text processing and peculiarities; stemming, lemmatization, stopwording
2. lsa and basics
3. weighting functions and their effect ot LSA results [**?**].
4. lsa used for information retrieval; lsa used for defining the main concepts in texts. precision vs. recall.
(first explain the basics of LSA, then explain how factors can influence lsa)
Several factors influence the quality of results which LSA delivers. These factors are pre-processing (removal of stop-words, stemming, lemmatization), frequency matrix transformations, choice of dimensionality, choice of similarity measure.
A study by Nakov, Popova, Mateev[**?**] has summarized the influence of those factors on LSA, and has concluded that...

2.2 storage

Text Processing and LSA

Text processing:
- retrieve documents from DB
- tokenize texts
- stem/lemmatize texts - this drops off as we will use the terms as a part of a tag cloud
- stop wording
- build SVD
- post queries on the matrix

The document collection consists of guides and manuals about Core-Media Content Mangement System 5.2.

Improving performance of LSA information retrieval method includes tf*idf weighting scheme, relevance feedback by implementing Tag Cloud, and choosing the number of dimensions for the reduced spacing. Stemming as a method for LSA improvement is not applied, as investigations showed at most modest improvements with this method.

Library/implementation used for LSA is S-Spaces from Airhead Research project of UCLA (University of California at Los Angeles). The implemented algorithm for SVD is Lanczos, ported from SVDLIBC implementation by Doug Rohde from Tennessee University.

Use the paper ”Weight functions impact on LSA performance” by Preslav Nakov, Antonia Popova, Plamen Mateev - very nice concise description of LSA + analysis.

IMPORTANT
I should test entropy and idf , as sometimes entropy global weighting function has a better performance.

For text processing, Snowball project is used, from the laboratory of Martin Potter, the author of the infamous Porter Stemming algorithm. !!! No stemming or lemmatization should be done on the input document collection, as the resulting terms/tags from LSA will be used in a TagCLoud!

11818 words in word space

63552ms to run LSA on 4000 documents
and IDEA blocks 9

Due to the problem above, the process of SVD calculation has to be performed in a multi-threaded way, and the project has to be optimized with respect to performance, in order to be able to successfully run.

Chapter 3

Tag Clouds

Summary. *This chapter presents an overview of tag-clouds used as a method for representing text content.*

Tag Clouds are popular applications used for vaious purposes: annotation of documents, overview of textual content of websites ??. The importance or weight of words in the tag cloud are shown with size of font and/or color. The tag clouds are hyperlinks leading to a collection of items associated with the tag.
There are three types of tag clouds depeding on their

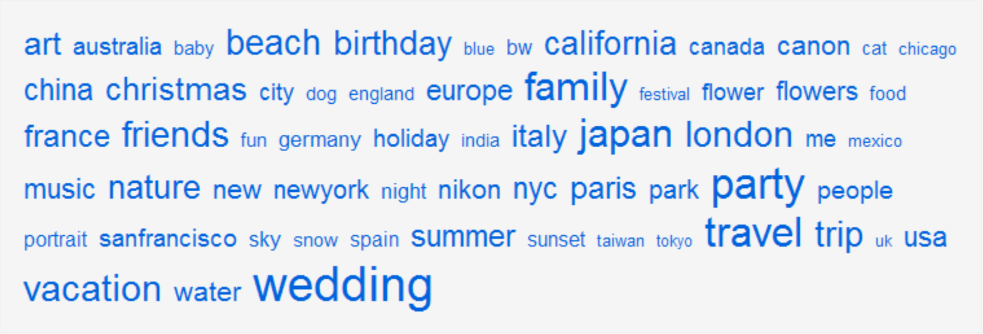
art australia baby beach birthday blue bw california canada canon cat chicago
china christmas city dog england europe family festival flower flowers food
france friends fun germany holiday india italy japan london me mexico
music nature new newyork night nikon nyc paris park party people
portrait sanfrancisco sky snow spain summer sunset taiwan tokyo travel trip uk usa
vacation water wedding

Figure 3.1: Tag Cloud

3.1 1

Related work SenseBot Search Results Summarizer is a plugin for Mozilla Firefox browser that generates a tag cloud of the main concepts returned as search results from Google.

LinkSensor SenseBotSummarizer All three are based on SenseBot - a semantic search engine. Made available from Semantic Firefox Extensions [1]

[1] http://www.semanticengines.com/plugins.htm

Chapter 4

Implementation and evaluation of results

Summary. *summarize me*

4.1 1

Chapter 5

Conclusion and outlook

Summary. *summarize me*

5.1 1

Appendix A

Some Code

You should not print the full source code :-). But note that the chapters are now called “appendix” and numbered with letters.

Bibliography

[1] C. D. Manning, P. Raghavan, and H. Schtze, *Introduction to Information Retrieval.* New York, NY, USA: Cambridge University Press, 2008.

[2] D. M. Mugo, "Connecting people using Latent Semantic Analysis for knowledge sharing," Master's thesis, Hamburg University of Technology, Jan. 2010.

[3] P. Nakov, A. Popova, and P. Mateev, "Weight functions impact on LSA performance," in *EuroConference RANLP'2001 (Recent Advances in NLP*, pp. 187–193, 2001.