CMPSCI 585 --- Fall 2016 --- Project Proposal

Pankaj Bhambhani, SPIRE ID - 30566626

Title - Content Summarization of a Book using its Index

Collaborators - NONE

Objective

With the globalisation of internet in today's world, we have an explosion of digital information all over the globe. A lot more people are now interested and curious in getting knowledge and information about various fields than previously, thanks to openly available content (for e.g. via Massive Open Online Courses - MOOCs as they call them). Though books have always been the key source of crucial information (and continue to be), these new generation of explorers prefer newer methods of learning, such as online blog posts and YouTube video tutorials and they find books pretty long to read (no wonder they've invented a new term for long explanation texts - TL;DR "Too Long; Didn't Read"). We want to generate a summary of the contents of a book using its Bibliographical Index as an attempt to bridge the gap between the two options. We want to allow people to be able to read (or at least attempt to read) books in a way that works best for them. The goal is to be able to generate a summary that allows a 400-pages book with a good index and a good table of contents to be read "in an hour". The reader can then use the summary and the index to lookup specific topics in the book, making him/her feel that he has already gone through the book once.

Scope

Automatic Text Summarization is one of the actively-researched fields in Natural Language Processing. The most commonly used methods for this can be roughly divided in three stages - 1) Extraction of relevant and important keywords from the text, 2) Extracting relevant phrases/sentences based on these keywords and other factors, such as its relevance to the title/subtitle and to the core-idea of the text, and 3) Generate natural language sentences which represent the extracted phrases and produce a summary of the text. In our case, we are dealing with a specific collection of document, i.e. books with table of contents and an index. We plan to use the index and the titles/subheadings as the relevant keywords instead of deriving them as in step 1), and then we use steps 2) and 3) as described above.

Prior work

I plan to use the following work for the project:

1) Al-Hashemi, Rafeeq. "Text Summarization Extraction System (TSES) Using Extracted Keywords." *Int. Arab J. e-Technol.* 1.4 (2010): 164-168. This paper summarizes the key steps

of summarizing texts using the steps we described above, namely parsing the text, extracting keywords, ranking sentences and then generating summary.

2) Steinberger, Josef, and Karel Ježek. "Evaluation measures for text summarization." *Computing and Informatics* 28.2 (2012): 251-275. This paper presents various ways of evaluating content summary. I plan to use this for evaluation the generated summary of the book.
3) Text Summarization using Tensorflow https://research.googleblog.com/2016/08/text-summarization-with-tensorflow.html I am considering using this library for text summarization though I am considering alternatives.

Data set

There are a lot of free e-books available on the internet, but the number of books with an Index are limited, so some time needs to be spent on filtering out such books. I plan to use Project Glutenberg for bulk sections of ebooks. In case I don't find enough books with an index, I plan to extract data from the ebooks available in the UMass libraries (they have ebook versions of many books).

Annotation

I plan to rely on the already annotated index of a book, so there's not much work required there, although, the contents of the book will have to extracted out - its index, table of contents and the remaining text. There are open-source libraries that can help in this process, I plan to use pdfrw (https://github.com/pmaupin/pdfrw), although I am still looking at other alternatives.

Algorithm/Experiment

Preprocessing the data is a key phase of this process. I need to extract the 1) index as a list of words and its associated locations within the document 2) table of contents to get a list of heading and subheadings and include them in my keywords list.

I then plan to extract relevant paragraphs containing the indexed keywords, and also plan to include surrounding paragraphs to maintain some of the context. I then plan to use this as my data set for summarization - to extract phrases/sentences that are relevant to the central idea, and finally generate natural language sentences for summarization using these phrases.

A key phase of this experiment would be validation against existing summary. I plan to use the author abstracts, introduction and conclusions and compare it with a summary generated by the algorithm and try to evaluate the result. This is still unclear, because an abstract might not always be the same as the summary. Another way could be to ask a reader of the book to generate a summary of equivalent length and then compare the results against that. I am open to suggestions here and looking for feedback on how to better validate my results.