Information Retrieval and Processing-Rainy report

Dickson Sareddy Rain

June 15, 2017

1 Introduction

1.1 Aim

For this year's project, we as team Rainy need to implement full text search function in an article database and get a presentation of extracts of the most similar parts in decreasing parts in decreasing order with the source mentioned on Harvard citation format with the title of the publication, its type and link to the publisher's full text PDF and my library's full text PDF.

This section should be updated based on what you are building.

2 Background

3 Methods

3.1 LSA

3.2 Similarity comparison

3.3 Metadata extraction

Metadata Extractor is the module which can extract metadata in PDF files such as title, subheading, doi, etc. We use python package pyPdf to extract metadata directly and use metadata to do couple things. First, we extract titles of articles returning those to the web page as search results. Second, we use metadata extractor to extract subheadings which are treated as the definition of break point to split PDF file i.e. one article will be separated into small pieces based on subheading. The small pieces will convert into .txt files by the .txt converter we built so that we can take .txt files into similarity comparison and get the most relative parts in the article.Third, We extract doi [explain what is doi] which can link to the original article source.

4 Results and discussion

5 Conclusions and future work

The results of this year are described below. We built a functional web server which contains full text search function, similarity comparison, and the article hyperlink which can connect to the source web site.