CP4101 HYP Interim Report
Citation Provenance*

Heng Low Wee

U096901R

April 2, 2012

1   Introduction

To cite previously published scientific papers is a necessary and common practice among researchers. It gives credit and acknowledgement to original ideas, and to those researchers who did significant amount of work in that particular field of research, and more importantly, to uphold intellectual property. A reader of such research papers often encounter these citations made by the authors in various sentences throughout the paper. Often enough, if a reader wishes to gain a better understanding of the current context, it is necessary to follow through these citations and read up on these cited papers. Readers would also be interested to know where the information is in the cited paper.

1.1   The Problem

Readers would read up on the cited papers to gain a better insight on the current topic of discussion. However, as frequent readers might find, most citations are only *mentions*. They do not directly refer to some particular section of the cited paper, for example, to make reference to the evaluation results made by the authors of the cited paper. Instead, they are general citations. These citations are important. However since it is not immediately clear where the cited information is from, a reader has to invest additional time to read through the entire cited paper before being able to find out what the critical information is. The difficulty is increased

---

*Project Code: H079820

when the cited document is a research journal, when the amount of content is comparably huge. The author might cite specifically using the journal's volume number, issue and page number, yet the problem remains. This makes little contribution to a reader's understanding of the current topic, and it is also time-consuming.

1.2 Our Goal

In general, we wish to improve the reading experience of scientific & research documents, from the various fields of research. We want to provide information about the provenance of a citation made in a paper. Readers will be informed of where exactly the cited information is from in the cited paper. For instance, if a citation is made to refer to the evaluation results computed by the authors of the referenced paper, the reader would be *guided* to the particular section of that paper. By *guiding*, we refer to having the section with the critical information highlighted, and so the reader can quickly read up on it, before resuming on the current paper. If the citation was a general citation, the reader will possibly be referred to the Abstract section of the paper.

We want to develop a reading tool for readers of scientific papers. This reading tool will be able to provide the reader with information about the cited papers, and depending on whether the citations are general or specific, the contents of the cited paper will highlighted accordingly, so that reader may make a quick reference to the cited paper and not interrupt the current reading.

With the above in mind, we investigate the following related works.

2 Related Works

2.1 Helping Users Make Relevance Judgements About Cited Documents

Practitioners and researchers often have very little time to read lengthy scientific papers, however it is also necessary for them to do so in order to keep themselves updated on the latest developments in their fields of work. Wan et al.[8] investigated the *literature browsing task* by conducting surveys on researchers who read scientific papers frequently to update themselves. In this initial study conducted by Wan et al., several key ideas were revealed. First, when researchers read scientific papers and see citations made by the author, their main concern, as time-constrained

professionals, is whether the cited paper would be worth their time and effort, and money, to follow up on and at the same time, whether to believe in the citation. Second, one of the difficulties readers faced was finding the exact text that justify the citation. Third, the surveys revealed that readers found it useful if a reading tool could identify important sentences and key words in the cited paper.

This study conducted by Wan et al. is based on the fundamental idea of improving the reading experience of practitioners and researchers. The goal is to save a reader's time by helping the reader make relevance judgements about the cited documents. As it is often that readers have to read up on the cited documents to gain a better insight on the current context, this task would be of relevance. The authors then developed the CSIBS[7] based on their studies. The CSIBS tool helpers reader determine whether to read on the cited papers by providing a contextual summary of the cited papers.

We are interested in the conclusions drawn from this study conducted by Wan et al. In our project, our goal is also to be able to locate the exact phrases and sentences that would validate a citation made by an author. Another key idea learnt was to be able to identify key words in the cited document. For a reader who has little time to read the entire text, it is useful for the reader to identify first what are the key points in the cited document before investing on reading the entire text.

2.2 Identifying Keywords In A Document

Much work has been done on automated extraction of keywords from documents. In Hulth[2]'s work, instead of the straightforward of determining keywords by means of term frequency, she showed that by adding linguistic knowledge it could improve the automated process of keyword extraction. In that paper, Hulth adopted a supervised machine learning approach to perform automated keyword extraction. In a later work, Hulth[3] combined machine learning techniques and natural language processing to perform automated keywords extraction. On the other hand, Zhang[9] introduced an approach to keyword extraction using CRF. In that work, Zhang demonstrated that using the CRF model outperforms other machine learning methods such as SVM. Matsuo & Ishizuka[5] presented a new keyword extraction algorithm that could be applied to a single document without the need of a corpus. In that work, they studied the importance of terms in the document by studying the co-occurrence distribution of terms in a document.

We are interested in works related to automatic keywords identification because we want to be able to locate the keywords in a document, particularly the documents cited by a paper. Since keywords in a document are also the most important terms in that document, these keywords would reside within the critical sections of the paper. Intuitively, a scientific paper is cited for its key ideas and findings, it is logical to conclude that these ideas are illustrated with the keywords. Identifying keywords in a document is an important feature that informs the reader what are the key ideas described in the cited paper, and could help the reader identify where the critical information is, and then to justify whether the citation is relevant. We highlight the work by Matsuo & Ishizuka, that their algorithm was able to achieve results comparable to a `tf-idf`[4] implementation. This presents a corpus-independent approach, even though the importance of terms is not determined by the field in which the paper belongs to.

3 Progress So Far

3.1 Analysing The Problem

3.1.1 Types Of Citation

In the scope of our project, all citations could be classified into 2 types: General, and Specific citations. We define citations as such to be inline with our goal. That is, to be able to tell, if specific, where the cited information is in the cited document. Otherwise, the citation would be deemed general. A general citation is essentially a *mention*, and the author is making reference to a paper as a whole, usually because the cited paper is of the same field of study. A specific citation is more direct. It refers to a particular section, paragraph, or even line of the cited paper. For example, this often happens in the case where the author wishes to refer to some development by the cited authors, or to make reference to the evaluation results in order to compare performance. In the field of Computer Science, it is often authors make reference to some particular computer system, or some computing algorithms, and to compare performance on speed & accuracy on solving problems.

3.1.2 Locating The Cited Information

Our problem can then be reduced to determining whether a citation is General or Specific. If a citation is general, the reader can be directed to the Abstract section of the cited paper. If a citation is specific, the reader can be directed to that specific paragraph or lines respectively. Therefore during computation, the cited document

must be broken down into *fragments*. We define a *fragment* as a chunk of k lines of the paper. Hence if given that a citation is specific, then there must exists a fragment that the citation refers to. For this we need to implement some ranking system that determines the location of this fragment. And if the ranking system is unable to determine a *best* fragment, then the citation is probably a general one.

3.1.3 Scope Of The Problem

In this project, we work on based on the fact that we are already know of the list of citations made by a paper. Clearly, this coincides with the References/Bibliography section of any paper. We abstract away the problem of locating the in-line citations in a paper, and reduce our problem to only determining the type of citations and its location. To solve the problem of locating the in-line citations, we utilize the open-source ParsCit[1] system developed by Councill et al. Conveniently, ParsCit provides the sentence that contains the citation, and these surrounding words would provide us with the context of the cite.

We will focus on determining the nature of citations, and its location in the cited document. After computing for the location of the cited information, the output of our application would be a visual display highlighting the cited information.

3.2 Modeling The Problem

In web search engines, an user enters a search query, and a search engine would use this query to search within its search domain – millions of web pages – and then display the best matching web pages as compared to the search query. That would be equivalent to having a search query for an entire corpus of research papers. Our problem can also be modeled as a searching problem, but a reduced version as compared to web search engines.

Consider reading a paper, `A`. We know the citations made by `A`, and these cited papers are listed in the References section of `A`. From this our search domain for any query from `A` would be the contents of the list of cited papers. We reduce this search domain further when we are investigating a particular citation in `A`, say now paper `A` cites the paper `B`. Now, for this citation, the scope of search would be the sub-domain – contents of paper `B`. So instead of searching for the best matching document in the corpus, we are now searching within `B`. Our problem analysis tells that we have to break down `B` into fragments, and the search query would be

for these fragments (Refer to Figure 1 for a simple illustration). With the help of ParsCit[1], the sentence that contains that cite can be extracted. The search query would then be made up of the surrounding words around the citation.
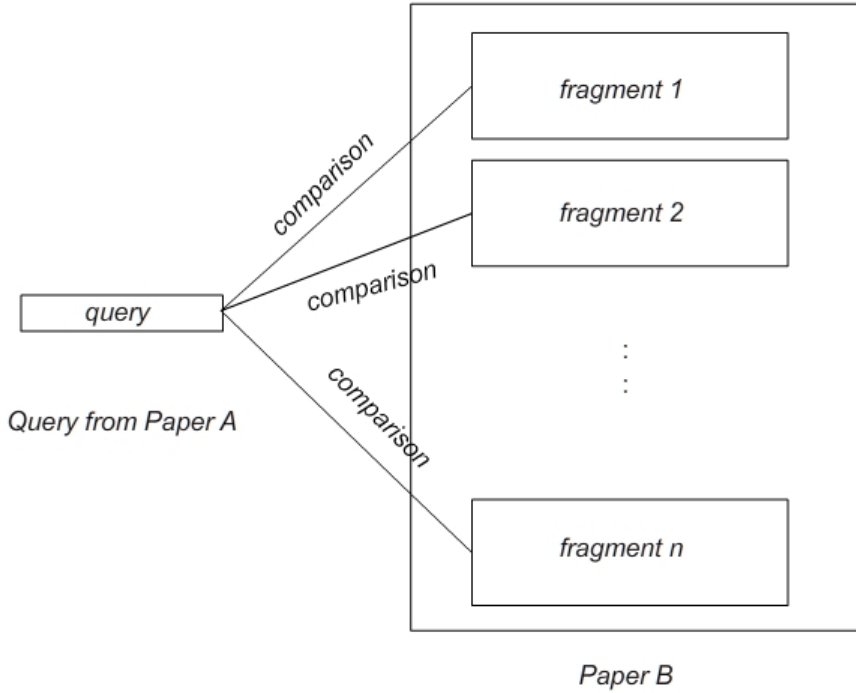

Figure 1: Modeling Our Problem

3.3 Tackling The Problem

3.3.1 Training Corpus

At this initial stage, we picked the ACL Anthology Reference Corpus[1] (ACL-ARC). The ACL-ARC consists of publications of the Computational Linguistics field. Note that in general, we wish to perform this citation provenance task on all publications from all fields of research. This corpus is chosen as a start, because it provides the interlink data that conveniently informs us of the citation-links between the papers in the corpus. For instance, in the interlink data, a link like `X98-103==>X96-1049` says that the paper `X98-103` cites `X96-1049`.

---

[1]http://acl-arc.comp.nus.edu.sg/

3.3.2 Baseline

The initial idea was to naively compare the search query by term matching frequency. Paper B will be divided into overlapping fragments of 5 lines, and then each fragment would be compared with the search query. This idea was short-lived as common terms like *to, the, is, in* etc. would get high matching frequency, and the results would be definitely skew towards fragments with high number of such terms.

We adopt a common mechanism used by search engines, that is to compare the search query and domain for Cosine Similarity[4]. In our problem, instead of comparing the query with a full document, we are comparing the query with fragments of the target paper. First, we collected the vocabulary set, V, using terms from the entire corpus. This is essential because during the comparison for cosine similarity, both vectors must have equal number of dimensions. Each dimension is represented by each unique term in V. Both the query and the fragment will be converted into vectors, v_q and v_f respectively.

We begin with assigning term frequencies to v_q and v_f. Initial tests revealed that common terms affected the comparison results and returned fragments that are not meaningful to be cited.. It becomes necessary we adopt the `tf-idf` weighting scheme in order to prevent the commons word affecting the comparison results. `tf` simply refers to *term frequency* in a document, while `idf` refers to the inverse of the *document frequency* of a term. The `df` of a term refers to the number of documents in the corpus that contains that term.

$$\texttt{tf-idf}_{t,d} = \texttt{tf}_{t,d} \times \log \frac{N}{\texttt{df}_t}, N = \text{no. of documents in corpus} \qquad (1)$$

By using this weighting scheme, the terms that appear rarely – with high `tf-idf` value – are *important*, whereas common terms that occur often, like *the* and *to*, are not. This is feasible, since in general, terms that appear often are less important, and we do not want non-important terms to affect our results.

We implemented the algorithms for computing `tf-idf` and cosine similarity ourselves instead of using the NLTK[2] package. Partly because of performance. We are able to pre-prepare information such as V, our vocabulary set, `tf` for each term

---

[2]http://www.nltk.org/

with respect to each document, and `df` for each term in V. In addition, we performed word stemming on the entire corpus. So instead of having the NLTK tool computing these information each time a query is ran, we speed up the computation process by removing this repetition, since these information remains constant for the same corpus.

4 What's Next

4.1 Gold Standard Annotations

Using the same corpus, we plan to gather gold standard annotations for the research papers. We will annotate for each cite as stated in the interlink data, whether each cite is a general or specific cite. We are particularly interested in specific citations, for that is the goal of our project – to determine citation provenance and if possible, show exactly where the information is from. Annotations for specific citations should be done down to the level of line numbers of the research paper. For instance, if the cited information appears between line 23 and 28, the annotation should capture it as `23-28`.

Collecting gold standard annotations is essential for the project. First, it is for evaluating the performance of our proposed solution. The current state of the corpus do not provide such information. Second, in order to introduce a machine learner component, we would require gold standard annotations to *train* on.

4.2 NLP Component

It is a common practice for authors, when citing another paper, to paraphrase the original sentences, often to achieve greater clarity for the current context. Sure enough, the terms in the paraphrase would be different from the original sentence written in the cited paper. The baseline we used is still a frequency-based technique to determine the importance of terms in the document. Queries are not context-sensitive, and so the result might not be able to retain similar context. Furthermore, the baseline would simply treat paraphrased queries as a mis-match. Therefore, we have to explore NLP techniques and mechanisms in order to handle paraphrasing. Shinyama et al.[6] described an approach to perform automated paraphrase acquisition from news articles, based on the scenario that these news articles are reporting on the same event. In our context, the citing and the cited

papers generally belong to the same topic. We may explore such similar approaches to tackle the paraphrasing problem.

References

[1] COUNCILL, I. G., GILES, C. L., AND KAN, M.-Y. *ParsCit: An open-source CRF Reference String Parsing Package*, vol. 2008. European Language Resources Association (ELRA), pp. 661–667.

[2] HULTH, A. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing* (Stroudsburg, PA, USA, 2003), EMNLP '03, Association for Computational Linguistics, pp. 216–223.

[3] HULTH, A. Combining machine learning and natural language processing for automatic keyword extraction /.

[4] MANNING, C. D., RAGHAVAN, P., AND SCHÜTZE, H. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[5] MATSUO, Y., AND ISHIZUKA, M. Keyword extraction from a single document using word co-occurrence statistical information.

[6] SHINYAMA, Y., SEKINE, S., AND SUDO, K. Automatic paraphrase acquisition from news articles. In *Proceedings of the second international conference on Human Language Technology Research* (San Francisco, CA, USA, 2002), HLT '02, Morgan Kaufmann Publishers Inc., pp. 313–318.

[7] WAN, S., PARIS, C., AND DALE, R. Supporting browsing-specific information needs: Introducing the citation-sensitive in-browser summariser. *Web Semantics: Science, Services and Agents on the World Wide Web 8*, 2-3 (2010).

[8] WAN, S., PARIS, C., MUTHUKRISHNA, M., AND DALE, R. Designing a citation-sensitive research tool: an initial study of browsing-specific information needs. In *Proceedings of the 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries* (Stroudsburg, PA, USA, 2009), NLPIR4DL '09, Association for Computational Linguistics, pp. 45–53.

[9] ZHANG, C. Automatic keyword extraction from documents using conditional random fields. *Journal of Computational Information Systems 4*, 3 (2008), 1169 – 1180.