Honours Year Project Report

Scientific Document Summarization Exploiting Citing Documents

By

Patrick Chen

Department of Computer Science

School of Computing

National University of Singapore

2014

Honours Year Project Report

Scientific Document Summarization Exploiting Citing Documents

By

Patrick Chen

Department of Computer Science

School of Computing

National University of Singapore

2014

Project No: 0
Advisor: 0
Deliverables:
Report: 1 Volume

Abstract

Abstract eventually goes here.

0.1 Introduction

The volume of scientific literature has grown rapidly. And there is a need to create an automatic summarization system which enables researchers to digest knowledge in a short time. In fact, Summarization is not a new field in NLP community; however, lots of previous work are done in the data like news, and how to summarize scientific documents is still under discovering.

Scientific documents have two distinct characteristics. First, its structure must follow certain conventions. That is, you will definitely see the sections arranged like introduction, previous work, method, experiments, discussion and so on. This provides a significant advantage for a summarization system, in being able to leverage this structure. Second, scientific documents will contain a special dimension unlike other texts – citations. Scientific achievements are an accumulation of knowledge, partially based on learning from other publications by other members in the community, to make their contribution; therefore, many explanations and criticism will be made by the community through citations.

The second attribute makes scientific document summarization extremely special and attractive to me. Many previous works solve it by either natural language or information retrieval methods and it produces promising results. So in this project, I will focus on using citations to make a good summary.

0.2 Summarization using Citing Sentences

Using citations to generate a summary is not a new idea, as past research have explored this direction. (S. Teufel, 2005) studies the argumentative zoning for improved citation. (S. Teufel et al., 2006) tries to understand the function of citation sentences and utilizes machine learning approach to do automatic classification. (V. Qazvinian et al., 2010) summarize by extracting significant key-phrases from the set of citation sentences. (V. Qazvinian & D. Radev, 2008) model the citing sentences in a graph and cluster it with Lexrank method. Besides, the effectiveness of citation is verified by (A. Elkiss et al., 2008) and (Mohammad et al., 2009). Inside both study, they use different measures to point out the fact that there are indeed some valuable information in the citation which cannot be obtained from the source papers.

0.3 System Description

In this project, we are going to create our own system which tries to improve and combine some methods mentioned above.

0.3.1 General System Requirements

In this section, I am going to discuss the basic functions that the baseline system should have, and in the next section, my baseline system will be explained based on the structure introduced in this section. In general, a complete summary is generated using the following pipeline:

(a) Parsing source paper: Parsing the source paper to get some important

information such as title and authors of the paper, or anything that might be useful in the later stages. Save these data and proceed to the next step.

(b) Find out all citing papers: In order to obtain the citing sentences, we must collect all the papers which cite the source paper first. One thing to note is that there is no way to verify the total number of citing papers. We may use some tools or databases to find out citing papers as more as possible, but it simply cannot promise the completeness. After getting the citing papers, the system is prepared to extract information from it.

(c) Extracting citing sentences: Inside each citing paper, there must be either explicit or implicit citation. Usually the citation will be written as author plus year inside the parentheses or brackets as [Authors, Year], sometimes it might just use a number and we need to go through the reference section in the paper to know the work being cited. For the explicit citation, citing sentences are just next to these indicators so you can easily find out the correct citing sentences; however, sometimes the citing sentences are not close to the indicators, or beside the closest sentence, adjacent sentences contain more important information, and its called implicit citation. The system should specify what kinds of citation it is going to extract and the methods to detect the citing sentences.

(d) Using citing sentences to generate the summary: The most important step in the system is generating the summary. After getting all available citing sentences, we will be able to use them to summary the source paper.

(e) Evaluate the summary: After generating the summary, we need to evaluate its quality. There are many existing methods to evaluate the result. It will be discussed in details in the section 4.2.

Baseline system

In this section, I will describe the details of my baseline system by explaining each component introduced in the last section.

(a) Parsing source paper: in the baseline system, all the information needed is author and title, and this task will be handled by the ParsCit library. ParsCit is a friendly tool to extract some useful information for scientific documents. The only inconveniency is that it requires the input format to be the XML, especially the XML file generated by the OmniPage commercial software. But generally the scientific document is easily found in pdf format; therefore, all the papers need to be passed into OmniPage first before extracting the author and title of the paper.

(b) Find out all citing papers: the simplest way to track the citations of certain paper is using Google Scholar Search Engine. Although it might not be complete, it provides a satisfactory result. Originally, I planned to use a library developed by WING group at NUS to automatically download the query results from Google Scholar; however, because there are some version problems, it does not function properly right now; therefore, I decide to manually download all the pdf files from Google at this stage. Also, because I believe this stage should not cost much time, if the download link requires me to login from NUS library, I will simply discard the paper.

(c) Extracting citing sentences: After getting the citing papers pdf files, I will run the OmniPage in batch to convert pdf to XML, and then again using ParsCit to get the citing sentences. Basically, ParsCit will return you whole context of citation. That is, it will include 3 more sentences before and after the sentence where citation indicator shows. In the baseline system, I

will simply extract just the sentence containing citation indicator and omit all the others first. Although (A. Abu-Jbara & D. Radev, 2012) has shown that there might be useful information in the surrounding sentences, in the first step, it is still better for me to extract single sentence only, or to extract sentences by human.

(d) Using citing sentences to generate the summary: in this baseline system, I am going to use C-Lexrank (V. Qazvinian & D. Radev, 2008) as the method to summarize. The spirit behind C-Lexrank is to classify citing sentences into different clusters by calculating cosine similarity. And it subsequently calculates LexRank within each cluster to find the most salient sentences of each cluster. More information can be found in the original paper.

(e) Evaluate the summary: in the original paper, C-Lexrank is evaluated by the Pyramid method. The spirits of C-Lexrank is to use minimal sentences to describe as more content as possible; therefore, it directly meets the idea behind the pyramid method which calculates the percent of coverage of most important Summarization Content Units (SCUs). So it is reasonable for the original paper to use pyramid method to evaluate the C-Lexrank; however, there are two problems. First, this project is going to deliver a summary in the end, instead of only valuable sentences.

Apparently C-Lexrank needs some light alternations to meet this requirement. Second, the Pyramid method is a method which requires lots of human work. Humans need to manually annotate SCUs for both standard and system-generated summarizes. For this project, we do not have enough resources to do so; besides, if we need to tune or optimize the system in the

future, we need an automatic evaluation method to achieve it; therefore, in this project, we will not use Pyramid method to do the evaluation. On the other hand, ROUGE (Lin, 2004) has been proposed to use only few human summaries to judge the quality of summarization. It calculates recall, precision and F-measures by different units like N-grams or LCS. Although it might be unfair to directly apply ROUGE to C-Lexrank method directly in this stage, considering the resources we have, it is the best method we can have to evaluate the results.

0.4 Experiment

0.4.1 Data Preparation

The test data is obtained by randomly sampling 10 papers from ACL anthology, namely P99-1026, W10-1919, S12-1032, P07-3014, C08-1122, P10-1024, W93-0225, Y09-2051, D12-1074, O90-1002. I use their citing sentences to generate the summary; however, only C08-1122 (13 citations), P10-1024 (10), P99-1026 (24) and W93-0225 (6) have enough (at least 5) citing sentences; therefore, the experiment was only be done on these 4 papers.

0.4.2 Evaluation

We are going to generate a summary containing 4-5 sentences and around 120-150 words. I will use ROUGE as the evaluation method as mentioned above. ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation. It counts the number of overlapping units such as n-gram, word sequences and word pairs between the system-generated summary and ideal

summaries created by humans. In this project, the gold summaries are generated by group members. We will do the extraction summarization on source paper to generate summaries based on the sentences in the source paper, and there are 2 for P10-1024 and C08-1122, 1 for W93-0225 and P99-1026 now.

0.5 Results and Discussion

Due to the fact that C-LexRank originally is not evaluated by ROUGE, it might be difficult to compare to other works; nevertheless, I find out in (Mohammad et al., 2009), they use ROUGE-2 to evaluate various methods including C-LexRank; and in (Mei and Zhai 2008), they use ROUGE-1 and ROUGE-L on random selection and other popular methods as MEAD. It might be unfair to compare to their values directly, but it is good to refer their results. So I will use them as a comparison baseline to point out what's the reasonable values C-LexRank should give. The evaluation result of my C-LexRank system is shown as follows:

P10-1024 ——————————————————

3 ROUGE-1 Average_R: 0.32819 (95%-conf.int. 0.32819 - 0.32819)

3 ROUGE-1 Average_P: 0.30797 (95%-conf.int. 0.30797 - 0.30797)

3 ROUGE-1 Average_F: 0.31776 (95%-conf.int. 0.31776 - 0.31776)

————————————————————

3 ROUGE-2 Average_R: 0.04669 (95%-conf.int. 0.04669 - 0.04669)

3 ROUGE-2 Average_P: 0.04380 (95%-conf.int. 0.04380 - 0.04380)

3 ROUGE-2 Average_F: 0.04520 (95%-conf.int. 0.04520 - 0.04520)

————————————————————

3 ROUGE-L Average_R: 0.30116 (95%-conf.int. 0.30116 - 0.30116)

3 ROUGE-L Average_P: 0.28261 (95%-conf.int. 0.28261 - 0.28261)

3 ROUGE-L Average_F: 0.29159 (95%-conf.int. 0.29159 - 0.29159)

P99-1026 ———————————————

1 ROUGE-1 Average_R: 0.45763 (95%-conf.int. 0.45763 - 0.45763)

1 ROUGE-1 Average_P: 0.39130 (95%-conf.int. 0.39130 - 0.39130)

1 ROUGE-1 Average_F: 0.42187 (95%-conf.int. 0.42187 - 0.42187)

———————————————

1 ROUGE-2 Average_R: 0.10256 (95%-conf.int. 0.10256 - 0.10256)

1 ROUGE-2 Average_P: 0.08759 (95%-conf.int. 0.08759 - 0.08759)

1 ROUGE-2 Average_F: 0.09449 (95%-conf.int. 0.09449 - 0.09449)

———————————————

1 ROUGE-L Average_R: 0.42373 (95%-conf.int. 0.42373 - 0.42373)

1 ROUGE-L Average_P: 0.36232 (95%-conf.int. 0.36232 - 0.36232)

1 ROUGE-L Average_F: 0.39063 (95%-conf.int. 0.39063 - 0.39063)

C08-1122 ———————————————

1 ROUGE-1 Average_R: 0.25498 (95%-conf.int. 0.25498 - 0.25498)

1 ROUGE-1 Average_P: 0.22069 (95%-conf.int. 0.22069 - 0.22069)

1 ROUGE-1 Average_F: 0.23660 (95%-conf.int. 0.23660 - 0.23660)

———————————————

1 ROUGE-2 Average_R: 0.05221 (95%-conf.int. 0.05221 - 0.05221)

1 ROUGE-2 Average_P: 0.04514 (95%-conf.int. 0.04514 - 0.04514)

1 ROUGE-2 Average_F: 0.04842 (95%-conf.int. 0.04842 - 0.04842)

———————————————

1 ROUGE-L Average_R: 0.23108 (95%-conf.int. 0.23108 - 0.23108)

1 ROUGE-L Average_P: 0.20000 (95%-conf.int. 0.20000 - 0.20000)

1 ROUGE-L Average_F: 0.21442 (95%-conf.int. 0.21442 - 0.21442)

W93-0225 ———————————————

4 ROUGE-1 Average_R: 0.36842 (95%-conf.int. 0.36842 - 0.36842)

4 ROUGE-1 Average_P: 0.36601 (95%-conf.int. 0.36601 - 0.36601)

4 ROUGE-1 Average_F: 0.36721 (95%-conf.int. 0.36721 - 0.36721)

———————————————

4 ROUGE-2 Average_R: 0.05298 (95%-conf.int. 0.05298 - 0.05298)

4 ROUGE-2 Average_P: 0.05263 (95%-conf.int. 0.05263 - 0.05263)

4 ROUGE-2 Average_F: 0.05280 (95%-conf.int. 0.05280 - 0.05280)

———————————————

4 ROUGE-L Average_R: 0.30921 (95%-conf.int. 0.30921 - 0.30921)

4 ROUGE-L Average_P: 0.30719 (95%-conf.int. 0.30719 - 0.30719)

4 ROUGE-L Average_F: 0.30820 (95%-conf.int. 0.30820 0.30820)

I summarize the results as follow(use F score):

C08-1122 W93-0225 P10-1024 P99-1026 ROUGE-1 0.237 0.367 0.318 0.422 ROUGE-2 0.048 0.053 0.040 0.088 ROUGE-L 0.214 0.308 0.292 0.391

And the results combined both (Mohammad et al., 2009) and (Mei and Zhai 2008) are shown as follows:

Random LEAD MEAD C-LexRank ROUGE-1 0.230 0.301 0.401 X ROUGE-2 0.10 X X 0.13 ROUGE-L 0.214 0.292 0.362 X

the results generated from my system vary slightly. I believe it is because of the number of citing sentences, and the quality of the summarization. For example, P99-1026 gives a good result and it utilizes 26 citing sentences. And for W93-0225, although it only contains 7 citing sentences, all 7 sen-

tences give a concrete and focused illustration of work of W93-0225, so the result is better than C08-1122, whose citing sentences contain less valuable information.

And comparing to other methods, my baseline system generally outperforms the random and LEAD method, and performs equally to MEAD, which reflects that it's value is reasonable. About the C-Lexrank in ROUGE-2, there is a significant gap between my result and (Mohammad et al., 2009). I guess it is because their experiments are focused on Dependency Parsing papers, and citing sentences will contain more information in that kind of papers.

But comparing to method developed in (Mei and Zhai 2008) which achieves 0.467 in ROUGE-1 and 0.444 in ROUGE-L, it seems like my baseline system is not good enough yet. It also reflects the characteristics that C-LexRank is trying to enhance the diversity of citations within minimal length, but it does not consider the coherence or other factors which will affect the quality of summary.

0.6 Conclusion

In this project, I build a summarization system based on the C-Lexrank method. It will extract citing sentences from the pdf documents, and automatically compose a summary. The experiments show a reasonable result and in the meantime reflect characteristics of C-Lexrank method. To further improve the result, I will work on finding functions of sentences inside each C-Lexrank cluster and try to map citing sentences to a template in the future.