Exploring Automatic Citation Classification

by

Radoslav Radoulov

A thesis
presented to the University of Waterloo
in fulfilment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2008

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Currently, citation indexes used by digital libraries are very limited. They only provide raw citation counts and link scientific articles through their citations. There are more than one type of citations, but citation indexes treat all citations equally.

One way to improve citation indexes is to determine the types of citations in scientific articles (background, support, perfunctory reference, etc.) This will enable researchers to query citation indexes more efficiently by locating articles grouped by citation types. For example, it can enable a researcher to locate all background material needed to understand a specific article by locating all "background" citations.

Many classification schemes currently exist. However, manual annotation of all existing digital documents is infeasible because of the sheer magnitude of the digital content, which brings about the need for automating the annotating process, but not much research has been done in the area. One of the reasons preventing researchers from researching automated citation classification is the lack on annotated corpora that they can use.

This thesis explores automated citation classification. We make several contributions to the field of citation classification. We present a new citation scheme that is easier to work with than most. Also, we present a document acquisition and citation annotation tool that helps with the development of annotated citation corpora. And finally, we present some experiments with automating citation classification.

Acknowledgements

First and foremost, I would like to thank my wonderful supervisor, Chrysanne DiMarco, without whom, this thesis would not be possible. I thank her for all the support, guidance, and advice.

I would also like to thank Pascal Poupart, Fred Kroon, Olga Gladkov, and Irene Chau for their help and support.

Last, but not least, I like to thank my family, girlfriend, and friends who also supported me throughout my studies.

Dedication

This is dedicated to my family and to the one I love.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Consider the situation that you, as a researcher, are new to a field (e.g., document classification) and you wish to review a selection of background articles by way of introduction, i.e., "to get your feet wet". A common approach is to search the Internet, your local library, or even consult an online document-indexing service, such as CiteSeer [24]. You manage to locate a few papers, but discover that they are not adequate as representative background for the area you are researching. Your next step might be to locate the papers referenced in the documents you have already found. You flip to the list of references and discover that the list is lengthy. However, not all references point to background material, so you return to the original text, locate the references, and check which ones refer to background material. Finally, you search for these potential new leads, and hope that these are the relevant articles that will provide the appropriate background in the field.

The process just described is usually performed repeatedly, until the researcher finds the exact document(s) she is looking for. With so many online documents now available though, this process can be very time-consuming and frustrating. However, authors cite articles for a reason [12, 13]. Consider the difference if the references were labelled with their citation *function* (or, *type*). In this case, the citations could be indexed and combined with references from other papers, thus creating a collection of papers with similar information (in our example, background material). Furthermore, the papers could be ranked by the frequency of citations, and now, if a researcher is searching for background material she will be able to locate the relevant references quickly and efficiently.

The example just given illustrates the usefulness of citation classification (labelling), and the benefits researchers could obtain from it. However, citations in scientific texts are currently not labelled, and manual labelling of all existing documents would be prohibitively costly to perform. Therefore, there is a need for an automated means of classifying citations in a predefined set of categories. The goal of an automated citation classifier is to analyze the sentences containing a citation (*citation context*) and, based on their structure or the words they contain, to label the citations with appropriate categories.

Many citation classification schemes have been developed over the years. However, most are either too general or are too difficult to use in annotation. In this thesis, we present a new citation classification scheme that is both fine-grained and yet at the same time easy for an annotator to work with.

Another problem with the state of citation research is the current lack of easily accessible corpora with pre-annotated citations. Each new research project on citation classification must begin by first searching for classification schemes that will handle its document corpora, then manually annotating a large number of citations just to have some basic training and testing sets. In this thesis, we try to resolve this problem by developing a Web-based citation annotation tool that can be accessed by anyone. We believe that an easily accessible annotation tool from the World Wide Web will make the development and collection of annotated corpora much easier.

Most current citation classification approaches look for linguistic patterns and cue words in the text to label citations. This thesis, however, also investigates the usefulness and applicability of probabilistic methods for classifying citations. Such methods have already been proven to work in similar tasks such as document classification and word sense disambiguation.

The thesis is organized as follows. In Chapter 2, we introduce the reader to background material needed to understand this thesis. Section 2.1 introduces digital libraries; Section 2.2 introduces citation classification; in Section 2.3, we discuss Information Retrieval and Word Sense Disambiguation and how they apply to citation classification; Section 2.4 presents Bayesian Networks, where we discuss its structure (Section 2.4.2), inference (Section 2.4.2), and learning (Section 2.4.4).

In Chapter 3, we look at the current state of the art in citation schemes and automated citation classification. In Section 3.1 we discuss common citation classification schemes, while we review the state of research on automated citation classification in Section 3.2.

Chapter 4 we motivate the need for more research in the fied of citation classification by presenting a preliminary experiment (Section 4.1). Then, we presents several contributions that we have made to the field of citation classification. A new citation classification scheme is presented in Section 4.2. The first Web-based tool for collection and annotation of papers for citation classification is presented in Section 4.3. We describe our annotated training/test corpus of 100 articles in biochemistry in Section 4.4, and in Section 4.5, we present the set of lexical and article-wide features that we will use for experiments.

Finally, we present a simple classification experiment with our features and annotated corpus in Chapter 5, before taking about future research and concluding in Chapter 6.

Chapter 2

Background

2.1 Digital Libraries and Citation Indexing

The World Wide Web has revolutionized research on digital libraries. With there now being an abundant number of online scholarly articles, the goal of digital libraries is to gather all relevant articles in one place. An example of a digital library is the Digital Library of the Association for Computing Machinery (ACM) [1]. It contains a collection of citations and full-text articles from ACM journals and conference proceedings and allows users to browse and search quickly for a specific article. Along with these basic features, for every article, the library also lists its references and all (known) articles that have cited it. This is accomplished by citation indexing [21] and allows researchers to go backwards in time through the list of references, or forward in time through the list of citing articles to find more recent information.

Citation Indexing is the process of cataloguing the citations that an article contains, linking the citing article with the cited articles. Eugene Garfield was one of the pioneers of citation indexing in the late 1950s and is the father of the first science citation index, the *Science Citation Index*, which was developed by his company, *The Institute for Scientific Information*, in the 1960s. Besides the obvious advantage of citation indexing—the ability to go forward and backward in time through the citation links—citation indices can also be used to analyze research trends, identify emerging areas of science, and determine the popularity of an article by the number of times it has been cited [29]. Such indices were compiled manually at first, but with the advance of technology and the transition to machine-readable electronic formats such as Postscript/PDF, indexing and retrieval has become completely automatic. For example, a digital library such as CiteSeer [24] uses many algorithms to search and index scientific literature. Briefly, CiteSeer locates scientific articles across the Web, indexes the full text of the article, performs autonomous citation indexing, extracts key information from the article such as title and author names, extracts a summary of the document, locates related documents, and more. For more information and details on the inner workings of CiteSeer, see

[28], [24], and [27].

The most interesting aspect of document indexing as it relates to this thesis is the Autonomous Citation Matching component [27]. This phase is now performed by most Digital Libraries like CiteSeer and search engines like Google Scholar [3], but it is mainly concerned with identifying citations in the article, matching each one with its corresponding reference in the bibliography section, and then parsing the bibliography item to search for the referenced document. However, this process does not distinguish between the functional types of the citations and I believe that this is the most significant limitation of the digital library system as it currently stands.

2.2 Citation Classification

In 1965, Garfield observed that there are numerous reasons for providing reference citations in papers [20]. He enumerated 15 categories (that we will review in more depth in the Literature Review section) that reference citations may fall into. However, until this day, no digital libraries distinguish between types of citations in their citation indexes. All citations, no matter whether they are *supporting*, *negative*, or some other form of relation to the citing paper, are lumped together as if they were the same. The researcher is shown only the citations contained in the article, and sometimes the context around the citation, but is left to determine the particular type of citation and whether he is in fact looking for that type.

As an example, consider the following two sentences from the same paper:

> Tight junctions (TJs) constitute the epithelial and endothelial junctional complex together with adherens junctions and desmosomes and are located at the most apical part of the complex ([B1]). TJs have dual barrier and fence roles.

> In addition to claudins and occludin, another type of integral membrane protein, JAM (junctional adhesion molecule) belonging to the immunoglobulin superfamily, was also reported to be concentrated at TJs ([B22]), but this molecule did not appear to constitute TJ strands per se but to laterally associate with strands ([B23]).

There are three references in the two passages above. In the first passage, the author gives a definition for *Tight junctions*. This reference type is usually referred to as *specific background* and is very different from the next two references.

The second passage contains two citations. For the first one (B2), the author references a *positive supporting* result from a previous work, but with the second citation, he *contrasts* the results.

With this example, the limitations of the present state of digital libraries can be clearly seen. Currently, all three citations above would be lumped together, but they all have different purposes. If a researcher was looking for background information on *Tight junctions*, he would want to look only at the first reference, and disregard the other two. Citation classification aims at resolving this problem by distinguishing the types of citations. The process requires first defining a set of citation types (categories) that reference citations fall into. The ultimate goal is to build and train an automated classifier that takes as an input a scholarly article and outputs the referenced citations in the paper together with their citation categories. This output will allow us to develop typed citation indices for digital libraries that will include more useful information about the purpose of citations in scientific articles and that will improve literature search techniques.

Needless to say, citation classification is very complex and this is a key reason why there are currently no digital libraries that include reference citation types in their citation indexes. We will review the state of the art of citation classification in the next chapter.

2.3 Information Retrieval and Word Sense Disambiguation

We will refer to Information Retrieval (IR) and Word Sense Disambiguation (WSD) concepts throughout this thesis. Here is some background information on these topics.

2.3.1 Information Retrieval

Baeza-Yates *et al* [10] describe Information Retrieval as the representation, storage, organization, and access of information items. In our case, the information items are citation types, and the Information Retrieval system is a binary citation classifier for each citation type or class. Looking at citation classification as a collection of binary classification tasks, we can use evaluation measures from IR for citation classification. The most common evaluation measures that we will also use throughout this thesis are:

- **Precision:** What is the percentage of the relevant retrieved documents.

$$Precision = \frac{Relevant \cap Retrieved}{Retrieved} \tag{2.1}$$

- **Recall:** What percentage of all relevant documents are retrieved.

$$sRecall = \frac{Relevant \cap Retrieved}{Relevant} \tag{2.2}$$

- **F-measure:** Combines precision and recall in one measure (weighted mean of precision and recall).

$$F = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2.3}$$

To undestand how these evaluation measures apply to binary citation classifiers, consider the confusion matrix for binary classification problems:

		Classified	
		True	False
Real	True	TP	FN
	False	FP	TN

Table 2.1: Confusion matrix for binary classification

Applying the confusion matrix to the above equations, we get:

- **Precision:** What is the percentage of the correctly classified citations.

$$Precision = \frac{TP}{TP + FP} \tag{2.4}$$

- **Recall:** What percentage of all citations with a given type are classified as citations of that type

$$Recall = \frac{TP}{TP + FN} \tag{2.5}$$

- **F-measure:** is the same as above.

2.3.2 Word Sense Disambiguation

Our reason for including Word Sense Disambiguation in this discussion is because we believe the citation classification task is very similar to word sense disambiguation. Moreover, proper training of citation classifiers should have a WSD task as a prerequisite. WSD will be discussed throughout this thesis so it is important for the reader to become familiar with the concept.

Word sense disambiguation is necessary because natural language is inherently ambiguous. The same word can have different meanings depending on context and for many Natural Language Processing tasks, determining the sense of every word is a prerequisite. Consider the example of machine translation. An accurate translator is very difficult (if not impossible) to build if the senses of the words to be translated are not known. For example, consider the following sentence:

> John felt very good on the day of his exam and he passed the *bar* with ease.

In the above sentence, the word *bar* is ambiguous. WordNet has nineteen synsets for *bar*, four of which are verb synsets and the rest are nouns. Here is an abbreviated sample:

- a room or establishment where alcoholic drinks are served over a counter
- a counter where you can obtain food or drink
- a rigid piece of metal or wood; usually used as a fastening or obstruction or weapon
- measure, bar (musical notation for a repeating pattern of musical beats
- an obstruction (usually metal) placed at the top of a goal
- the act of preventing
- the body of individuals qualified to practice law in a particular jurisdiction

The task of disambiguating this word for a human might be fairly easy (after all, the sentence talks about an exam), but it is not as trivial for automated word sense disambiguation. First, the *context window* containing the ambiguous word must be extracted. This context window can vary in size and is dependent on the implementation of the classifier. Next, this context has to be parsed to find the part-of-speech (POS) tags of the ambiguous word and those surrounding it. Then, the WSD classifier has to label the word with the correct sense based on features that it finds in the extracted context. There are several ways to approach the WSD classification task. Some of them are: information-theoretic, dictionary, and Bayesian.

The *information-theoretic* approach, also called *rule-based* (similar to the rule-based citation classification methods mentioned in the next chapters) is based on finding a single or more concrete features in the context of the ambiguous word that exactly identifies the sense. The classifiers are usually rule-based and the rules are designed by domain experts (human linguists). This type of work was performed in the early 1970s when the field was first emerging. More details on this early WSD work can be found in [49]. There, Weiss describes one of the earliest WSD systems, developed by himself, that was able to disambiguate only a few words. Although the rule-based approach is fairly straightforward to implement, it is very time-consuming and costly to construct the rules.

The *dictionary*, also known as the *knowledge-based*, approach makes use of dictionary definitions of the ambiguous word and searches for matching patterns (words) in the context of the ambiguous word and its definition. Early work using this approach was performed by Lesk [30]. The dictionary apporach is still being used today for WSD either on its own or in combination with other methods. The great advantage of this approach is that it is absolutely unsupervised. Electronic

dictionaries are now in abundance and have been developed for tasks other than WSD. Unfortunately, this approach is not useful for citation classification as there is no variety of dictionaries with definitions of citation types that can be used.

Finally, the *Bayesian approach* makes use of probabilistic classifiers, where a probabilistic network is trained to disambiguate a word by learning from annotated examples of the use of the different senses of the word. In other words, sentences containing sense-tagged (only for learning purposes) ambiguous words are fed to the classifier and thus the classifier learns from examples. This process is called *supervised learning* and is a popular word sense disambiguation method because it does not require a human expert to first define the rules for the classifier. The drawback, on the other hand, is the need for a large collection of annotated corpora on which the classifier can be trained. This is a problem that we address later on in this thesis for automated citation classification. More detail on Bayesian networks is presented in the following sections.

2.4 Bayesian Networks

In the Experiments section of this thesis, we describe how we trained Bayesian Network classifiers for the task of citation classification. This experimental work motivates the current section. The material researched for this section includes: [26], [18], [19], [33], [44], and others.

2.4.1 Definitions

A few definitions before we begin:

- **Conditional Probability** $P(A|B)$: The probability of an event A occurring given that some other event B occurs. It is calculated by

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \tag{2.6}$$

- **Multiplication rule**: follows from (2.6)

$$P(A \cap B) = P(B)P(A|B) = P(A)P(B|A) \tag{2.7}$$

- **Chain Rule**: follows from (2.7)

$$P(A_1 \cap ... \cap A_n) = P(A_1)P(A_2|A_1)P(A_3|A_1 \cap A_2)...P(A_n| \cap_{i=1}^{n-1} A_i) \tag{2.8}$$

- **Independent events**: Two events are independent if:

$$P(A \cap B) = P(A)P(B) \text{ or } P(A) = P(A|B) \text{ or } P(B) = P(B|A) \tag{2.9}$$

- **Conditional Independence**: Events A and B are conditionally independent given C, if:

$$P(A \cap B|C) = P(A|C)P(B|C) \tag{2.10}$$

- **Bayes Rule**: Let us calculate $P(B|A)$ in terms of $P(A|B)$ based on the multiplication rule above

$$P(B|A) = \frac{P(B \cap A)}{P(A)} = \frac{P(A|B)P(B)}{P(A)} \tag{2.11}$$

2.4.2 Structure of Bayesian Networks

Bayesian Networks are directed probabilistic graphical models. Nodes represent random variables ($\mathbf{X}$) and links between nodes (Graph G) represent conditional dependency between the nodes, while the absence of links represent conditional independence. The entire Bayesian network (nodes $\mathbf{X}$ and the links in G) is a representation of the joint probability distribution $P(\mathbf{X})$ of the random variables $\mathbf{X} = \{X_1, ..., X_n\}$. This type of model does not store the joint probability of all random variables, but instead stores local conditional probabilities. The joint probability distribution of $\mathbf{X}$ can be calculated as the product of all local probability distributions within the graph as follows:

$$P(\mathbf{X}) = \prod_{i=1}^{n} p(x_i|Pa_i), \text{ where } Pa_i \text{ are the parents of node } x_i$$

An example of a simple Bayesian network is presented in Figure 2.1. In this Bayesian network, we have four random variables:

$$\mathbf{X} = \{Cloudy, Rain, Cold, Swim\}$$

These random variables are joined with edges representing conditional independence, or causality. There are several ways of constructing a Bayesian network by hand. The first one is to enumerate all conditional dependencies and determine whether there is really a dependency. For example:

$$p(Rain|Cloudy, Swim) = p(Rain|Cloudy)$$
$$p(Swim|Cold, Rain, Cloudy) = p(Swim|Cold, Rain)$$
$$\text{and so on ...}$$

Designing simple Bayesian Nets is also intuitive as causal relationships often correspond to edges in a Bayes Net (or conditional dependence). This way, edges are drawn from causes to their effects. For example, in Figure 2.1, $Cloudy$ is a cause for $Rain$, and $Swim$ is dependent on $Rain$ and $Cold$.

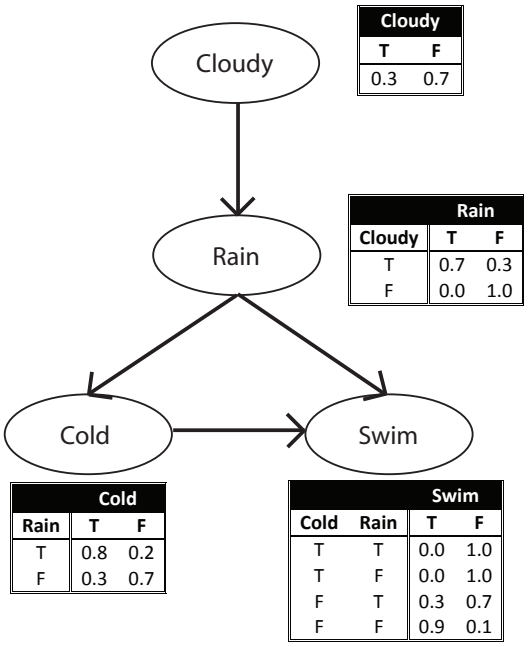


Figure 2.1: A simple Bayesian Network with conditional probability tables

2.4.3 Inference

Continuing with the example in Figure 2.1, if we want to know the probability of it being *Cloudy* given observations in the rest of the model, we would have to perform inference as this posterior probability is not known to us. Since the joint probability is represented by the model, we can infer any specific probability, but this process is not always computationally feasible and we often have to settle with approximations. The probability of being *Cloudy* is thus:

$$p(cloudy|rain, cold, swim) = \frac{p(cl, ra, co, sw)}{p(ra, co, sw)} = \frac{p(cl, ra, co, sw)}{\sum_{cl'} p(cl', ra, co, sw)} \tag{2.12}$$

However, the computation of this equation is not very tractable. There are ways to overcome the complexity by using conditional independence assumptions from the structure of the Bayes Net, as we will see later in the discussion of Naive Bayes Nets. However, for most Bayes Nets, exact inference is intractable even with some conditional independence assumption. This is why inference in Bayes Nets is commonly done with approximation algorithms. There are many flavours of approximation algorithms for Bayes Nets. In the sampling variety, there is the Monte-Carlo method, where random samples are drawn from the distribution. There are the variational methods, one of which is the mean-field approximation that exploits large numbers. And finally, there is the loopy belief propagation that leverages Pearl's algorithm.

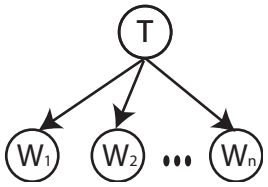


Figure 2.2: Simple Naive Bayes Net

2.4.4 Learning Bayes Nets

Bayes Nets are often much more complex than the example shown here. This is why considerable effort has been done within the research community regarding learning Bayes Nets. There are two major considerations when learning Bayes Nets. The first is whether we have to learn the structure, the parameters, or both. The second consideration concerns whether the model is fully observable or partially observable. Partially observable networks occur when some nodes are hidden or cannot be directly observed. This is common in classification tasks with unlabelled data where the class or type is not labelled. However, in our research, we have used mostly Naive Bayes and we will discuss learning parameters in the case of Naive Bayes.

2.4.5 Naive Bayes Nets

A Naive Bayes Net is a very simple structure with only one parent node and many leaves that are connected only with the parent node (Figure 2.2). In this model and in the context of Words Sense Disambiguation (or citation classification), every citation context c_i(sentence that contains the citation) contains a set of words $w_n \in V =< w_1, w_2...w_{|V|} >$ from vocabulary V, and is generated by a probability distribution over a set of parameters θ. Types of citations are represented by mixture components $t_j \in T = \{t_1, t_2...t_{|T|}\}$. It is assumed that when a citation context c_i is present, it is generated by first choosing a mixture component t_j with probability $P(t_j|\theta)$ and then the mixture component chooses the context with probability $P(c_i|t_j)$. This model makes the assumption that words w_n in a citation context c_i are all independent from each other, given a citation type t_j. This assumption is known as the Naive Bayes Assumption:

$$P(c_i|t_j) = P(w_1, w_2...w_n|t_j) = \prod_{l=1}^{n} P(w_l|t_j) \tag{2.13}$$

This assumption makes inference in a Naive Bayes Net very simple. Suppose we have a context c_i and we want to find the type t_j of the citation where it appears, i.e., we want to find $P(t_j|c_i;\theta)$. We first use Bayes Rule and then we make the Naive Bayes assumption as follows:

$$P(t_j|c_i;\theta) = P(t_j|w_1, w_2...w_n;\theta) = \frac{P(t_j|\theta)P(w_1, w_2...w_n|t_j;\theta)}{P(w_1, w_2...w_n|\theta)}$$

$$= \frac{P(t_j|\theta)\prod_{l=1}^{n} P(w_l|t_j;\theta)}{P(w_1, w_2...w_n|\theta)} \tag{2.14}$$

Therefore, Equation 2.14 allows us to obtain the probability of a given citation type to produce the citation context we observed. If we want only to classify the citation and we don't need the entire probability distribution for all classes, we can discard the denominator and take the log of Equation 2.14, resulting in: $\arg\max\left[\log P(t_j|\theta) + \sum \log P(w_l|t_j;\theta)\right]$.

Learning parameters

We saw that using the Naive Bayes classifier is very simple. However, we need to first learn its parameters. To do this, we need to estimate the maximum likelihood ($\arg\max_\theta P(D|\theta)$) or the maximum a posteriori ($\arg\max_\theta P(\theta|D) \propto \arg\max_\theta P(D|\theta)P(\theta)$). The only difference between the two, is that the latter uses a prior represented by a Dirichlet, which makes sure that there are no probabilities equal to zero for words that appear rarely. Thus we have Equations 2.15 for maximum likelihood and Equations 2.16 for maximum a posteriori. We use both learning models in our experiments.

$$\begin{aligned} P(w_l|t_j;\theta) &= \frac{\#(w_l, t_j)}{\#(t_j)} \\ P(t_j;\theta) &= \frac{\#(t_j)}{|C|} \end{aligned} \tag{2.15}$$

$$\begin{aligned} P(w_l|t_j;\theta) &= \frac{1 + \#(w_l, t_j)}{|V| + \#(t_j)} \\ P(t_j;\theta) &= \frac{1 + \#(t_j)}{|T| + |C|} \end{aligned} \tag{2.16}$$

where $\#(w_l, t_j)$ is the number of times the word w_l appears in the contexts labeled t_j in our training sample set, $\#(t_j)$ is the number of times the citation type t_j appears in our training set, $|T|$ is the number of distinct citation types, $|C|$ is the size of our training sample (number of labeled contexts), and finally, $|V|$ is our vocabulary size (the number of attributes/words/features we have).

Chapter 3

Literature Review

In his review of citation studies, White [50] surveys the three main streams of citation research: citer motivations, content analyses of citation contexts, and classifying citations.

Citer motivation studies are concerned with finding out why authors make citations. This is a very similar area of research to *classifying citations* (discussed later), but it is more of a sociological study for the reason that authors cite works and are generally not concerned with trying to classify existing citations into one or more predefined categories. Garfield [20] was one of the first researchers in the area and proposed a list of reasons for citations A.1. Of course, this scheme is very similar to other schemes for citation classification (to be discussed later), but the significant distinction lies in who interprets the citation functions. While, with citation classification, a person other than the author is tasked to find out the *implicit* reason of the citation only from the text, in citer motivation studies, the authors themselves are surveyed.

Content analyses of citation contexts (or context analysis) is concerned with using repeating phrases in the citation contexts[1] of a citation to the same paper as "descriptions" or "symbols" of the cited work. Both context analysis and citation classification work with the citation context of a citation. However, White argues that context analysis is a more promising approach as it relies on explicit information contained in the citation contexts, while citation classification tries to determine implicit information.

An example of research on context analysis is given by Nanba and Okumura [40]. Their work is concerned with automatic multi-paper summarization (generating a survey). Although the first step of their work is citation classification, their main goal is to use citation contexts as descriptions of the cited works. The key idea here is that citation contexts provide brief summaries of the referenced paper, and by finding multiple citation contexts describing the same paper, the paper can

[1] A *citation context* is the sentence containing the citation and optionally the sentences surrounding it.

be summarized automatically from the citation contexts. A similar approach is taken by Nakov et al. [38]. They suggest using the citation contexts as data from which to build "semantic interpretation models" and they provide some early work on normalizing (paraphrasing) citation contexts of the same type, so they can be used in further applications. This work again assumes the ability to cluster citation contexts of the same meaning together (which essentially is classifying the citations).

Classifying citations (or *Citation Classification*) is concerned with identifying the nature of the connection between the citing and cited papers. It is by far the most difficult of the three streams of citation research and Small goes as far as saying that "their application cannot be delegated to a computer even in principle". The reason for this statement is that citation classification is often implicit in the citation context. However, even though White claims that context analysis is a more promising field, as we saw in the examples above, many context analysis applications require some sort of citation classification (or at least clustering of common citations).

The fact that citation classification is a prerequisite for many applications motivates our research on this problem. In the next two sections, we review some work on citation schemes and automatic citation classification. For information on the other streams of research, refer to White [50].

3.1 Citation Schemes

The purpose of a citation scheme is to identify all the reasons why a citation has been made. The first citation schemes came mainly from Citer Motivation studies, where the researchers were curious as to the reasons that authors cite. When citation indexing became more established, new schemes were developed with the idea of classifying citations in order to improve document indexing. In this section we review in chronological order several well-known classification schemes.

3.1.1 Garfield

The earliest citation scheme (see Appendix A.1) was proposed by Garfield [20]. There he lists his thoughts on the reasons why authors cite other works. He only mentions the reasons in passing without making any in-depth studies of their frequencies or how they align with citations in papers from different fields. Nevertheless, his classification scheme should be mentioned as he was one of the pioneers in the citation-indexing field. It should be noted that Weinstock [48] used Garfield's classification without citing him and other works have mistakenly attributed Weinstock for Garfield's classification scheme (as in [39], [43], [46], and others).

Garfield lists fifteen reasons for citations (see Appendix A.1). The citation functions are quite diverse and are general enough that they can apply to many

fields. However, we have a problem with the scheme because it does not make a few distinctions that we believe are particularly important. For example, there is no distinction between *general* and *specific background* and no function for *usage* of data or other materials. As well, there are categories that we feel are overlapping such as "Pay homage to pioneers" and "Identifying original publications in which an idea or concept was discussed". That said, it is still a novel scheme that laid the foundation for the whole field of citation classification.

3.1.2 Moravcsik and Murugesan

As far as we can tell, Moravcsik and Murugesan's [37] scheme (See Appendix A.3) is the first citation classification scheme that was developed keeping in mind that citations can have more than one function. The scheme contains four main categories and each citation can fall into more than one category. The scheme was developed from 30 articles randomly selected from the *Physical Review* spanning five years from 1968 to 1972, and containing 702 citations. The purpose of the study was to question the validity of citation studies that used simple citation counts. The results of the study are somewhat controversial in the citation classification community as the authors found that 14% of the citations they analyzed fell into the negative category. According to White [50], this number of negative citations is the largest ever found. Because of this, Moravcsik and Murugesan are often cited by critics of current bibliometric practices (that do not include the type of citations) to determine the influence of a paper in the research community. Nevertheless, the fact that their classification scheme was developed with the idea that citations can be part of more than one function is novel, and is a concept that we also wanted to use in our scheme.

3.1.3 Chubin and Moitra

Cubin and Moitra's [15] scheme (see Appendix A.4) is a slight revision of Moravcsik's scheme. They made Moravcsik's categories mutually exclusive and dropped the "Evolutionary/Juxtapositional" dimension. This move was made in an attempt to further generalize the scheme and apply it to more than one speciality in more than one journal. In contrast to Moravcsik and Murugesan, Chubin and Moitra applied the scheme to experimental and theoretical high energy physics from a sample of four journals. Their results are quite interesting. First and foremost, they found that only about 5% of all references are negative, and furthermore, they were only partially negative. Also, they discovered that the distribution across the "affirmational" type categories varied from source to source. Therefore, the effectiveness of citation schemes depend heavily on the specialty and source.

3.1.4 Spiegel-Rosing

Ina Spiegel-Rosing [45] analyzed citations from several *Science Studies* volumes. She analyzed 66 articles in total, ranging across multiple disciplines. Her classification scheme (See Appendix A.5) contains 13 categories, of which "substantiating a statement or an assumption made or pointing to further information" (category 8) is the most prevalent of all, spanning 80% of all citations reviewed. What is interesting about this scheme is that, as Teufel [46] pointed out, "more than one category can apply to a citation; for instance positive and negative evaluation (category 9 and 10) can be cross-classified with other categories".

3.1.5 Oppenheim and Renn

Oppenheim and Renn's [42] did something different. They wanted to find out why old papers were still being cited in physics/chemistry papers. They came up with a classification scheme (See Appendix A.6) of seven main reasons and their findings may help us deal with older citations.

3.1.6 Finney

We include here Finney's [17] scheme (See Appendix A.7) only for completeness. We were not able to obtain a copy of Finney's Master's thesis, but we felt that we should cite it because we base our classification scheme on Garzone's [23], which borrows heavily from Finney. According to Garzone, Finney's scheme "is the most comprehensive scheme designed such that the assignment of categories is 'capable of being automated'."

3.1.7 Garzone

Garzone ([23], [22]) reviewed most of the classification schemes discussed here, and more. On the basis of this investigation, he concluded that Finney's [17] scheme was the most complete and most fitting for his work on classifying citations in articles from the fields of Physics and Biochemistry. However, he states that Finney's scheme has limitations in that "many of [her] categories are too broad in that each encapsulates other more finely discriminating functions" and that "Finney's citation classification does not cover some functions of citations at all". This prompted Garzone to extend Finney's schemes by breaking up some of her categories and by borrowing from other works in citation classification. The end result was a citation classification scheme (See Appendix A.8) that was the most complete and fine-grained of all with 35 categories in total, and the one that he used in his experiments on automatic citation classification.

3.1.8 Nanba and Okumura; Pham and Hoffmann

Nanba and Okumura [40] completely stripped Garfield's citation scheme (while at the same time mistakenly citing Weinstock as the originator of the scheme) and came up with a simple classification composed of only three categories (See Appendix A.9). This is an example of a scheme that is highly specialized in that the only purpose for it is to classify citation contexts for use in a larger automatic summarization system.

Pham and Hoffman [43] also borrow from Garfield (while citing Weinstock) and strip down his scheme to the four "most relevant" categories as applied to their experimental corpus.

3.1.9 Teufel et al.

Teufel et al.'s [46] scheme (See Appendix A.11) is an adaptation of Spiegel-Rosing's scheme discussed above. It contains 12 mutually exclusive categories that have top-level classifications, such as: Neutral, Weakness, Contrast, and Positive. Their classification scheme was developed as a result of analysis of a corpus of computational linguistic papers. They annotated 26 articles, containing 548 citations. More than half (65%) of all classified citations fell under their Neutral category, 15.7% fell under the Usage of Algorithm category, and the rest had significantly lower frequencies.

3.2 Automatic Citation Classification

Considering how many citation classification schemes have been developed over the years, there has been considerably less research done on classifying citations automatically. Some of this reluctance to pursue automatic citation classification could be due to the realization that the function of a citation is often only implicit in an article, or that extratextual information (i.e., information outside the citing article) may sometimes be needed to properly classify the citations [50].

Nevertheless, there have been some promising results in the area of Automatic Citation Classification. In this section, we discuss representative work in this area.

3.2.1 Garzone [23]

Garzone built a rule-based automatic citation classifier. As we discussed in the previous section, Garzone used his own modified scheme for citation classification containing 35 categories in the development and testing of his classifier. His classifier used two types of rules: semantic-grammar parsing rules and template-matching rules.

The rules were developed by analyzing a design set of fourteen articles (eight physics articles and six biochemistry articles). Every article was read carefully by Garzone, and for every citation, cue-word phrases which helped in classifying the citation were extracted. After this initial phase, more-generalized matching rules (which Garzone calls parsing rules) were formed from the previously extracted cue-word phrases. And finally, the cue-word phrases that did not generalize into a parsing rule, were converted into template-matching rules.

To better understand the forms of the different rules, and how they are used to classify citations, consider the following example:

Sample parsing rule:

```
<par-1> := [not=20N]<usage-verb-1><head-modifier><head-1>
<usage-verb-1> := use | using | uses | introduce | ...
<head-modifier> := any part of speech which modifies the head noun
<head-1> := <equipment-head=18> | <equation-head=19> | ...
<equipment-head> := apparatus | applications | arrays | ...
<equation-head> := algorithm | components | ...
```

Sample template-matching rule:

```
(Category 34/Results Cue Words) := further details of | more on
```

Citation contexts to label

1. "In [2], John uses a special algorithm to arrive at the same results"
2. "More on this research can be found in [1]"

Example 1: Rule matching in Garzone's Classifier

In Example 1, one can observe a parsing rule *par-1* and a sample of a template-matching rule that matches Garzone's classification category 34. The parsing rule deals with Garzone's *usage* categories. For every citation context (in Garzone's case, the citation contexts are the sentences where the citation appears), Garzone's classifier tries to match the context to one of his parsing rules. If this fails, it tries to then match with a template-matching rule, and if that fails, it defaults to the default category of the section in the article to which the citation belongs. The rules are also section-specific, so there could be different citation assignments for the same parsing rule for citations in different sections.

Returning to the example, consider the two citation contexts from Example 1 above. When Garzone's classifier is given the first sentence as input, it tries to match it with one of his parsing rules. In this case, the sentence would be matched by parsing rule *par-1* because it contains the *usage-verb-1* "uses", a head modifier "special", and the *head-1* "algorithm". In this parsing rule, the head noun is actually the one that matches a citation category, which can be seen by

the statement "equation-head=19" and the word "algorithm" from the example are part of *equation-head*.

For the second example sentence, there is no parsing rule that matches, but it is matched by the template-matching rule for category 34 because it contains the cue phrase "more on".

This is basically how Garzone's rule-based classifier works. To test his classifier, Garzone performed two sets of tests. The first test was performed on the design set of articles (eight physics and six biochemistry). For the citations from the physics articles, the classifier was correct for 78% of the citations, partially correct for 11% (for citations that have more than one category, at least one was classified correct), and wrong for 11%. When tested on the set of biochemistry articles, the classifier performed better by classifying 84% correctly, 8% partially correctly, and 8% wrong.

For the second set of tests, Garzone used six previously unseen articles (three from physics and three from biochemistry). Again the classifier did better on the biochemistry papers by classifying 61% correct, 12% partially correct, and 27% wrong, while the physics articles were classified as 41%, 21%, and 38% respectively. Garzone's reasoning for the accuracy discrepancy between the two disciplines was the observation that the biochemistry articles were better structured in well-defined sections.

Although the results of Garzone's classifier were somewhat satisfactory, there are several problems with this approach. First, this approach does not generalize well for unseen data. This is evident from the test results for the previously unseen citations, and from the finding that results were not significantly better with defaulting the citation classes rather than classifying them with the parsing rules. Garzone's numbers for defaulting the citations are 21%, 30%, and 49% for the physics articles, and 57%, 6%, and 37% for the biochemistry articles.

Secondly, designing the parsing rules is very time-consuming and requires expert human knowledge to construct the rules for classifying. The citations have to first be manually annotated, then cue words have to be extracted identifying each citation type. Lastly, these cue words have to be generalized to form parsing rules, which requires a considerable amount of expertise and time. This whole process is prone to human errors and requires subjective reasoning to first determine the appropriate cue words.

Thirdly, we believe that adding new rules would be problematic. To add a new rule, all current rules would have to be examined to see if the new rule will conflict with them. Also, if it does conflict somehow, the previous rules will have to be repaired, which could pose a whole new set of problems.

3.2.2 Nanba and Okumura [40]

Nanba and Okumura use citation classification in their method for automatic document summarization. As we previously mentioned, they use a simplistic classification scheme containing three citation classes: Type B, Type C, and Type O (see Appendix A.9 for description of the citation types). Their approach to citation classification is similar to that of Garzone's as they use a rule-based approach with extracted cue words. Briefly, their method of classifying citations is as follows:

- Extract the citation context with cue words.
- If a cue word of Type C is found in the citation context, label the citation as Type C.
- If a cue word of Type B is found in the citation context, label the citation as Type B.
- Otherwise, label the citation with Type O.

Extraction of citation context

Extraction of citation context is also performed with cue words. The algorithm starts with the citation context being just the citation sentence. Then, each additional sentence around (before and after) the current citation context is searched for previously extracted cue words that identify a citation context. If the sentences include a cue word, they are added to the citation context and the process repeats. Otherwise, if no context cue words are found, or the citation context already spans a paragraph, the process ends and the citation context is returned.

Cue words for determining citation contexts were extracted from a corpus of previously extracted 100 citation contexts. The process of extracting those cue words (as described in [40]) are:

1. Create the reference area corpus by hand.
2. Apply n-word gram analysis of the corpus.
3. Select 86 cue words manually, by checking the list of frequently used expressions made in step 2.

The authors identified 86 cue phrases in total from six different types: anaphora (e.g., *For this*), negative expression (e.g., *but*), 1^{st} person pronoun (e.g., *I*, *we*), 3^{rd} person pronoun (e.g., *they*, *their*), adverb (e.g., *furthermore*), and other (e.g., *drawback*). Testing of their citation context extraction method on 50 previously unseen citation contexts revealed 79.6% Recall and 76.3% Precision.

Cue words for classification

Cue phrases that determine the type of a citation are extracted (as described in [40]) by the following procedure:

1. Collect sentences for types B and C from corresponding sections.
2. Calculate n-word gram separately.
3. Apply cost criteria, which tends to extract longer expressions, to the result of n-word gram statistics.
4. Select 76 cue words for type C and 84 for type B manually, by checking the list of frequently used expressions made in step 3.

Examples of type C cue words are "Although", "however", "but the", and some examples of Type B cue words are "based mainly on", "the basic", "used by", and so on.

To extract the necessary cue words, Nanba and Okumura used 282 citation contexts with manually annotated citation types. They came up with 160 total cue words, of which 76 identified type C citations, and 84 identified type B. They tested classification on both the development set of citation contexts and on 100 previously unseen citation contexts. For the design set, they reported accuracy of 90.1% and for the test set, accuracy of 83%. However, they don't provide results for the whole process: extraction of citation contexts and classification.

Problems with this approach

Similar to Garzone's work, Nanba and Okumura's citation classification method requires defining a set of cue words that uniquely identify the type of citation. This process is time-consuming and requires expert linguistic knowledge. However, insertion of new cue words is easier with this method because of the simplicity of the classification scheme.

Another major flaw with this approach is that a single cue word identifies a type of citation. While this can work with very small classification schemes like the one Nanba and Okumura used, they would not work with Garzone's scheme.

3.2.3 Pham and Hoffmann [43]

Pham and Hoffmann developed a rule-based knowledge acquisition system that they applied to citation classification. Their system is based on Single Classification Ripple Down Rules, which is a model for incremental knowledge acquisition.

Model for classification: Ripple Down Rules

The model for Ripple Down Rules is similar to that of decision trees, where each node has a rule (or a set of rules), and has none, one, or two children. There are exactly two types of node links: an 'else' link (for the case when the condition of the node does not meet) and an 'except' link (when the condition of the node does meet, but as well the conditions of the leaf connected with 'except' meet) (See Example 2 for a sample Ripple Down Rule tree for citation classification).

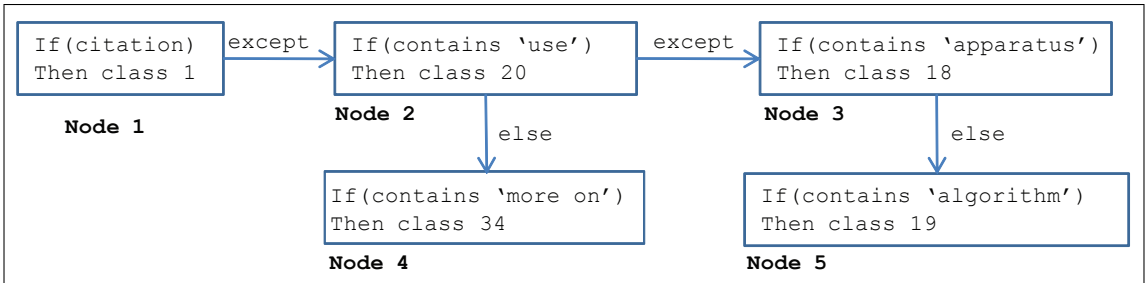
The beauty of an incremental knowledge-based (KB) system is that there is no need for a knowledge engineer (like the one needed to extract cue words for classification in the classification methodologies of Garzone and Nanba above). Instead, only a domain expert is needed and the knowledge acquisition process is as follows:

- When the system is clean (i.e., there is no knowledge, and therefore, no nodes with rules), there is only one default node that matches everything (i.e., there are no rules).
- Samples for classification are evaluated as they are passed from node to node, starting from the root node.
- When the system is presented with a sample that mismatches at a particular node, a new node is added as a child of the node that mismatches (called the *cornerstone node*).
- If the *cornerstone node* does not have an 'except' link, the new node is added with an 'except' link. Otherwise, it is added with an 'else' link.
- Then the domain expert formulates a rule for the new node that is satisfied by it, but is not satisfied by the cornerstone node.

To better visualize how Ripple Down Rules (RDR) trees operate, consider Example 2, where the system with the given RDR tree in the example has to classify the same citation contexts given in Example 1. Given the first citation context, the system starts at *node 1*, determines that the context contains a citation and then proceeds to *node 2*. The citation context also satisfies *node 2* as it contains the verb 'uses', so the process continues to *node 3*. *Node 3* does not match, but there is an 'else' child (*node 5*) that matches with the word "algorithm" and, as this is the last node, the citation is annotated as category 19. If *node 5* did not match, the last matching RDR node would have taken over, which was at *node 2*.

The second example matches in a similar fashion at *Node 4* and is annotated as citation category 34.

Now, let us consider the third citation context in the example. Let us say that this citation context is not category 34 as the one right above it, but is category 33 because of the cue word "only". The system misclassifies the citation as 34,



Citation contexts to label

1. "In [2], John uses a special algorithm to arrive at the same results"
2. "More on this research can be found in [1]"
3. "More on this research can only be found in [1]"

Example 2: An example of Ripple Down Rules tree

but as the system is under supervision, the domain expert tells the system that it was misclassified by providing the correct classification (35) and the cue word that differentiates it with the *cornerstone node*. This in turn will add another node as a child on *node 4* with an 'except' link.

Citation Classification with RDRs

As we saw in the previous section, Pham and Hoffmann use a simplified citation classification scheme with four categories. In their classification system, they build one RDR tree for each citation class. As the annotator builds up the knowledge base, the system checks for consistencies, and if a new rule causes another previously matching rule not to match, then the system will prompt the domain expert to revise the rule until all rules in the tree are satisfiable. Also, when adding cue words, they provide the domain expert with synsets[2] from WordNet for the current rule, so the domain expert can quickly augment the rules by using a whole synset instead of a single word.

Testing and limitations

The knowledge system was built incrementally from a set of 482 citation contexts, resulting in 70 nodes for their *basis* citation category, 24 for *support*, 23 for *limitation*, and 54 for *comparison*. They compared their results against Nanba and Okumura's, and found that their system outperforms them. The accuracy was 94.0% for *basis*, 97.3% for *support*, 95.3% for *limitation*, and 96.7% for *comparison*. The tests were performed on a set of 150 unseen citation contexts, but their results

[2] A *synset* is a set of near-synonyms, e.g., (*look*, *gaze*, *stare*).

cannot be considered accurate as they were still performing incremental knowledge acquisition while performing the test as evidenced by their statement that more nodes were added as a result of the tests.

We see several problems with this approach. First, the system has to be supervised and sometimes the domain expert will have to make complex decisions as to how two classes differentiate in the case of a fine-grained citation scheme such as Garzone's. Also, the fact that the domain expert would not be allowed to enter a rule if there were inconsistencies only amplifies this problem.

3.2.4 Teufel et al [47]

Teufel et al. are the only researchers to date that we know have used machine learning methods for citation classification. In their experiments, they use a mutually exclusive classification scheme (as previously described, see Appendix A.11), containing 12 categories.

Classification algorithm

The machine learning algorithm used in their experiments is the IBk algorithm (with k=2), a variation of the k-nearest neighbours algorithm. This algorithm is fairly simple. Given a new, unclassified sample, Y, the idea is to find the k closest samples (X) from the training set to X, given a similarity function $sim(X, Y)$.

Features

The features used with the IBk algorithm for classification in Teufel et al.'s experiments are impressive in their magnitude. They employ a set of 1782 different cue phrases developed from 80 different articles, a feature set modelled by 185 patterns denoting two different agent types (the authors, and the rest), 20 manually acquired verb clusters, 12 additional features that record the presence of 892 additional cue phrases identified by the annotators, as well as verb tense, modality, the location of the citation context in the article, and lastly, an indicator as to whether the citation points to the author's own or other work.

Results and discussion

Design and tests were performed on 360 conference articles. Testing was performed on 116 articles containing 2829 citations and the accuracy for each category is as follows:

Our criticism with Teufel et al. concerns their method of annotating citations. First, they mark only explicitly signalled citation functions. That is, citations would not be annotated and thus present in the training and testing sets unless

Weak	CoCoGM	CoCoR0	CoCo-	CoCoXY	PBas
.78	.81	.77	.56	.72	.76
PUse	PModi	PMot	PSim	PSup	Neut
.66	.60	.75	.68	.83	.80

Table 3.1: Teufel et al's accuracy results of citation classification

they have explicit textual cues in the citation context that points to the citation function. As Teufel et al. say: "Our guidelines explicitly state that a general linguistic phrase such as 'better' or 'used by us' must be present; this increases the objectivity of defining citation function." Not only does this increase the objectivity of the citation function, but it also skews their results by making it more difficult to classify citations that are not annotated. This would be equivalent to Garzone only labelling citations with the cue phrases he found, so that in turn he would get much better results, especially for the unseen data.

Chapter 4

Contributions of the Thesis

In this chapter, we present our contributions to the field of citation classification. First, we present our first experiment with citation classification to motivate the need for more research in the field. Next, we define a new classification scheme that is easier to annotate with. Then, we describe our Web-based annotation tool that reads PDF files, which any user may access to annotate citations. Lastly, we describe the development of a manually classified corpus, annotated with our own annotation tool.

4.1 Citation Classification Experiment

In our first experiment with Automatic Citation Classification, we framed the problem of classifying citations as a word sense disambiguation (WSD) task. We did this because we felt that the disambiguation of a word is closely related to the 'disambiguation' of a citation. Both a word and a citation may have multiple meanings, and early WSD approaches (rule-based) are very similar to early Automatic Citation Classification approaches (again, rule-based). It was only natural that we try probabilistic classifiers for citation classification as they are now very popular for WSD tasks.

For our experiment, we decided to use Garzone's (Appendix A.8) scheme as we felt that it was the most complete scheme and it was designed initially based on, and intended to work with, biochemistry articles, which we used for the experiment.

4.1.1 Probabilistic Model

We chose Naive Bayes as our probability model despite there being many classifier structures (General Bayes Net (GBN), Tree Augmented Bayes Net (TAN), BN Augmented Bayes Net, etc.) that have been shown to work better than Naive Bayes [[14], [19], [32]]. Our motivation for choosing Naive Bayes Nets is because

they have been shown to work well for text categorization [31] and also because of their simplicity. Naive Bayes Classifiers are very fast at classifying, and are also fast at learning because no structure has to be learned. Additionally, we performed a few simple experiments on our data sets with Naive Bayes, TAN, and GBN and found that TAN and GBN reduced to Naive Bayes anyway.

We have previously discussed this model in Section 2.4.5. However, one of the main drawbacks of using this method is that it needs a vast amount of annotated learning examples, which we did not have. Therefore, we adapted a semi-supervised boosting approach for our experiment by first learning from a small set of labelled data and then expanding this set with the EM algorithm discussed in the following section.

4.1.2 Adding Unlabelled Data

As discussed previously, labelled data is difficult to find so one solution therefore is to train our classifier with both labelled and unlabelled data. One way of combining labelled and unlabelled data for word sense disambiguation is presented by Yarowsky [52]. We argued earlier that word sense disambiguation and citation classification are very similar so that the techniques used for one of these tasks can be applied to the other. However, this is where they differ: we cannot use Yarowsky's algorithm for citation classification! This is because he makes the following key observations that allow him to use unlabelled data: 1) one sense per collocation; and 2) one sense per discourse. Although the first observation is valid for citations, the second, more powerful observation is not! In the case of citation classification, the goal is to disambiguate only one "word": a citation. This "word" may have many meanings (types) in a given document.

Another algorithm for combining labelled and unlabelled data is *co-training* and is presented by Blum and Mitchel [11]. Although very popular, we cannot as yet use this algorithm for citation classification. The key idea behind this approach is to use two classifiers that use two different 'kinds' of information to classify a document (e.g., text on a webpage and captions of links pointing to the webpage). Unfortunately, there are not two kinds of information in citation contexts—the only information is the words that are collocated with the citation. Of course, there are other kinds of information that can be used (e.g., verb tense, location of citation in the document); however, these pieces of information only 'reduce' the number of citation types and cannot pinpoint a single citation type, even weakly (e.g., citations that are related to future research would not appear in the background section of a paper, but many other types of citations can appear in the background).

Since we cannot use the algorithms mentioned above, we use the EM algorithm as applied to text classification and presented by Nigam et al. [41]. The basic steps of the algorithm are as follows:

- Data is split into a set of labelled citation contexts C^l and a set of unlabelled citation contexts C^u.

- Build initial classifier only with C^l using equations 2.15 for Maximum Likelihood Estimation or equations 2.16 for Maximum a Posteriori estimate.
- Iterate the following until there is no significant change in log likelihood (see equation 4.1 below; for ML use only the second part of the equation).
  - (E-step) Use the current classifier to classify unlabelled citation contexts from C^u.
  - (M-step) Re-estimate the classifier parameters using equations 2.15 for Maximum Likelihood Estimation or equations 2.16 for Maximum a Posteriori estimate.

The log likelihood measures how well the data fits our model and is expressed as:

$$LL(\theta|D) = \log\left(\prod_{t_j \in T} P(t_j|\theta) \prod_{w_l \in V} P(w_l|t_j;\theta)\right) + \sum\sum \log\left(P(t_j|\theta)P(c_i|t_j;\theta)\right) \tag{4.1}$$

The next section describes our experiments with the probabilistic classifier just described. We have also implemented an 'enhancement' to the model as described in [41]; however, we will delay this discussion for now.

4.1.3 Experiments and Discussion

Here we describe the experiments we have performed with our citation classifier and the results we have obtained. All experiments were performed on a corpus of 9462 citations collected from 900 biomedical scientific texts[1]. We classified 177 citations that spanned 10 categories from Garzone's scheme which we then used for training and testing.

Methodology

We implemented our citation classifier under WEKA [51]. We did this because WEKA has many useful tools and provides useful statistics of the performance of classifiers. We also used scripts from the Duluth [2] system that participated in Senseval3. These scripts were helpful in tokenizing our citations and converting them into .arff files that are readable by WEKA. In WEKA, citation contexts are represented by vectors of terms (words).

The whole process is as follows: we first split the citations into training and test sets; then, we ran the citations used for training through an n-gram statistics

[1]We thank Prof. Chrysanne DiMarco and her research group for the segmented articles.

package (from Duluth [24]) while stripping any unnecessary stop words. This script output a file with Perl regular expressions that was used by another script to process the training and test data separately, outputting .arff files for each set. The reason for this separation is that we collect the attributes (words) that the classifier is aware of only from the training set. Words that do not appear in the training set but appear in the test set are discarded.

Initial Experiments

During our initial tests, we observed that the classifier performed worse when it was trained with supervised and unsupervised (from now on, we will call this unsupervised) training compared to when it was trained only with supervised training. We initially thought that these observations were a product of the attributes and size of the training set so we subsequently ran experiments on various sizes and types of training and test sets. We experimented by splitting the set of 177 labelled citations in two different ways: (3/4 training + 1/4 test) and (1/2 training and 1/2 test). Each of the splits was performed with WEKA's StratifiedRemoveFolds filter to ensure that the training and test sets would have similar distribution of citation types. We also experimented with representing citations with different size of terms: unigrams and bigrams. Finally, we performed several dimensionality reduction techniques to see what effect they would have on citation classification. The results are summarized in Figure 4.1. Unfortunately the results are not at all very encouraging. For all test sets, unsupervised training performs worse than supervised training. This is alarming! Not only do we have no use of the unlabelled data, but the performance of the classifier degrades, on both the training and test data. A possible explanation is that the natural clustering of the unlabelled data does not correspond one-to-one to citation types so that the longer the algorithm runs, the lower the accuracy of the classifier becomes. It can be observed that this is exactly the case in Figure 4.2 (we present only one data set, but the same phenomenon is evident in all training sets).

Although these results are disappointing, Nigam [41] found that certain data sets exhibit the same behaviour. He proposes several improvements to the EM algorithm. One improvement consists of weighting the data in the unlabelled set. The rationale behind this is that, since natural clusters in the unlabelled data do not correspond to classification labels and the labelled data is several orders of magnitude larger than the labelled data, we should reduce the effect of the unlabelled data. We implemented this feature and even went one step further. We introduced flexible weight for every sample of data in the unlabelled set. This variable weight is proportional to the confidence with which the classifier labelled the unlabelled sample during the E-step. The results of these 'improvements' (on the Cross Validation data sets) are shown in Figure 4.3 (we show only results for unigrams, as results for bigrams are similar). Clearly, it is evident from the diagram that the weighting of unlabelled data did not have a significant impact on unsupervised learning.

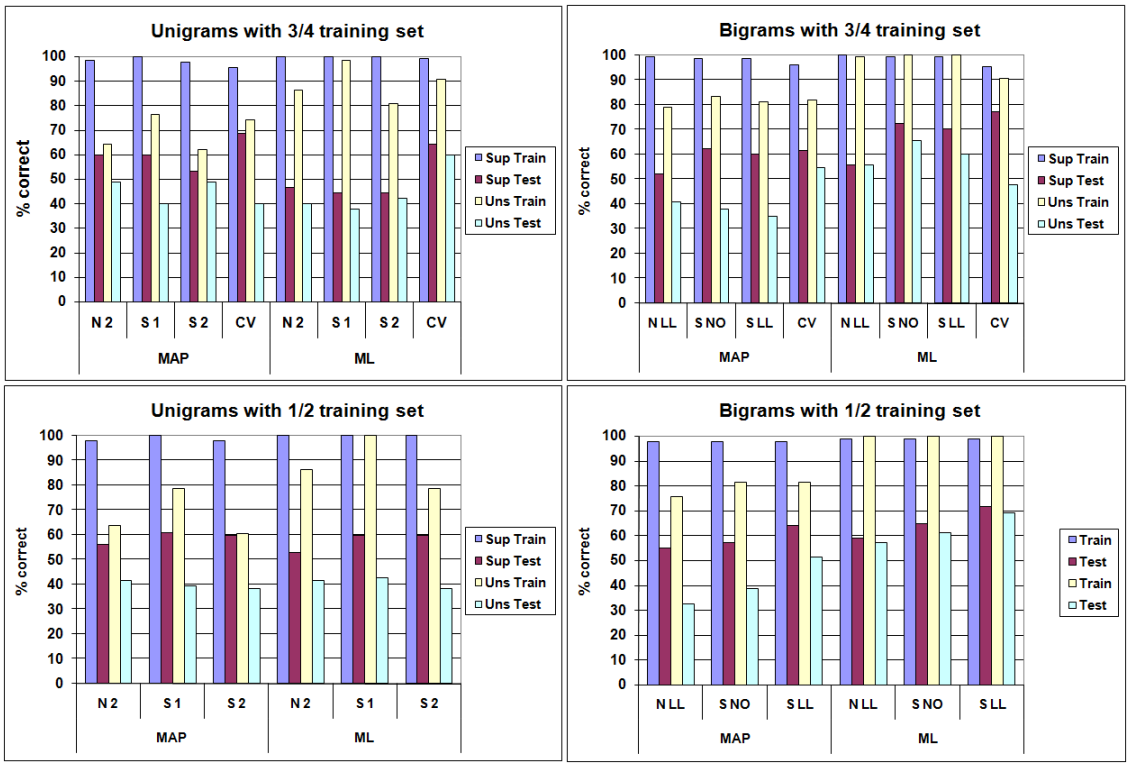


Figure 4.1: Supervised and Unsupervised accuracy results of different test sets using MAP and ML. Legend: "N 2"—Stop words NOT removed, words that appear twice or more are kept (Frequency ≥ 2); "S 1"–Stop words removed, Frequency ≥ 1, "S 2"–Stop words removed, Frequency ≥ 2; "CV"–dimensionality reduction was perfomed via leave-one-out cross validation using χ^2(chi-square). The legend is similar for bigrams: Frequency was always 1, but dimensionality reduction was sometimes performed by log likelihood ("LL")—measures how likely the components of the bigrams are to appear collocated; "NO" means no "LL" performed.

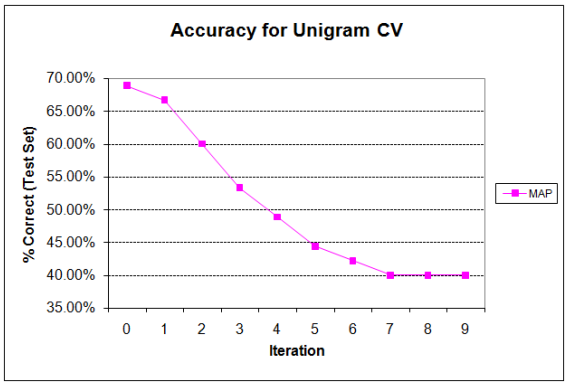


Figure 4.2: Accuracy with each EM iteration. (Iteration 0 is supervised accuracy

We also experimented with soft labelling (probabilistically choosing a class based on the probability distribution for the current unlabeled sample), as opposed to hard labelling (choosing the class with maximum probability), but we do not present the results as soft and hard labelling have almost identical performance.

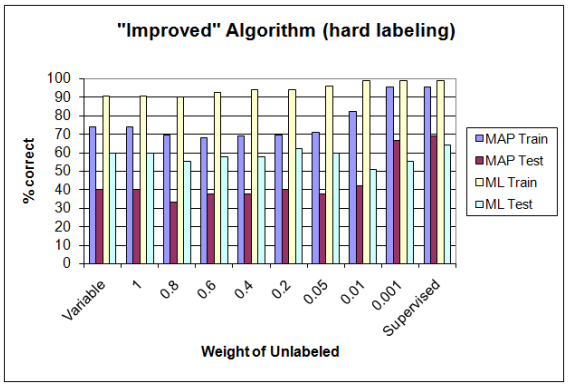


Figure 4.3: Varying weight of unlabeled data. Note: weight=1 is equivalent to regular unsupervised learning; weight=0 is equivalent to supervised learning.

Discussion of the results

At this point, the results do not favour the use of probabilistic network classifiers for citation classification. However, we believe that many of the problems are coming from the labelled data and not the algorithms themselves. We need to perform more tests to determine if unsupervised learning can be performed in the context of citations. First and foremost, we need to augment the labelled examples to cover every possible citation type. Currently only a small subset of citation types was present in the labelled data (10/34 types, using the classification scheme of Garzone [22]). This obviously creates a problem because the algorithm is forced to cluster together citations of different types and possibly different natural clustering.

Another concern with the labelled data is that the distribution of types is not uniform across the labelled samples. For example type #17 appears 38/177 times, type #20 appears 29/177 times, whereas type #32 appears 12 times, and types #16 and 29 appear less than eight times. This creates a problem because the prior probabilities of citation types #17 and #20 will be higher than types #16 and #29, and this might not be the case in reality. Therefore, more unknown citations will be labelled with the types that have higher priors and this in turn will make these priors increase in the subsequent M-Step of the algorithm. For example, $P(t_j = \#17|\theta) = 0.21$ and $P(t_j = \#32|\theta) = 0.07$ after supervised learning, whereas $) = 0$ after unsupervised learning.

The issues just presented were not taken into consideration when labelling the data and were only discovered when most of the experiments were already completed.

Need for more preliminary research

An obvious starting point for the next stage of this research is to redo all experiments with better labelled data. Given the new results we could decide if Naive Bayes Nets are suitable for citation classification. However, improvements have to be made for the annotation scheme and process.

First, we found that it was extremely difficult to use so many categories (34 in total) as we were sometimes tempted to label with the first 'satisfactory' class, and not the best class. We believe that it is necessary to revisit Garzone's categories and improve upon the scheme to make it more user-friendly and accessible.

Secondly, there is currently no tool that can help us with the annotation. It would be a great improvement to have an integrated interface for reading and annotating the citations. However, for this experiment we were forced to manually tag the citations first on paper, and then to transfer the annotated data to the computer. We feel that this process can and should be automated. This will also solve the problem of limited annotated data as a well-designed tool will motivate more research in the area. Also, with a single tool, multiple annotators can work together without fear of human error due to mistyping the correct label, or recording annotations in a different format.

There are also a number of other issues that have to be resolved. Currently, the classifier considers only words in the same sentence where the citation appears. Although most of the valuable information for citation classification is found there, some useful information can also be found in the text around the citation sentence. If we increase the citation context to include these sentences, we will have to deal with the problem where two different citations have overlapping citation contexts. In this case, the assumption that one mixture component corresponds to one class will be violated, and we will have to modify our model to address this issue or resolve the overlapping problem in a different manner.

There are number of other possible improvements that the citation classifier can benefit from. For example, we could bootstrap the learning algorithm with cue words that have been used in rule-based classifiers and start the EM algorithm from there. An approach like this is described in [35]. Another improvement could be to incorporate information about the location of the citation in the article. This could be very useful in improving the accuracy of the classifier. Similarly we could lemmatize the words in citations contexts and store information such as tense in separate variables. This will also increase accuracy as, for example, "shown" and "showed" will be matched by the classifier, whereas they are now treated as different variables in the current implementation.

In the remainder of this chapter and this thesis, we try to address all these issues in an attempt to improve upon our initial findings.

4.2 An Improved Citation Classification Scheme

4.2.1 Classification Scheme Requirements

Yes, we know. You must be saying to yourself "yet another one?" Those are our sentiments exactly, but we did not have a choice.

Initially, we did not plan to develop a new classification scheme. In fact, we were quite happy with Garzone's (Appendix A.8) scheme. It satisfied both of our main requirements. First, it is a very exhaustive and fine-grained scheme containing an enormous number (35) of citation categories, the most of any classification scheme we reviewed. A very detailed scheme was important for our work because our goal is to differentiate citations as much as possible (at least) during the annotation process. After all, even if we do not achieve good results with the automatic classifier over all categories, it is always possible to combine a few difficult categories in the end. On the other hand, if we had only a few categories to begin with, and achieved good results, we could not then easily make the categories more detailed after they were already annotated. If we want to expand our categories afterwards, we would have to annotate citation data all over again, and this is manually very intensive work.

Our second requirement is that the scheme has to be designed from, and intended for use with, scientific articles, preferably biochemistry articles, as our corpora is composed solely of articles from this discipline. As we showed in the previous chapter, not only does Garzone's scheme provide good coverage for biochemistry articles, classification with his scheme is actually better for biochemistry articles than the physics articles that he experimented with.

4.2.2 Redesign of Garzone's scheme

We realized that we needed a different classification scheme during our initial experiments (as described above) and our judgement was only reconfirmed during the design of our annotation tool (discussed in detail in Section 4.3). There were just too many categories, so that not only did they not all fit in the limited space of the 'floating' annotation window over the citation, but they also confused the human annotators. This concern was voiced by a number of people, as well as the situation that the full-text category labels (e.g., "Citing work is totally supported by cited work") only further cluttered the annotation window.

These problems prompted us to modify Garzone's scheme from both usability and usefulness perspectives. We started out by trying to reduce the number of categories without compromising very much on the fine granularity of the scheme. We felt that dropping Garzone's "Partially" versus "Totally" distinction was reasonable as it is difficult to distinguish between the two categories effectively. Instead, we introduced a method for an annotator to identify the strength of her certainty

about the classification. For example, if a citation is labelled as "not supported" and "not very sure", this will indicate that it is a "partially not supported" citation.

We then took the modification of Garzone's scheme one step further by reducing the full-sentence categories in their citation function components as shown in Table 4.1, which actually represents our new classification scheme.

Reason	**Object**
Confirms	General background
Supports	Specific Background
Illustrates/Clarifies	Historical account
Interprets results	Pioneering Work
Extends model	Related work/Bibliographic Lead
Contrasts	Concept
Mentions in Passing	Method
Future Research	Product
Uses	Data
Direction	
How sure?	

Table 4.1: Our modified classification scheme

The key advantage of using citation function *components*, instead of citation function *categories* is that we can now express categories by simply combining the components. An example of this type of combination was given above in the definition of the "partially not supported" citation category, which we had previously dropped from the classification scheme.

Here we describe all citation function components with examples:

We have three types of components: *Reason*, *Object*, and *Other* components. *Reason* citation function components answer the question "why is this article cited?" while *Object* components are more concerned with the "what". Another specificity of our classification scheme is that all *Reason* components have polarity. In other words, they can be positive, negative, or neutral. For example, a citation can be either supporting a cited work (positive or neutral) or it can state that it does not support the cited work (negative).

Confirms: This component is present when the citing paper somehow confirms or validates the cited work. An example of this case is *"The assignment of disulfides in the C-terminal domain experimentally validates the primary disulfide pattern predicted for NTR modules ([B49])."*

Supports: When the citing work supports some aspect of the cited work. Example: *"This protein has been identified previously as a nuclear serine/threonine kinase that interacts with the NK homeodomain transcription factor ([B46]), acts as a corepressor for the NK homeodomain, and cooperates with Groucho and HDAC-1 in enhancing transcriptional repression ([B47])."*. Here, support is shown by implicitly agreeing with the previous results.

Illustrates/Clarifies: One work clarifies or illustrates something from a different work. Example: *"For example, FIX Q50P has been studied by two different groups ([B43] , [B44])."*

Interprets results: When one work is used to interpret results of another. Example: *"Because EGF1 of activated protein C has a major loop inserted at a position corresponding to FIXa residue 54, it seems unlikely that this part of EGF1 in FIXa makes a direct contact with FVIIIa ([B15] , [B19])."*

Extends model: A model is either extended or created from finding in the cited work. Example: *"In this model, the key interacting regions of FIXa and FVIIIa can be aligned as previously reported with only minor reorientations ([B13] , [B32])."*

Contrasts: When two works are compared. Example: *"Surprisingly, mutation of the first two leucines in the LXXLL motif decreased steroid binding capacity and transcriptional activity without altering receptor levels, cell-free steroid binding affinity, or hsp90 binding ([B25])."*

Mentions in Passing: The work is cited as a perfunctory reference. Example: *"The mammalian BNaC/ASIC branch of the superfamily contains four genes, encoding at least six isoforms: BNaC1 (also known as BNC1, MDEG, and ASIC2) ([B2]) and its differentially spliced isoform, BNaC1 (MDEG2) ([B17]); BNaC2 (ASIC or ASIC1) ([B4] , [B18]) and its differentially spliced isoform, BNaC2 (ASIC) ([B19]); DRASIC (ASIC3 or TNaC) ([B20]); and ASIC4 (SPASIC) ([B24] , [B25])."*

Future Research: Points to future research. Example: *"An open question is whether the described disassembly of transcriptional regulatory complexes by p23 requires ATP ([B61]), as the requirement of hsp90 or hsp70 for the effect of p23 remains to be elucidated."*

Uses: Use of a method, equation, product, etc. Example: *"To study the NF-Y-TFIID connections, we employed the mouse MHC class II Ea promoter system ([B51] , [B52])."*

General Background: Background that is not necessarily needed to understand the citing paper. Example: *"Hitherto, the search for paxillin-binding proteins has involved either yeast 2-hybrid screens ([B8]) or GST-fusion protein pull-down assays ([B6] , [B10] , [B12] , [B26] , [B30])."*

Specific Background: Background that is specific for the citing article. Example: *"The p110 isoforms of PI 3'-kinase played significant roles in cell migration, and differential activation of specific p110 isoforms is responsible for particular signaling events in different cell types ([B32] , [B33])."*

Historical: This is also a background component, but is mentioned chronologically, Example: *"Earlier reports have shown that pV and tumor necrosis factor induced NFB activation in Jurkat cells, and only pV-induced activation of NFB is inhibited by wortmannin ([B21])."*

Pioneering Work: Citing work of pioneers in the field. This is another category that is very difficult to annotate without having an in-depth knowledge of the field. Example: *"Recent evidence indicates that Sina, together with phyllopod, promotes the ubiquitin/proteasome-dependent degradation of tramtrack, a negative regulator of neuronal differentiation ([B29] , [B30])."*

Related work/Bibliographic Lead: The author either describes related work or gives leads for further reading. Example: *"Consistent with previous reports ([B35] , [B37]), myc-tagged Siah-2 was found to be expressed at a relatively low level in transfected PC12 cells, perhaps as a result of self-regulating its own stability (see "Discussion")."*

Concept: A use of a model, definition, hypothesis. Example: *"This staining showed strong colocalization with EEA1 (Fig. F5, C and F), which is consistent with the idea that mVps4 regulates the morphology and the transport functions of endosomes ([B54])."*

Method: Use of method. Example: *"Sequence analyses show that Hrs, Eps15, STAM1, and STAM2 contain UIMs ([B55])."*

Product: Use of a product or material. Example: *"To do this, we used a recently described phage system that displays a highly diverse and random assortment of short peptides fused to the C terminus of the M13 gene-8 major coat protein ([B9])."*

Data: Use or analysis of data. E.g. *"As has been found with virtually all previously examined ligands for type 1 PDZ domains ([B3]), the C-terminal residue (position 0) was found to be hydrophobic."*

Direction: This component represents the three possible directions of a citation: (i) the citing paper describes or uses material from the cited paper (most common type); (ii) two works are compared to each other, i.e., a direction between two cited papers; (iii) and the final, and most interesting, citation direction is from a cited paper to the citing paper. This last citation direction is not very common, but it does occur in papers that are published almost simultaneously. An example of such a citation is: *"Work on applying machine learning techniques for automatic citation classification is currently underway (Teufel et al., 2006)".*

How sure?: This component represents the scale of the annotator's certainty with his classification. Although this was originally meant to judge whether the labelled citation should be included in testing or training sets, it can also

be used to keep track of the strength of the given relationship between the papers.

We believe that the scheme just presented is superior to other annotation schemes both because it is more descriptive (many citation concepts can be chosen for one citation, making it as much or as little fine-grained as one desires), and because it is more intuitive to use.

The change that we made in separating the categories into citation function concepts allows us more flexibility in annotating. For example, consider a statement that cites data that can be used for future research. Even with its many categories, Garzone's scheme cannot handle this citation type properly. With Garzone's scheme, one can annotate this type of citation as 26 (Used in making suggestions for future research) and 28 (Use of numerical data). However, the data was not used in the current work. It only points to the data as a point of future research and therefore this classification would be misleading. With our approach however, because we separate the "why" and the "what", the citation can be described as **Future research** and **Data**, which is a more correct annotation.

As well, because the citation function concepts are separated, if there is an ambiguous citation, the annotator can still annotate what she is sure about and not have to make incorrect annotations. In the above example, even if the annotator was not sure whether the data has been used or not, she has the luxury of annotating exactly what she is certain about (for example, that it is a mention of a data, but it is not clear what they do with it). It is clearly a more natural approach to have citation properties/concepts rather than preset categories.

Finally, we believe that having simple words to describe the concepts is much more natural for an annotator to work with and therefore it is easier to train new annotators. For example, having to click on "General Background" is both more natural and intuitive than having to memorize a classification scheme and annotating with "13" or having to read a long description.

4.3 Citation Acquisition System

Because citation classification is a fairly new field, there are no accessible annotated corpora available to researchers in the field. Another reason for the lack of citation corpora is that there is no common, agreed-upon classification scheme. As we have described in the previous chapter, there are numerous classification schemes, and all attempts to build automatic classifiers have been done with a different classification scheme. It would be much easier for the individual researcher, and beneficial for the advancement of the entire research field, if researchers shared their data and made their classification tools accessible for everyone to use.

In this section, we introduce our Web-based citation acquisition and annotation system that we used for the development of our citation corpus. We designed the

system with the goal of facilitating the addition of new articles to the corpus so that, in this way, authors themselves can upload their own papers and annotate properly their citations.

System Overview

Our system is designed as a Web application: it runs on top of the Apache Web Server and is built with PHP server-side scripting. It has several components as shown in Figure 4.4. The first step in the system is to acquire a document, which can be done either in bulk by a system administrator or by a system user (usually an author or an annotator) uploading a PDF file from anywhere on the web. Then, the system 'kicks off' a workflow that adds the document to our database,[2], keeps on record the hash of the file for later searches, processes it by the $PDF \rightarrow HTML$ converter, parses it with the document parser, tags all citations, and finally presents the annotation tool to the user.

The system can be accessible to anyone who has a javascript-enabled browser. It allows for multiple annotators by requiring a login for annotations, which ensures that annotations will stay consistent. Annotators have the luxury to annotate all or some of the citations in an article. Their work is saved at the end of the annotation process and they can always come back and continue annotating or even change previous annotations. Changing annotations is actually very useful for new annotators as they improve their judgement with more annotations. If the annotators feel that they have made mistakes earlier, they can always come back and edit their annotations.

We believe that this tool will be invaluable to citation classification. Next, we describe the different parts of the system.

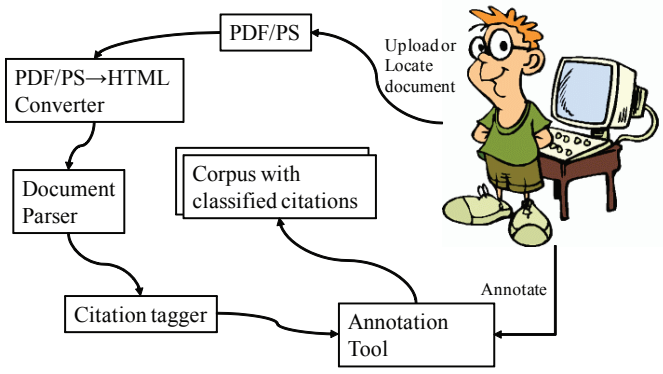


Figure 4.4: Overview of the citation acquisition system

[2]Currently our database is based on flat files for easy movement of the system between servers. However, major DBM Systems can also be used by storing binary/big files as blobs. DBMS is actually the preferred way for a full-scale deployment of the system, but we are not at that stage yet

Acquiring documents

As previously mentioned, our system allows users to provide their own documents. Our simple interface may be observed in Figure 4.5.

Figure 4.5: Searching for a document with our system

The user has three options. First, she can upload a document in PDF format. Nowadays, these formats are ubiquitous so users should not have a problem with finding the document they want to annotate in one of these formats. The second option is to provide a link to a PDF document on the web by selecting the *URL* option and pasting the link to the document in the search text box. The third and final option is to search for a document by title. The system searches the documents already in the database, and documents on the web with Google's "filetype:pdf" option. If the title is found in the local database, then the user can just select it and start annotating it. Otherwise, clicking one of Google's results will copy the URL of the document to the search box, and the system will retrieve the file as in the URL mode. In all cases (except finding a local document), the hash and titles of the uploaded document are first checked against our database, and if a matching local document is found, it is presented to the user and the uploaded document is discarded[3].

PDF converter

Our PDF converter relies on the open source pdftohtml [6] program to convert PDF files to HTML. It is a similar program to the one used in CiteSeer (ps2ascii), but this is no longer supported so we had to use an alternative. However, pdftohtml has one big advantage to ps2ascii. Whereas ps2ascii converts to plain text, pdftohtml converts the document to HTML, which means that it keeps the hierarchical structure and visual representation of the document. Consider the following sample output of pdftohtml:

[3]One of the main drawbacks of this document acquisition method, however, is the issue with copyright. This is why we urge our users to upload only documents that are freely available on the web.

```
<div class="ft0" style="position:absolute;top:67px; left:93px;">
  The Frequency of Hedging Cues in Citation</div>
<div class="ft1" style="position:absolute;top:148px; left:100px;">
  Robert E. Mercer 1 , Chrysanne Di Marco 2 ...</div>
<div class="ft4" style="position:absolute;top:287px; left:108px;">
  Abstract. Citations in scientific writing ... </div>
```

Here, we can easily determine the title of the paper, even if it is not the first line of the file because we have access to the font information (provided by class *ft0* above and its corresponding CSS definition). This font information also helps with the discovery of sections and other elements of a document that have different fonts (footnotes, formulae, definitions, etc.) The way the HTML document is structured also provides invaluable information. Every line in the document is a different *div* element positioned absolutely. With this, we can calculate the distance between lines of text in a paragraph (as there are more lines inside paragraphs than outside). Knowing this information, we can determine the start and end of paragraphs, indented text, and more. This structural information can be very beneficial in further processing the document, but this is not the only advantage of HTML. Another important advantage is that the text is represented to the annotator in the same way that it was in its PDF/printed version with columns, pages, and other textual information preserved. This helps users who have either written the document themselves or have read the document outside our system.

The only flaw with pdftotext is that it does not do OCR, so older articles will not be converted as they store content as images and not text. Also, sometimes it does not recognize certain parts of the document and further processing needs to be done by the Document Parser to fix them.

Document Parser

As mentioned earlier, as good as pdftotext is, it is not perfect. Our Document Parser component tries to resolve errors introduced mainly in the conversion of equations by scanning the converted text and looking for short and misaligned lines. When broken lines split into multiple *div* elements are detected, the parser tries to reconstruct them as well as possible on one line, without exceeding the average line width of the document. Besides fixing minor conversion errors, our Document Parser leverages the benefits of the HTML format discussed in the previous section. Although there is much more information that we could extract from the text, our current implementation of the Document Parser only extracts the title, section names, and the reference section of the document. It also parses the reference section, extracting the reference anchors (if available) and the authors of references articles.

Citation Tagger

To represent the document in our annotation tool, we first need to find the citations in the text and tag them. Detecting references is not the trivial task it may seem because of the many styles of writing in the scientific literature. Table 4.2 lists some common styles for citation anchors. Before we match the references in the text,

[1] [1,2] [1-3]	[Last et al.]
Last08	[Last, 2008]
(Last, 2008)	[Lasta and Lastb]
See Last (2008)	Last and others (2009)
Lasta and Lastb (2009)	AbR08

Table 4.2: Sample citation anchors

we first examine the parsed bibliography from the previous section. With regular expressions, we try to detect the kind of reference it is. For example, the regular expression $\backslash[\backslash([0-9]+[,-]?\backslash)+\backslash]$, will only match the anchors *[1] [1,2] [1-3]* from Table 4.2 above and none of the other anchors in the table. Another, more difficult, example is trying to match the *[Last et al.]* anchor style. In this case, we obtain the parsed author last names from the reference section and append them in a regular expression as follows:

```
\[\(\(Last\)\|\(Lasta\)\|\(Lastb\)\|\( +and +\)\)+\( +et al\.?\)?\]
```

The above regular expression will match the style of *[Last et al.]*, *[Lasta and Lastb]* and any other permutation with the last names *Last*, *Lasta*, and *Lastb*. We match the other types of anchors with regular expressions in a similar fashion.

When reference anchors are located in the text, they are tagged by converting them to hyperlinks that link with the annotation tool that we discuss next.

Annotation Tool

The citation annotation tool is the last step in creating labelled citation corpora. The human annotator is presented with the document in HTML form along with a floating box containing the annotation scheme, as shown in Figure 4.6. The floating box first pops up for the initial unclassified citation. The annotator can move the box as he desires to see the text around the citation as he tries to annotate it properly. Every citation anchor in the document is a hyperlink that, when clicked, displays the classification box for that particular citation. The annotated citations are kept in memory until the annotator presses the *Finished* button at the bottom of the article. Later, the annotator can return to the same document and continue annotating from where he left off. When logging on to the system, the annotator is presented with a list of all articles. The articles are colour-coded, representing whether it has been classified completely, partially, or whether annotation has not

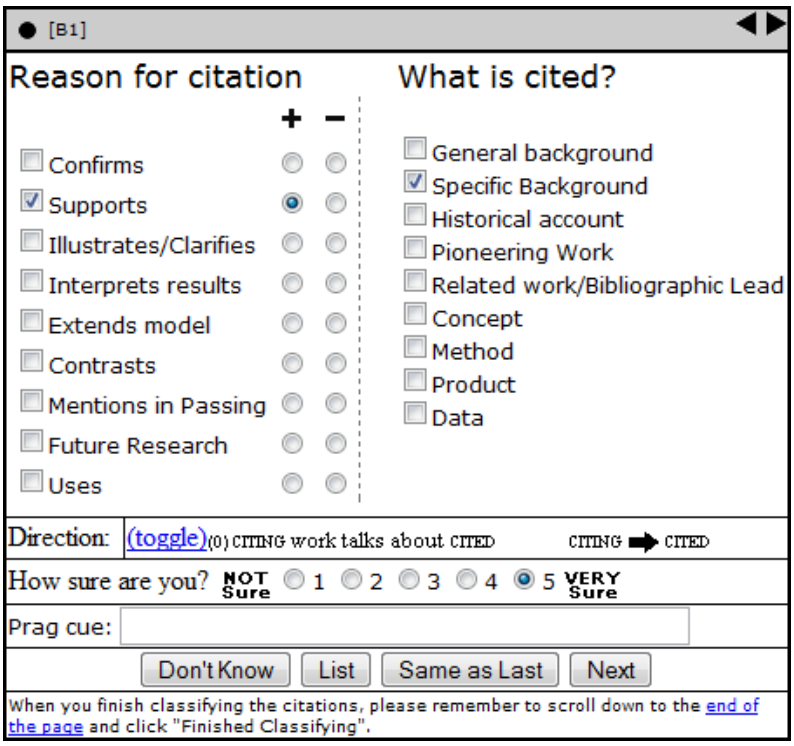


Figure 4.6: Annotation Tool: A floating box containing the classification scheme.

started for the article. Colour-coding is also available for the list of citations within the article. To access the list of citations, the annotator presses the *List* button on the classification box and a pop-up window with a list of all citations appears. The annotator can then quickly click on a citation that was not classified and the box will reposition itself appropriately under the requested citation.

As can be seen in Figure 4.6, multiple citation function components can be selected for the same citation. As mentioned earlier, the "Reasons" for a citation contain positive, negative, and neutral dimensions (neutral is achieved by not selecting positive/negative, or if already selected, by clicking it again to deselect it). Direction can be toggled between its three states as discussed earlier. There is also a pragmatic cue box that we added recently to collect lexical information on cue words and other important textual features from linguistic experts.

When the annotator is finished annotating, the system stores the results in a text file under the corresponding classified article. There can be many users of the system and annotation results are stored individually in different files. Here are shown a sample of several annotated citations:

```
cit2:rate=5|+supports|sbackground|dir=0|name=[B1]|
cit19:rate=5|+supports|data|dir=0|name=[B16]|
cit26:rate=5|+uses|product|dir=0|name=[B22]|
cit51:rate=5|+illustrates|data|dir=0|name=[B30]|
```

The format is quite easy to understand. Each citation is recorded on a single line, and the classified citation components are appended with | in between. For example, in the sample above, we can see that citation 19 (represented as [B16] in the text) of the article has been classified as "positive support of data", where the citing document is supporting the data contained in the cited document. The annotator was very confident in her classification as evidenced by the high rating of 5 out of 5. Annotated data could have also been stored as XML files, but at the time of design, we found it easier to work with Linux command-line utilities and this representation was easier for us to handle. Conversion to XML standard of annotated data is trivial and can be easily automated.

4.4 Annotated Corpora

We collected a corpus of eighty-seven articles from the Journal of Biological Chemistry [8] and thirteen articles from Proceedings of the National Academy of Sciences [7]. The corpus contains roughly 83 citations per paper with a total of 8258 citations. Most articles had the same structure, but some deviated from the norm. It can be observed in Tables 4.3, 4.4 and 4.5 that there was distribution of citations in every section.

Glossary	1
Conclusions	9
Methods	30
Results and Discussion	247
Materials and Methods	377
Experiment	721
Results	1668
Discussion	2356
Introduction	2849
Total	**8258**

Table 4.3: Distribution of citations per section

Results and Discussion	207
Materials and Methods	249
Experiment	716
Results	1457
Discussion	2072
Introduction	2399
Total	**7100**

Table 4.4: Distribution of citations per section in JBC articles

Glossary	1
Experiment	5
Conclusions	9
Methods	30
Results and Discussion	40
Materials and Methods	128
Results	211
Discussion	284
Introduction	450
Total	**1158**

Table 4.5: Distribution of citations per section in PNAS articles

With the help of a domain-expert research assistant[4], we classified a collection of 100 biochemistry articles. The research assistant followed Protocol 1 to annotate all citations. An overview of the annotation results is given in Table 4.6.

It may be observed in Table 4.6 that the results are heavily skewed toward "positive support" and "specific background". This result is similar to the result of our first attempt at annotating with Garzone's scheme. Evidently, the skewing has to do with the natural distribution of citation classes for the particular field, and not with the sample size as we originally feared during our initial experiments.

Another observation that can be made from the annotated data is that there are so few citations that "use" something from another work. We expected to see many more "use of materials" citations, but this was not the case in the corpus of biochemistry articles that we annotated.

4.5 Features

Another step that we have to take before running experiments with our data is to identify the learning features with which to build our classifier. In our initial steps, we used n-gram models, without performing any linguistic analysis of the citation contexts. This time, we decided to consult with a linguistic expert [5] to find if there are any features in the citation contexts that we can explore instead of using n-gram models.

The result of the analysis was a list of lexical features (cue words) and syntactic features. For example, from the lexical category, there were features such as "time and sequence terms", which is a list of cue words 'currently', 'initial(ly)', 'original(ly)', 'previously', 'recent(ly)'. From the syntactical features, examples are "negation" and "parenthesis", and so on. From this list of features, we compiled a collection of 364 attributes which we will use to build out citation classifier. Part

[4]We thank Irene Chau, a Master's graduate in Biochemistry from the University of Toronto.

[5]We would like to thank Olga Gladkova for her invaluable linguistic help and expertise

Protocol 1 Protocol for citation annotation

1. Classify each citation according to its order of appearance in the article (e.g. B1, B4, B2 etc.).

2. Read the sentence before the citation. Even if the sentence contains several different parts (and citations), read the full sentence as it provides a better idea of the purpose of citation. If necessary, read also the previous sentence(s).

3. Select "WHAT IS CITED?" (i.e., the cited content) by considering each category. Multiple categories can be chosen.

4. Select "REASON FOR CITATION" by considering each category. Multiple categories can be chosen. For each category, determine if it is positive or negative.

5. Select "DIRECTION" of citation by clicking the TOGGLE button. Determine if (a) the citing work is talking about the cited work; (b) the citing work is talked about in the cited work; or (c) other work is talking about other work.

6. Select "HOW SURE ARE YOU?" according to the classifier's confidence level for his or her decision, with (5) being very sure and (1) being not sure.

7. Select "DON'T KNOW" in cases where decision could not be made.

8. After finishing with the classification of one citation, continue with the next one by clicking the "NEXT" button.

9. After finishing with classification of an article (or even if it is not finished), save progress by clicking the "FINISHED CLASSIFYING" button.

10. After classification of the first ten articles, re-classify the first five articles and make changes accordingly. Then, continue with the classification of the rest of the corpus.

11. Changes to any classified citations could be made at anytime during the classification process by clicking on the link to the article, and selecting the link to individual citations.

(Direction) Citing talks about cited	8255
(How sure?) 5 out of 5	8099
Supports positively	5662
Specific Background	4553
Data	3777
Illustrates Positively	2514
Uses Positively	1518
Method	1464
Related	824
Mentions in passing positively	802
Concept	725
Product	723
Interprets positively	517
Historical account	404
Confirms positively	355
Pioneer	346
Extends positively	331
Contrasts positively	249
General Background	210
Does not support	181
(How sure?) 4 out of 5	127
Uses	30
Illustrates	27
Supports	26
Future research (positive)	24
Illustrates negatively	18
Mentioned in passing	8
Interprets results	7
Extends	6
Interprets negatively	4
(How sure?) 3 out of 5	3
(Direction) Other talks about other	3
Extends negatively	3
Contrasts	2
Confirms negatively	1

Table 4.6: Distribution of citation function categories

of this list are also hedging cues that [16, 36] found to occur more frequently in citation contexts. We wanted to see if these prevalent cues in citation contexts can help in the classification task. Here is the list of all attributes:

- Locational and article wide features. These features are the most important:
  - Section: In which section of the document does the citation appear
  - Self Reference: Is the citation a reference to a publication by one of the authors
  - Years apart: How far apart are the publications of the citations.
  - In figure: Does the citation appear inside a figure caption?
  - Shared: Does the citation have overlapping context with another citation? For example, the sentence immediately before the sentence containing a citation is the sentence immediately after another citation.
  - Context0/2: Are there citations inside the context of another citation?
- Part of speech tags: what are the POS tags and how frequent are they in the citation context
- Syntactical features: Presence of parentheses or negations.
- Cue words: *a few, a typical, abnormally, accessible in, according to, achieved, acts as, all, also, analogous, analogous to, analyze, another, appear to, applied, appropriate, arising from, as, as also shown in this work, as described, as evidenced by, assumed, assures, attributed, backbone, bars, based on, basic, both, broad, but, but also, by analogy with, call, can, carried out, central role, characteristic, characterized, check, clarified, class, classical, compare, comparison, considered, consist, contains, corresponds to, could, currently, define, defined as equation, demonstrated, denoted, derive, describe, detail, determined, developed, devoted to, difference, different, direct, distinct, distinguish, diverse, done, due to, earlier, eliminating from, encounter, essential, established, estimated, evidence for, evolved, examine, example, excellent, except, except in, executed, exhibit, expanded, expect, explain, explanation, extend, extending, extension, figure, finally, find, finds, found, full-length, further details of, furthermore, general characteristics, generally agreed, gift, giving, group, handled, has, has not yet been, hatched line, have also been made, have been reported in another paper, have since been, however, identified, identify, implemented, implicate, implicated, implies, important, in addition, in addition to, in agreement, in common, in contrast, in progress, in the past, in vitro, in vivo, include, indeed, indicated, indicating, initially, integral, interestingly, interpreted, investigated, is, is known, is present, is related to, is said to be, isolate, justified, key, known to, less, like, manuscript, many, may, mediate, member, method, might, model, modelled by, more, more on, must await, narrow, not reproduce, notably, novel, observed, obtained, occur, occurs also for,*

on the basis of, on the other hand, only, open boxes, originally, parallel to, performed, plays a major role, pointed out, postulated, preparation, present, presented, previously, primary, probably, proposed, purchase, rather, reads, recent, recently, recognized, reacts, regard, rename, reported, represent, reproduce, reproduces, reproducing, required, requires, respectively, responsible for, resulted, resulting, results, revealed, review, role, seem to occur, series, set, several, shaded boxes, should be, show, shown, shown in fig, similar, similar results for, similar to, similarly, solid line, species, specific, specifically, strongly, studied, study, subjected to, such as, suggested, suggesting, suggests, super, supported, table, take into account, tempting, than in, the following, their agreement with, thereby, therefore, this, thought, thought to, thus, to obtain, transform, type, typical, ubiquitous, ultimately, uncovered, unknown, unusually, used, various, very close to, was based on, was first, we, whole, would be particularly interesting, written as, yet, yields

- Hedging cues: *about, almost, apparent, apparently, appear, approximate, approximately, around, assume, attempt, believe, calculate, consistent, essentially, estimate, evidently, generally, imply, indicate, likely, most, mostly, normally, note, occasionally, partial, partially, possibility, potentially, predict, presumably, probable, probably, propose, quite, rarely, relatively, report, see, seek, seem, slightly, some, somewhat, speculate, suggest, suspect, unlikely, usually, virtually*

In the next chapter, we make use of all of these contributions to perform some preliminary citation classification tests. We present results of our experiments, followed by a discussion.

Chapter 5

Experiments

5.1 Methodology

The first step was to acquire a large, annotated corpus, which was discussed in the previous chapter. This corpus had to be additionally processed before we could experiment with the data. Here, we describe the process of data preparation for our experiments.

1. Since our corpus was in HTML format, we had to convert it to text before further processing. We took the most natural step. Instead of devising rules to parse the entire HTML document, we parsed it visually through *lincs*[1]. This saved a significant amount of time, but introduced other problems. Lines were split internally by *links*, which we had to subsequently fix. Also, visual information was lost and the detection of superscripts and other font information was no longer available. We had to use regular expressions to fix much of the noise introduced by this process.

2. Next, we ran sentence detection with Maximum Entropy sentence detection models from OpenNLP [5]. This process did not go very well because the models were not trained on biomedical data with a large number of different abbreviations of which the sentence detector was unaware. This forced us to manually correct the errors of the detector. However, since we had already performed this step for these experiments, we can just train a new sentence detector model for subsequent similar tasks to avoid segmenting the sentences by hand.

3. After the articles were tokenized into sentences, we performed some article-wide parsing to collect information about the *Self-Reference*, *In Figure*, and *Years between publications* Attributes. To detect *Self-Reference*, the title and author sections were detected and the names of authors were converted

[1] lincs is a textual web browser popular in Linux and Unix environments

into regular expressions that matched text from the bibliography section. Then, for each citation, we searched the bibliography section for the author regular expression. We performed a similar task for calculating years between publications, as the publication date of most articles was supplied in footnotes at the end of the articles. For the remaining articles, we had to manually locate the date of their publication.

4. The next step was to extract the contexts of each citation. This was performed by first locating a citation and extracting the sentence where it was located. Then, sentences on either side of the citation sentence were detected and the presence of other citations or shared context was recorded in the *Context0/Context2* attributes. The *Shared* attribute is just a binary variable denoting whether the context is shared or not. *Context0* is the Context immediately before the citation sentence. Its values can be either -1 (denoting that the citation sentence is at the start of a section), 0 (denoting that there is a sentence above the citation sentence and that its attributes are used in the context), and > 0 in the case if there are citations found in the previous sentence. In the case of citations, we do not use that sentence in the citation context, but instead we just record the number of citations in the previous sentence in this variable. The attribute *Context2* is the same as *Context0*, but it relates to the sentence immediately following the citation sentence.

5. Following this, we parsed the citation contexts with the Maximum Entropy parser from the OpenNLP tool set. This was followed by lemmatization of Nouns, Adverbs, Adjectives, and Verbs. We used WordNet [9] and the Java WordNet Library [4] to accomplish this task.

6. Then, our attributes had to be prepared for searching through the citation contexts. This was accomplished with regular expressions. Each lexical attribute was searched for in the lemmatized version of the citation context. Because the lemmatization was not perfect, a fall-back mechanism was in place where the original citation context is searched (with an alternate regular expression) if nothing matched the lemmatized one.

7. And finally, the last step before the experiments was to encode all attributes in separate *.arff*[2] files. Every feature in our feature set was a separate attribute and therefore a separate dimension in the feature vectors.

5.2 Results

As mentioned earlier, we used Weka [51] as our platform for classification and analysis of our citation corpus. For every citation function concept in our scheme,

[2].arff files were needed for processing with Weka [51]. This format represents a list of the attributes, followed by a list of observation vectors with those attributes

we built a separate Naive Bayes Net classifier and we validated our results by performing 10-fold cross-validation[3]. Naive Bayes was used to be consistent with our earlier approach, and also because it is a good exploratory mechanism (the model is trained considerably fast and inference is also fast).

We also used several other classifiers, but the only other one we will mention here in detail (as all other approaches yielded similar results) is the k-nearest neighbour (k-NN) algorithm. This is the same algorithm used by Teufel *et al.* in their citation classification experiments [47], where they used $k = 2$. In our trials, we experimented with different values for k as well, but we did not find significant difference between the number of neighbours and here we present the results for 1-nearest neighbour. The reason for our inclusion of the k-nearest neighbour algorithm is not to compare our results to those of Teufel *et al.*'s (as there is no basis for comparison: different scheme, corpus, and features), but because it is another good exploratory model. Actually, there is no learning involved when using the k-NN algorithm. The model is the training data. However, there is a significant amount of computation involved when classifying, as each test instance has to be compared with every training instance. This model is particularly impractical with a large amount of training data and it is virtually infeasible for large-scale classification tasks such as citation classification. With that said, it is still a useful exploratory classifier and our corpus contained only 8258 citation contexts, so the computational penalty is not very high (yet).

We actually found k-nearest neighbour quite useful, because it allowed us to find a significant flaw in our corpus. In our collection, there were many citation sentences with a long sequence of citations. For every one of those citations that shared the same context, most of the features (as they are from the same citation context) were the same. The only features that were different were the number of *Years* between citing and cited article and whether the citation is a *Self* citation. There were a total of 2674 citations with duplicate citation contexts and their presence was skewing the results in favour of the 1-nearest neighbour (with Euclidean distance measurement). We would have overlooked this flaw if it were not for the k-nearest neighbour algorithm, and that alone makes it a worthy classifier[4].

With that settled, our corpus was limited to only 5583 citations (out of 8258), and we repeated all the experiments with the new numbers. The results of the classification tasks for Naive Bayes and 1-nearest neighbour on each citation function concept from our scheme can be found in Appendix B on page 65. For all tests, a 10-fold cross-validation was performed on the entire corpus and the results shown are the combination of the results from all 10 folds. One classifier was build

[3]10-fold cross-validation is the process of separating all annotated data in 10 samples. Nine samples are used for training and the remaining sample is used for validating the classifier. This process is performed 10 times, where each time a different set of training and testing samples is selected from the initial partition

[4]We wonder whether the results of Teufel *et al.* also suffer from the same problem. They use 10-fold cross validation just as we do, and if their data is not clean of duplicates, their results with the k-nearest neighbour are suspect.

for every citation function concept, and per class Precision, Recall, and F-Measure (Section 2.3.1) are presented in Appendix B along with the detailed confusion matrix. In the confusion matrix, the top-left-bottom-right diagonal represents the correctly classified instances, and the top-right-bottom-left diagonal represents the incorrectly classified citations. Percentage accuracy can be calculated by summing the correctly classified citations (top-left diagonal) and dividing by the total number of instances (sum of all cells in the confusion matrix, or 5583).

Let's consider first the performance of the classifiers on the binary citation function concepts. Those with the highest Kappa statistics[5] for Naive Bayes (at or above 0.4) are *sbackground* (Tables B.1 and B.2), *data* (Tables B.3 and B.4), *method* (Tables B.5 and B.6), and *product* (Tables B.7 and B.8). These concepts all had above 0.5 F-Measure on the presence of the concept in the test sets, with over 70% accuracy over all. 1-near neighbour also performed very well on these concepts.

Not surprisingly, the classifiers erred more for the remainder of the binary concepts, as they were represented in the entire data set at or under 10% in total. However, all of them were classified above Kappa 0.1 (for Naive Bayes), with *General Background* (Tables B.17 and B.18) being the lowest at 0.1146 for Naive and almost equal to chance for the 1-NN at 0.0441. But this is not surprising, given the little amount of total examples of this type at under 2.4%. *Related* (Tables B.9, B.10), *Concept* (Tables B.11, B.12), and *Historical* (Tables B.15, B.16) were at Kappa around 0.25 and accuracy over 76%. The most surprising of all, however, is *Pioneer* (Tables B.13, B.14), with a surprising 0.58 Kappa by the 1-NN classifier with a low F-Measure of over 0.6. We believe this is due to a few very similar examples, where the Euclidean distance score is relatively low.

The functional categories with the extra positive/negative dimension performed worst than the concept categories above. Only two categories had a Kappa statistic more than 0.2, and they were *Uses* (Tables B.19, B.20) and *Supports* (Tables B.21, B.22), with K over 0.72 and over 0.45. and accuracy of over 90% and 70%, respectively. We believe that this is due to the presence of many examples in our data set for these two categories (22% and 67%). With agreement with other studies, most citations of this sort are either positive or non-existent. To better illustrate this point, consider that our *Supports* category has the most citations in the negative dimension (i.e., the cited paper is not supported by the current work), and for all 5583 citation context, negative support appears only 143 times. That is less than 3% of all citations! Neutral citations do not even come close to this number across all categories (a total of 79 citations for all categories).

There really is not much data across all other categories to make any significant conclusions on the classifier or the features. The only exception is the category *Illustrates* (Tables B.25, B.26). 27% of all citations are of type *illustrates*. However, the Kappas for both classifiers are under 0.17 and the accuracies are 59% for Naive

[5]Measures agreement between two sets of data, where 1 is complete agreement, 0 is agreement expected by chance, and -1 is total disagreement

Bayes, and 66.8% for 1-NN. Our only explanation is that our features do not cover well this category, and so we need to do more research in the area. The rest of the categories in this section have fairly high accuracies, but their F-Measures and Kappas are low because of the small number of training samples. Their accuracies, Kappas, and low F-Measures are respectively: *Extends* (Tables B.23, B.24), 5% of the entire corpus with 84%, 0.18, and 0.24; *Confirms*(Tables B.27, B.28), 5% of the corpus with 81%, 0.17, and 0.24; *Interprets*(Tables B.29,B.30), 6% of the corpus, with 79%, 0.17, and 0.25; *Passing*(Tables B.31,B.32), 4% of the corpus, with 85%, 0.16, and 0.22; *Contrasts*(Tables B.33,B.34), 3% of the corpus, with 83%, 0.10, and 0.15; and finally, *Future*(Tables B.35,B.36), 0.29% of the corpus, with 95%, 0.01, 0.01.

In conclusion, we can see that our simple classifier performs fairly well with context features extracted from only a handful of articles. Unsupervised machine learning techniques should be able to augment the number of features. Nevertheless, work in citation classification using Machine Learning methods looks promising.

Feature Selection

In the previous section, we provided the results of the classification tasks with our attributes. Let's take a look now at the most descriptive features for every class. Feature selection was performed by correlation-based feature subset selection [25]. With this method, the individual predictive ability of each feature is considered along with how redundant the features are. The preferred features are those that have high correlation with the class, but low correlation with the rest of the features. The results of the feature selection can be seen in Appendix C. Again, 10-fold validation was performed and the numbers show high-correlation between the feature and the class per each fold. As can be seen, there are only a handful of important features per each class. We need to expand this set as our model will not generalize well on unseen data, which can also be observed in the results section above.

We also experimented with building classifiers only with the extracted features, but that did not have sufficiently significant results for discussion. Also, as there are co-dependent classes, we tried learning a General Bayes Net, but the structure generalized to a Naive Bayes, and the classifications from that model were similar to those already discussed. Iterative improvement was also attempted, where the classifiers first learn without evidence from the other classifiers during the first pass (the classifiers are actually treated as random variables that are unobserved). During the second pass, the classifications of the classifiers are fed into each other. After the second pass, we observed a slight improvement (on average 0.3 F-Measure), but during the third pass, performance decreased significantly.

With this, we conclude this chapter. In the next chapter, we discuss our findings in this thesis and provide ideas for future research before we conclude.

Chapter 6

Conclusion and Future Work

6.1 Future Work

In this thesis we have only scratched the surface of Automatic Citation Classification with Machine Learning techniques. More work needs to be done on a general annotation tool and a standardized classification scheme that can be used by researchers from different organizations. Currently, research in this area is very ad hoc. Everyone adapts their own citation scheme and tackles developing their own corpus, which means they have to invest a great deal of time acquiring suitable data (especially for machine learning tasks).

There is already work underway by the *IN3SCAPE* group of the University of Waterloo, led by Prof. Chrysanne DiMarco, to develop a more general annotation tool, which is actually based on the annotation tool presented in this thesis. It not only collects annotated citations, but is able to annotate text with other pluggable annotation schemes. A tool like this can be invaluable to the state of citation classification research for the collection and annotation of a standardized corpus for evaluation and performance analysis of citation classification systems. Currently, there is only one researcher in citation classification who has tried to compare results with previous work (Pham [43] comparing with Nanba [40]). But Pham's work was done with a completely different scheme (although derived from the same source), and also trained and evaluated on completely different corpus. A comparison like this is not sensible, and this is why we have not compared our results with previous works.

Another area of future research that is greatly needed is the mining of linguistic patterns and discourse relations that can be used for citation classification. Similar work has been done by Marcu and Echihabi [34], where they present an unsupervised approach for finding patterns to recognize four different discourse relations. In their method, they first build a few patterns that roughly identify the relations they want to be able to distinguish (e.g. "[Beg-of-Sentence] ... [EOS] [BOS But ... EOS]" is one of their patterns to identify the "CONTRAST" relation.) Then,

they find many examples of this relation (of course, some will not be part of the relation, but their method hinges on the fact that in a lot of data, they will find some noise, but predominantly the patterns they are looking for) and use them to build a Naive Bayes classifier to distinguish between the relation by first removing the simple pattern they used to find the relation. Similarly, for the task of finding patterns for citation classification, we start with a few cue phrases or cue words that identify a class relatively well. Then we mine a large collection of citations (in Marcu's method, he uses more than a million examples for each relation) that we currently don't have, but can be acquired in the future with the ubiquity of electronic articles on the Internet (currently, the only concern being copyright). Next, we remove the pattern we used for mining the examples (so we do not learn this pattern again), identify pairs of words (bigrams or larger n-gram models) $(w_i, w_j) \in W_1 x W_2$ (where W_1 and W_2 are two text spans in the citation context), and compute their probability. At the end of the process, the idea is to end up with a list of the most predictive patterns (w_i, w_j) that identify the citation category. The reason why we haven't taken this approach in this thesis is the lack of a massive corpus needed for this task, and because of the copyright implications that may arise from such an undertaking.

And the final area of research is to build, once a good collection of patterns is collected, a classifier that is more robust than a Naive Bayes or a 1-NN classifier. As we mentioned earlier, Naive Bayes makes independent assumptions that may not always be justified, while the main problem with k-NN algorithms is that they are slow at classifying (due to the lack of model-building), and in addition the estimation of k and distance measures are very ad hoc. Naive Bayes also has problems such as the inability to use continuous variables and also cannot handle large priors effectively (as we saw in our results section, where categories with large priors were predicted least accurately by Naive Bayes).

Instead, we propose to look at Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs) as alternative models for citation classification. They handle continuous variables well and also work well with sequential data like citations in a discourse. With such methods, the class of a preceding classification can be exploited to predict the next one.

6.2 Conclusion

In this thesis, we explored the field of Citation Classification, a young research field that needs a lot more work to mature as we saw in the previous section.

In the beginning, we started with Garfield's ideas of citation indexes and explored the field of citation analysis in detail. We looked at several of the most important citation classification schemes and discussed their findings. We also looked at research on Automatic Citation Classification and discussed the pros and cons of each, while motivating the need for more research in the field.

Next, we presented our preliminary work on citation analysis and motivated the need for a larger annotated corpus, the need for a better annotating experience and classification scheme, and the need for better features for classification.

We then introduced our annotation scheme based on citation function concepts, rather than categories, which gives better flexibility and experience during the annotation process.

We also introduced our unique, Web-based annotation tool that allows annotators to provide their own articles for annotation or annotate existing articles. It provides mechanisms for storing the annotations, resuming annotating capabilities, editing previously annotated citations, and all this in a multi-user environment.

With the help of this tool, we acquired and annotated a corpus of 100 biochemical articles. We described the annotation process in detail and provided some statistics on the corpus.

We also presented our features for the machine learning task and described the methodology for detecting the features in text. These features were used to build several classifiers and we presented the results of two (Naive Bayes and 1-near neighbour classifiers). None of the classification tasks were below Kappa of 0 (expected classification by chance) and most (with good representation in the corpus) were above 0.5.

Finally, we presented the most prevalent features for every citation classifier, and we concluded with a look at possible future developments in the field. We hope that you enjoyed reading this thesis and that it motivates you in plunging into this new and exciting area of research!

Appendix A

Citation Classification Schemes

A.1 Garfield [20]

1. Paying homage to pioneers
2. Giving credit for related work (homage to peers)
3. Identifying methodology, equipment, etc.
4. Providing background reading
5. Correcting one's own work
6. Correcting the work of others
7. Criticizing previous work
8. Substantiating claims
9. Alerting to forthcoming work
10. Providing leads to poorly disseminated, poorly indexed, or uncited work
11. Authenticating data and classes of fact – physical constants, etc.
12. Identifying original publications in which an idea or concept was discussed
13. Identifying original publication or other work describing an eponymic concept or term as, e.g. Hodgkin's Disease, Pareto's Law, Friedel-Crafts Reaction, etc.
14. Disclaiming work or ideas of others (negative claims)
15. Disputing priority claims of others (negative homage)

A.2 Weinstock [48]

Weinstock actually does not propose a new classification scheme. Instead, in his work, he uses Garfield's scheme A.1 above without citing it and is mistakenly used as a starting point for other researcher's schemes. This is included here only for referencing purposes since some of the works we review are based on Weinstock's (but actually Garfield's) scheme.

A.3 Moravcsik and Murugesan [37]

1. Conceptual/Operational
   (a) Conceptual: Concept or theory used
   (b) Operational: A tool or physical technique used
2. Organic/Perfunctory
   (a) Organic: The reference is truly needed for the understanding of the paper
   (b) Perfunctory: The reference is just and acknowledgement of previous work
3. Evolutionary/Juxtapositional
   (a) Evolutionary: The paper is a continuation of the cited work or the cited work is a foundation for the paper
   (b) Juxtapositional: The paper is an alternative to the work cited
4. Confirmative/Negational
   (a) Confirmative: The referenced paper is correct/the current paper agrees with the referenced paper
   (b) Negational: The paper contradicts or diminishes the referenced paper

A.4 Chubin and Moitra [15]

Affirmative

1. Basic: The referenced paper is central to the paper. A reference on which the findings of the current paper depend on (e.g. the source of the derivation of a fundamental equation or a detailed description of the experimental conditions)

2. Subsidiary: The specific method, tool, or a mathematical result cited is not directly connected to the subject of the paper, but is still essential for the citing paper.

3. Additional Information: The referenced paper contains an independent supportive observation (idea or binding) with which the citing paper agrees.

4. Perfunctory: Related to the research in the citing paper, but not essential.

   Negational

5. Partial: The cited work is erroneous in part, but not completely.

6. Total: The cited work is completely wrong and the citing paper provides a correct solution.

A.5 Spiegel-Rosing [45]

1. Cited source is mentioned in the introduction or discussion as part of the history and state of the art of the research question under investigation

2. Cited source is the specific point of departure for the research question investigated

3. Cited source contains concepts, definitions, interpretations used (and pertaining tothe discipline of the citing article)

4. Cited source contains the data (pertaining to the discipline of the citing article) which are used sporadically in the citing text

5. Cited source contains the data (pertaining to the discipline of the citing article) which are used for comparative purposes, in tables and statistics

6. Cited source contains data and material (from other disciplines than citing article) which is used sporadically in the citing text, in tables or statistics

7. Cited source contains the method used

8. Cited source substantiates a statement or assumption, or points to further information

9. Cited source is positively evaluated

10. Cited source is negatively evaluated

11. Results of citing article prove, verify, substantiate the data or interpretation of cited source

12. Results of citing article disprove, put into question the data or interpretation of cited source

13. Results of citing article furnish a new interpretation/explanation of the data of the cited source

A.6 Oppenheim and Renn [42]

1. Historical background

2. Description of other relevant work, e.g., describing or discussing the workin some detail or quoting from its results, or saying how the theory could be used

3. Made specific use (other than for comparison) of information contained in the cited paper

4. Made use of data for comparison purposes

5. Use of theoretical equation for calculation purposes

6. Use of practical or theoretical methods in the cited paper to solve a problem

7. Criticism of the cited paper

A.7 Finney [17]

The following is Garzone's [23] account of Finney's thesis as we could not locate the original source:

1. Assumed Knowledge references: "References which are providing the background to the research are assumed, and which represent the core literature."

2. Tentative references: "Concepts, ideas, theories, etc., have been extracted from these references and represent the unknown, unproven areas around the established work and they help to shape the objectives of the research."

3. Methodological references: "Methods, techniques, apparatus and criteria for diagnosis have been extracted from these references and have been utilized directly in the research."

4. Confirmational references: "Either these references are used to support the author's findings or they are confirmed by the author."

5. Negational references: "Either these references do not support the author's findings or they are refuted by the author."

6. Interpretational/developmental references: "Specific results from these references are used to elucidate the author's findings and/or to develop a new hypothesis."

7. Future research references: "References which are brought in to suggest future implications for the reported research findings."

A.8 Garzone [23]

Negational Type Categories

1. Citing work totally disputes some aspect of cited work.
2. Citing work partially disputes some aspect of cited work.
3. Citing work is totally not supported by cited work.
4. Citing work is partially not supported by cited work.
5. Citing work disputes priority claims.
6. Citing work corrects cited work.
7. Citing work questions cited work.

Affirmational Type Categories

8. Citing work totally confirms cited work.
9. Citing work partially confirms cited work.
10. Citing work is totally supported by cited work.
11. Citing work is partially supported by cited work.
12. Citing work is illustrated or clarified by cited work.

Assumptive Type Citations

13. Citing work refers to assumed knowledge which is general background.
14. Citing work refers to assumed knowledge which is specific background.
15. Citing work refers to assumed knowledge in an historical account.
16. Citing work acknowledges cited work pioneers.

Tentative Type Categories

17. Citing work refers to tentative knowledge.

Methodological Type Categories

18. Use of materials, equipment, or tools.
19. Use of theoretical equation.
20. Use of methods, procedures, and design to generate results.
21. Use of conditions and precautions to obtain valid results.
22. Use of analysis method on results.

Interpretational/Developmental Type Categories

23. Used for interpreting results.
24. Used for developing new hypothesis or model.
25. Used for extending an existing hypothesis or model.

Future Research Type Categories

26. Used in making suggestions of future research.

Use of Conceptual Material Type Categories

27. Use of definition.
28. Use of numerical data.

Contrastive Type Categories

29. Citing work contrasts between the current work and other work.
30. Citing work contrasts other works with each other.

Reader Alert Type Categories

31. Citing work makes a perfunctory reference to cited work.
32. Citing work points out cited works as bibliographic leads.
33. Citing work identifies eponymic concept or term of cited work.
34. Citing work refers to more complete descriptions of data or raw sources of data.
35. Citing work makes citation in connection with its figures and tables.

A.9 Nanba and Okumura [40]

Very simple system design to help with automatic summarization

1. **Type B**: The references to base on other researcher's theories or methods.
2. **Type C**: The references to compare with related works or to point out their problems.
3. **Type O**: The references other than B and C.

A.10 Pham and Hoffmann [43]

1. **Basis**: One work is based on another work.
2. **Support**: One work is supported by another work
3. **Limitation**: One work has been criticized to have some limits or weaknesses.
4. **Comparison**: Two approaches are compared.

A.11 Teufel et al [46]

1. **Weak**: Weakness of cited approach
2. **CoCoGM**: Contrast/Comparison in Goals or Methods (neutral)
3. **CoCoR0**: Contrast/Comparison in Results (neutral)
4. **CoCo-**: Unfavourable Contrast/Comparison (current work is better than cited work)
5. **CoCoXY**: Contrast between 2 cited methods
6. **PBas**: author uses cited work as starting point
7. **PUse**: author uses tools/algorithms/data
8. **PModi**: author adapts or modifies tools/algorithms/data
9. **PMot**: this citation is positive about approach or problem addressed (used to motivate work in current paper)
10. **PSim**: author's work and cited work are similar
11. **PSup**: author's work and cited work are compatible/provide support for each other

12. **Neut**: Neutral description of cited work, or not enough textual evidence for above categories or unlisted citation function

Appendix B

Classification Results

Legend:

- For **binary** features (e.g. *concept*), **0**feature, denotes the absence of the feature, whereas **1**feature, denotes the presence of the feature.
- For binary features that have a positive/negative dimension, the syntax is the same (0 for absence, 1 for neutral presence), with the added **2**feature (the feature is present and positive) and **-1**feature (denoting a present, but negative citation.)
- The only exception is the directionality element *dir*, that has *0dir* representing a citing paper talking about cited paper, *1dir*, a cited paper talking about the citing paper, and *2dir*, which is when the citing document compares two works with each other.

	Precision	Recall	F-Measure	Class
Naive	0.779	0.626	0.694	0sbackground
	0.694	0.827	0.755	1sbackground
IB1	0.701	0.655	0.677	0sbackground
	0.684	0.728	0.705	1sbackground

Table B.1: Accuracy by class for Specific background

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0sBackground	1726	1030	1805	951
	1sBackground	490	2337	770	2057

Table B.2: Confusion matrix for Specific background

	Precision	Recall	F-Measure	Class
Naive	0.732	0.675	0.703	0data
	0.649	0.709	0.678	1data
IB1	0.688	0.709	0.698	0data
	0.644	0.621	0.632	1data

Table B.3: Accuracy by class for data

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0data	2040	982	2142	880
	1data	745	1816	970	1591

Table B.4: Confusion matrix for data

	Precision	Recall	F-Measure	Class
Naive	0.907	0.88	0.893	0method
	0.573	0.64	0.605	1method
IB1	0.873	0.893	0.883	0method
	0.533	0.484	0.507	1method

Table B.5: Accuracy by class for method

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0method	3924	536	3983	477
	1method	404	719	579	544

Table B.6: Confusion matrix for method

	Precision	Recall	F-Measure	Class
Naive	0.97	0.869	0.917	0product
	0.376	0.748	0.501	1product
IB1	0.946	0.937	0.942	0product
	0.455	0.497	0.475	1product

Table B.7: Accuracy by class for product

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0product	4385	663	4730	318
	1product	135	400	269	266

Table B.8: Confusion matrix for product

	Precision	Recall	F-Measure	Class
Naive	0.947	0.785	0.859	0related
	0.258	0.629	0.366	1related
IB1	0.915	0.958	0.936	0related
	0.41	0.245	0.307	1related

Table B.9: Accuracy by class for related

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0related	3921	1071	4783	209
	1related	219	372	446	145

Table B.10: Confusion matrix for related

	Precision	Recall	F-Measure	Class
Naive	0.951	0.808	0.873	0concept
	0.227	0.574	0.325	1concept
IB1	0.925	0.944	0.934	0concept
	0.281	0.224	0.249	1concept

Table B.11: Accuracy by class for concept

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0concept	4106	977	4797	286
	1concept	213	287	388	112

Table B.12: Confusion matrix for concept

	Precision	Recall	F-Measure	Class
Naive	0.989	0.856	0.918	0pioneer
	0.202	0.789	0.321	1pioneer
IB1	0.98	0.987	0.983	0pioneer
	0.663	0.553	0.603	1pioneer

Table B.13: Accuracy by class for pioneer

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0pioneer	4570	767	5268	69
	1pioneer	52	194	110	136

Table B.14: Confusion matrix for pioneer

	Precision	Recall	F-Measure	Class
Naive	0.97	0.838	0.899	0historical
	0.158	0.54	0.245	1historical
IB1	0.959	0.974	0.966	0historical
	0.364	0.268	0.309	1historical

Table B.15: Accuracy by class for historical information

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0historical	4428	857	5145	140
	1historical	137	161	218	80

Table B.16: Confusion matrix for historical information

	Precision	Recall	F-Measure	Class
Naive	0.99	0.838	0.907	0gbackground
	0.086	0.634	0.151	1gbackground
IB1	0.978	0.98	0.979	0gbackground
	0.07	0.061	0.065	1gbackground

Table B.17: Accuracy by class for General background

		Classified			
		Bayes		IB1	
		0	1	0	1
Real	0gBackground	4568	884	5345	107
	1gBackground	48	83	123	8

Table B.18: Confusion matrix for General background

	Precision	Recall	F-Measure	Class
Naive	0	0	0	-1uses
	0.958	0.957	0.958	0uses
	0.059	0.357	0.101	1uses
	0.815	0.72	0.765	2uses
IBk	0	0	0	-1uses
	0.954	0.951	0.953	0uses
	0.091	0.107	0.098	1uses
	0.809	0.813	0.811	2uses

Table B.19: Accuracy by class for uses

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1uses	0	0	0	0	0	0	0	0
	0uses	0	4183	10	178	0	4159	7	205
	1uses	0	2	10	16	0	2	3	23
	2uses	0	181	150	853	0	198	23	963

Table B.20: Confusion matrix for uses

	Precision	Recall	F-Measure	Class
Naive	0.07	0.411	0.119	-1support
	0.786	0.592	0.675	0support
	0	0	0	1support
	0.838	0.783	0.809	2support
IBk	0.08	0.065	0.071	-1support
	0.714	0.644	0.677	0support
	0	0	0	1support
	0.808	0.851	0.829	2support

Table B.21: Accuracy by class for supports

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1supports	51	7	1	65	8	17	0	99
	0supports	235	1074	31	475	19	1169	3	624
	1supports	2	3	0	10	0	4	0	11
	2supports	445	282	62	2840	73	447	19	3090

Table B.22: Confusion matrix for supports

	Precision	Recall	F-Measure	Class
Naive	0	0	0	-1extends
	0.978	0.857	0.913	0extends
	0	0	0	1extends
	0.148	0.569	0.235	2extends
IBk	0	0	0	-1extends
	0.965	0.975	0.97	0extends
	0	0	0	1extends
	0.264	0.203	0.229	2extends

Table B.23: Accuracy by class for extends

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1extends	0	1	0	1	0	2	0	0
	0extends	3	4578	5	759	4	5210	1	130
	1extends	0	3	0	1	0	3	0	1
	2extends	0	99	1	132	0	184	1	47

Table B.24: Confusion matrix for extends

	Precision	Recall	F-Measure	Class
Naive	0.007	0.071	0.013	-1illustr
	0.809	0.614	0.698	0illustr
	0.008	0.091	0.015	1illustr
	0.375	0.535	0.441	2illustr
IBk	0	0	0	-1illustr
	0.771	0.787	0.779	0illustr
	0	0	0	1illustr
	0.379	0.356	0.367	2illustr

Table B.25: Accuracy by class for Illustrate/Clarify

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1illustrates	1	5	0	8	0	11	0	3
	0illustrates	107	2499	168	1295	7	3201	11	850
	1illustrates	0	7	2	13	0	13	0	9
	2illustrates	30	577	81	790	18	925	9	526

Table B.26: Confusion matrix for Illustrate/Clarify

	Precision	Recall	F-Measure	Class
Naive	0	0	0	-1confirms
	0.976	0.828	0.896	0confirms
	0	0	0	1confirms
	0.148	0.591	0.237	2confirms
IBk	0	0	0	-1confirms
	0.961	0.983	0.972	0confirms
	0	0	0	1confirms
	0.374	0.204	0.264	2confirms

Table B.27: Accuracy by class for confirms

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1confirms	0	0	0	1	0	0	0	1
	0confirms	0	4399	0	914	1	5221	0	91
	1confirms	0	0	0	0	0	0	0	0
	2confirms	0	110	0	159	0	214	0	55

Table B.28: Confusion matrix for confirms

	Precision	Recall	F-Measure	Class
Naive	0	0	0	-1interpret
	0.961	0.811	0.88	0interpret
	0	0	0	1interpret
	0.163	0.525	0.249	2interpret
IBk	0	0	0	-1interpret
	0.94	0.965	0.952	0interpret
	0	0	0	1interpret
	0.195	0.122	0.15	2interpret

Table B.29: Accuracy by class for interprets

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1interprets	0	1	0	1	0	2	0	0
	0interprets	3	4230	16	967	0	5034	0	182
	1interprets	0	2	0	3	0	5	0	0
	2interprets	1	170	0	189	0	316	0	44

Table B.30: Confusion matrix for interprets

	Precision	Recall	F-Measure	Class
Naive	0	0	0	-1interpret
	0.971	0.877	0.922	0interpret
	0	0	0	1interpret
	0.146	0.444	0.219	2interpret
IBk	0	0	0	-1interpret
	0.959	0.97	0.965	0interpret
	0	0	0	1interpret
	0.154	0.117	0.133	2interpret

Table B.31: Accuracy by class for passing

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1interprets	0	0	0	0	0	0	0	0
	0interprets	0	4676	12	644	0	5173	0	159
	1interprets	0	1	0	2	0	3	0	0
	2interprets	0	137	1	110	0	218	1	29

Table B.32: Confusion matrix for passing

	Precision	Recall	F-Measure	Class
Naive	0	0	0	-1contrasts
	0.98	0.847	0.909	0contrasts
	0	0	0	1contrasts
	0.086	0.456	0.145	2contrasts
IBk	0	0	0	-1contrasts
	0.972	0.979	0.975	0contrasts
	0	0	0	1contrasts
	0.135	0.105	0.118	2contrasts

Table B.33: Accuracy by class for contrasts

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1contrasts	0	0	0	0	0	0	0	0
	0contrasts	0	4584	0	826	0	5295	0	115
	1contrasts	0	1	0	1	0	2	0	0
	2contrasts	0	93	0	78	0	153	0	18

Table B.34: Confusion matrix for contrasts

	Precision	Recall	F-Measure	Class
Naive	0	0	0	-1future
	0.997	0.953	0.975	0future
	0	0	0	1future
	0.008	0.125	0.014	2future
IBk	0	0	0	-1future
	0.997	0.998	0.998	0future
	0	0	0	1future
	0	0	0	2future

Table B.35: Accuracy by class for future

		Classified							
		Naive				IB1			
		-1	0	1	2	-1	0	1	2
Real	-1future	0	0	0	0	0	0	0	0
	0future	0	5308	0	259	0	5557	0	10
	1future	0	0	0	0	0	0	0	0
	2future	0	14	0	2	0	16	0	0

Table B.36: Confusion matrix for future

	Precision	Recall	F-Measure	Class
Naive	1	1	1	01dir
	0	0	0	1dir
	0	0	0	2dir
IBk	1	1	1	01dir
	0	0	0	1dir
	0	0	0	2dir

Table B.37: Accuracy by class for direction

		Classified					
		Naive			IB1		
		0	1	2	0	1	2
Real	0dir	5582	0	0	5580	0	2
	1dir	0	0	0	0	0	0
	2dir	1	0	0	1	0	0

Table B.38: Confusion matrix for direction

Appendix C

Feature Selection Results

No. folds (%)	attribute
10(100 %)	(c)product
10(100 %)	(c)method
10(100 %)	(c)uses
10(100 %)	(c)supports
10(100 %)	indicate
10(100 %)	identified
10(100 %)	Section
9(90 %)	(c)passing
9(90 %)	previously
9(90 %)	call
9(90 %)	VBD
6(60 %)	known to

Table C.1: Correlation with Specific Background

No. folds (%)	attribute
10(100 %)	(c)related
10(100 %)	(c)product
10(100 %)	(c)gbackground
10(100 %)	(c)uses
10(100 %)	(c)confirms
10(100 %)	show
10(100 %)	Section
9(90 %)	(c)dir
9(90 %)	suggesting
9(90 %)	revealed
8(80 %)	as described
7(70 %)	was first
7(70 %)	in vitro
6(60 %)	similar to

Table C.2: Correlation with Data

No. folds (%)	attribute
10(100 %)	(c)uses
10(100 %)	method
10(100 %)	contains
10(100 %)	as described
10(100 %)	EX
9(90 %)	determined
7(70 %)	isolate
7(70 %)	RBS
6(60 %)	revealed
6(60 %)	denoted

Table C.3: Correlation with Method

No. folds (%)	attribute
10(100 %)	(c)uses
10(100 %)	reects
10(100 %)	gift
10(100 %)	full-length
10(100 %)	check
9(90 %)	the following
9(90 %)	in addition to
9(90 %)	group
8(80 %)	in agreement
8(80 %)	direct
7(70 %)	some
7(70 %)	respectively
6(60 %)	appropriate
5(50 %)	propose
5(50 %)	proposed
5(50 %)	RBR

Table C.4: Correlation with Product

No. folds (%)	attribute
10(100 %)	(c)pioneer
10(100 %)	(c)historical
10(100 %)	(c)data
10(100 %)	(c)sbackground
10(100 %)	(c)confirms
10(100 %)	report
10(100 %)	we
10(100 %)	reported
10(100 %)	demonstrated
9(90 %)	study
9(90 %)	is known
8(80 %)	PRP$
7(70 %)	JJR
6(60 %)	abnormally
5(50 %)	suggest
5(50 %)	recent

Table C.5: Correlation with Related

No. folds (%)	attribute
10(100 %)	(c)interpret
10(100 %)	(c)future
10(100 %)	(c)extends
9(90 %)	(c)method
9(90 %)	reproduce
9(90 %)	explanation
9(90 %)	NNPS
9(90 %)	MD
7(70 %)	Context2
6(60 %)	due to
6(60 %)	Years
5(50 %)	specifically

Table C.6: Correlation with concept

No. folds (%)	attribute
10(100 %)	(c)related
10(100 %)	recently
10(100 %)	recent
10(100 %)	novel
10(100 %)	analyze
10(100 %)	Years
9(90 %)	(c)extends
9(90 %)	CD
8(80 %)	interestingly
8(80 %)	full-length
6(60 %)	strongly

Table C.7: Correlation with pioneer

No. folds (%)	attribute
10(100 %)	(c)related
10(100 %)	previously
10(100 %)	originally
10(100 %)	indicated
10(100 %)	earlier
10(100 %)	consist
9(90 %)	approximately
8(80 %)	initially
6(60 %)	investigated
6(60 %)	characterized
5(50 %)	extend

Table C.8: Correlation with Historical

No. folds (%)	attribute
10(100 %)	(c)data
10(100 %)	(c)illustrates
10(100 %)	review
10(100 %)	many
10(100 %)	central role
10(100 %)	NNP
9(90 %)	another
9(90 %)	Context2
7(70 %)	generally
7(70 %)	role
7(70 %)	implicate
6(60 %)	(c)interpret

Table C.9: Correlation with General Background

No. folds (%)	attribute
10(100 %)	(c)product
10(100 %)	(c)pioneer
10(100 %)	(c)data
10(100 %)	(c)sbackground
10(100 %)	(c)supports
10(100 %)	used
10(100 %)	role
10(100 %)	obtained
10(100 %)	member
10(100 %)	mediate
10(100 %)	however
10(100 %)	distinct
10(100 %)	VBZ
10(100 %)	VBP
10(100 %)	VBG
10(100 %)	VBD
10(100 %)	PRP$
10(100 %)	CD
10(100 %)	Section
9(90 %)	(c)method
9(90 %)	(c)illustrates
9(90 %)	like
9(90 %)	find
9(90 %)	describe
9(90 %)	MD
9(90 %)	Figure
9(90 %)	Years
8(80 %)	(c)historical
8(80 %)	required
8(80 %)	many
8(80 %)	in contrast
8(80 %)	done
8(80 %)	appear to
8(80 %)	also
8(80 %)	UH
7(70 %)	some
7(70 %)	furthermore
7(70 %)	carried out
7(70 %)	analyze
6(60 %)	most
6(60 %)	several

Table C.10: Correlation with uses

No. folds (%)	attribute
6(60 %)	more
6(60 %)	isolate
5(50 %)	calculate
5(50 %)	such as
5(50 %)	but
5(50 %)	both
5(50 %)	WRB

Table C.11: Correlation with uses (cont)

No. folds (%)	attribute
10(100 %)	(c)sbackground
10(100 %)	(c)uses
10(100 %)	(c)contrasts
10(100 %)	(c)confirms
10(100 %)	Section
9(90 %)	(c)dir
7(70 %)	classical
6(60 %)	review

Table C.12: Correlation with supports

No. folds (%)	attribute
10(100 %)	(c)concept
10(100 %)	proposed
10(100 %)	model
10(100 %)	might
9(90 %)	(c)pioneer
9(90 %)	(c)supports
9(90 %)	supported
9(90 %)	extend
9(90 %)	could
8(80 %)	may

Table C.13: Correlation with extends

No. folds (%)	attribute
10(100 %)	(c)sbackground
10(100 %)	(c)uses
10(100 %)	(c)passing
10(100 %)	example
9(90 %)	(c)dir
9(90 %)	Section
8(80 %)	as described
7(70 %)	include
6(60 %)	presented
6(60 %)	earlier
5(50 %)	(c)gbackground

Table C.14: Correlation with Illusrates/Clarifies

No. folds (%)	attribute
10(100 %)	(c)supports
10(100 %)	consistent
10(100 %)	results
10(100 %)	observed
10(100 %)	known to
10(100 %)	indeed
10(100 %)	in agreement
10(100 %)	find
9(90 %)	this
6(60 %)	shown in fig
6(60 %)	finds
6(60 %)	appear to
5(50 %)	(c)related
5(50 %)	(c)data
5(50 %)	analogous

Table C.15: Correlation with confirms

No. folds (%)	attribute
10(100 %)	(c)concept
10(100 %)	suggest
10(100 %)	likely
10(100 %)	appear
9(90 %)	required
9(90 %)	may
9(90 %)	indicating
9(90 %)	exhibit
7(70 %)	(c)gbackground
7(70 %)	used
7(70 %)	furthermore
6(60 %)	(c)passing
6(60 %)	unlikely
5(50 %)	potentially

Table C.16: Correlation with interprets

No. folds (%)	attribute
10(100 %)	(c)illustrates
10(100 %)	such as
10(100 %)	like
10(100 %)	include
8(80 %)	(c)interpret
8(80 %)	initially
6(60 %)	can
6(60 %)	NNP
5(50 %)	(c)data

Table C.17: Correlation with passing

No. folds (%)	attribute
10(100 %)	(c)supports
10(100 %)	on the other hand
10(100 %)	in contrast
10(100 %)	however
9(90 %)	(c)related
7(70 %)	(c)interpret
7(70 %)	but
6(60 %)	only
6(60 %)	more
5(50 %)	different

Table C.18: Correlation with contrasts

No. folds (%)	attribute
10(100 %)	(c)concept
8(80 %)	unknown
8(80 %)	requires
8(80 %)	recognized
8(80 %)	finally
6(60 %)	results

Table C.19: Correlation with future

No. folds (%)	attribute
9(90 %)	(c)data
9(90 %)	(c)supports
9(90 %)	(c)illustrates

Table C.20: Correlation with direction

References

[1] Association for Computing Machinery (ACM). http://www.acm.org/.

[2] Duluth. http://www.d.umn.edu/~tpederse/senseval3.html.

[3] Google Scholar. http://scholar.google.ca/.

[4] Java WordNet Library. http://sourceforge.net/projects/jwordnet.

[5] OpenNLP. http://opennlp.sourceforge.net/.

[6] pdftohtml. http://pdftohtml.sourceforge.net/.

[7] Proceedings of the National Academy of Sciences. http://www.pnas.org/.

[8] The Journal of Biological Chemistry online. http://www.jbc.org/.

[9] WordNet. http://wordnet.princeton.edu/.

[10] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

[11] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*, 1998.

[12] Terrence A. Brooks. Evidence of complex citer motivations. *Journal of the American Society for Information Science*, 37:34–36, 1986.

[13] Donald Owen Case and Georgeann M. Higgins. How can we investigate citation behavior? A study of reasons for citing literature in communication. *JASIS*, 51(7):635–645, 2000.

[14] Jie Cheng and Greiner Russell. Comparing bayesian network classifiers. In *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pages 101–108, San Francisco, CA, 1999. Morgan Kaufmann Publishers.

[15] Dale E. Chubin and Soumyo D. Moitra. Content analysis of references: Adjunct or alternative to citation counting? *Social Studies of Science*, 5:423–441, 1975.

[16] Chrysanne DiMarco and Robert E. Mercer. Hedging in scientific articles as a means of classifying citations. In *Working Notes of the American Association for Artificial Intelligence (AAAI) Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*. Stanford University, March 2004.

[17] Brenda Finney. The reference characteristics of scientific texts. Master's thesis, The City University of London, 1979.

[18] Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian Network classifiers. *Machine Learning*, 29(2-3):131–163, 1997.

[19] Nir Friedman and Moises Goldszmidt. Building classifiers using bayesian networks. In *Proceedings of the National Conference on Artificial Intelligence*, pages 1277–1284, 1996.

[20] Eugene Garfield. Can citation indexing be automated? In Mary Elizabeth Stevens, Vincent E. Giuliano, and Laurence B. Heilprin, editors, *Statistical Association Methods for Mechanized Documentation*, volume 269 of *National Bureau of Standards Miscellaneous Publication*, pages 189–192, Washington, December 15 1965. Symposium Proceedings.

[21] Eugene Garfield. *Citation Indexing : its theory and application in science, technology, and humanities*. New York : John Wiley & Sons, 1979.

[22] Mark Garzone and Robert E. Mercer. Towards an automated citation classifier. In *AI '00: Proceedings of the 13th Biennial Conference of the Canadian Society on Computational Studies of Intelligence*, pages 337–346. Springer-Verlag, 2000.

[23] Mark Arthur Garzone. Automated classification of citations using linguistic semantic grammars. Master's thesis, University of Western Ontario, August 1997.

[24] C. Lee Giles, Kurt Bollacker, and Steve Lawrence. CiteSeer: An automatic citation indexing system. In Ian Witten, Rob Akscyn, and Frank M. Shipman III, editors, *Digital Libraries 98 - The Third ACM Conference on Digital Libraries*, pages 89–98, Pittsburgh, PA, June 23–26 1998. ACM Press.

[25] M. A. Hall. *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, University of Waikato, Hamilton, New Zealand, 1998.

[26] David Heckerman. A tutorial on learning with Bayesian Networks. Technical report, Microsoft Research, 1996.

[27] Steve Lawrence, Kurt Bollacker, and C. Lee Giles. Autonomous citation matching. In Oren Etzioni, editor, *Proceedings of the Third International Conference on Autonomous Agents*, New York, 1999. ACM Press.

[28] Steve Lawrence, Kurt Bollacker, and C. Lee Giles. Indexing and retrieval of scientific literature. In *Proc. CIKM99*, 1999.

[29] Steve Lawrence, C. Lee Giles, and Kurt Bollacker. Digital libraries and autonomous citation indexing. *IEEE Computer*, 32(6):67–71, 1999.

[30] Michael E. Lesk. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the Fifth International Conference on Systems Documentation*, pages 24–26, Toronto, CA, 1986. ACM.

[31] David D. Lewis and Marc Ringuette. A comparison of two learning algorithms for text categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 81–93, Las Vegas, US, 1994.

[32] Michael G. Madden. A new Bayesian Network structure for classification tasks. In *AICS '02: Proceedings of the 13th Irish International Conference on Artificial Intelligence and Cognitive Science*, pages 203–208, London, UK, 2002. Springer-Verlag.

[33] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, 1999.

[34] D. Marcu and A. Echihabi. An unsupervised approach to recognizing discourse relations. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, PA, 2002.

[35] Andrew McCallum and Kamal Nigam. Text classification by bootstrapping with keywords. In *ACL99 - Workshop for Unsupervised Learning in Natural Language Processing*, 1999.

[36] Robert E. Mercer, Chrysanne DiMarco, and Frederick W. Kroon. The frequency of hedging cues in citation contexts in scientific writing. In Ahmed Y. Tawfik and Scott D. Goodwin, editors, *Canadian Conference on AI*, volume 3060 of *Lecture Notes in Computer Science*, pages 75–88. Springer, 2004.

[37] Michael J. Moravcsik and Poovanalingam Murugesan. Some results on the function and quality of citations. *Social Studies of Science*, 5(1):86–92, February 1975.

[38] Preslav I. Nakov, Ariel S. Schwartz, and Marti A. Hearst. Citances: Citation sentences for semantic analysis of bioscience text. In *SIGIR'04 workshop on Search and Discovery in Bioinformatics*, 2004.

[39] H. Nanba, N. KANDO, and M. Okumura. Classification of research papers using citation links and citation types: Towards automatic review article generation. In *11th SIG Classification Research Workshop, Classification for User Support and Learning*, pages 117–134, 2000.

[40] Hidetsugu Nanba and Manabu Okumura. Towards multi-paper summarization using reference information. In Thomas Dean, editor, *IJCAI*, pages 926–931. Morgan Kaufmann, 1999.

[41] Kamal Nigam, Andrew K. McCallum, Sebastian Thrun, and Tom M. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39(2/3):103–134, 2000.

[42] C. Oppenheim and S. P. Renn. Highly cited old papers and the reasons why they continue to be cited. *Journal of the American Society for Information Science*, 29(5):227–231, 1978.

[43] S. B. Pham and A. Hoffmann. A new approach for scientific citation classification using cue phrases. In *Australian Joint Conference in Artificial Intelligence*, Perth, Australia, 2003.

[44] Stuart Jonathan Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall series in artificial intelligence. Prentice Hall, Upper Saddle River, 2nd edition, 2003.

[45] Ina Spiegel-Rosing. Science studies: Bibliometric and content analysis. *Social Studies of Science*, 7(1):97–113, February 1977.

[46] Simone Teufel, Advaith Siddharthan, and Dan Tidhar. An annotation scheme for citation function. In *7th SIGdial Workshop on Discourse and Dialogue*, pages 80–87, Sydney, July 2006. Assosiation for Computational Linguistics.

[47] Simone Teufel, Advaith Siddharthan, and Dan Tidhar. Automatic classification of citation function. In *2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, pages 103–110, Sydney, July 2006. Assosiation for Computational Linguistics.

[48] Melvin Weinstock. Citation indexes. In *Encyclopedia of Library and Information Science*, volume 5, pages 16–40. Marcel Dekker Inc., New York, 1971.

[49] Stephen F. Weiss. Learning to disambiguate. *Information Storage and Retrieval*, 9(1):33–41, 1973.

[50] Howard D. White. Citaton analysis and discourse analysis revisited. *Applied Linguistics*, 25(1):89–116, 2004.

[51] Ian H. Witten and Eibe Frank. *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann, San Francisco, San Francisco, 2nd edition, 2000.

[52] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Meeting of the Association for Computational Linguistics*, pages 189–196, 1995.