AN EVALUATION OF THE INFLUENCE OF A DOCUMENT'S TEXT-TYPE ON THE USE OF DISCOURSE RELATIONS

Félix-Hervé Bachand

A thesis
in
The Department
of
Computer Science & Software Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Computer Science
Concordia University
Montréal, Québec, Canada

September 2014

Concordia University
School of Graduate Studies

This is to certify that the thesis prepared

By: **Félix-Hervé Bachand**

Entitled: **An Evaluation of the Influence of a Document's Text-Type on the Use of Discourse Relations**

and submitted in partial fulfillment of the requirements for the degree of

Master of Computer Science

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining commitee:

Dr. Brigitte Jaumard _____ Chair

Dr. Sabine Bergler _____ Examiner

Dr. Olga Ormadjieva _____ Examiner

Dr. Leila Kosseim _____ Supervisor

Approved _____
Chair of Department or Graduate Program Director

_____ 20 _____ _____

A. Asif, Ph.D. Dean
Faculty of Engineering and Computer Science

Abstract

An Evaluation of the Influence of a Document's Text-Type on the Use of Discourse Relations

Félix-Hervé Bachand

In this thesis, we will discuss the work we have conducted on the relationship between discourse relations in English documents and their associated text-types. Obtaining an understanding of the text-type of a given document is a step towards identifying its larger discourse schema which, in turn, is instrumental in effectively identifying discourse relations. In order to study the relationship between discourse relations and discourse structures, and the text-type of a document, we have created a corpus of documents belonging to seven distinct text-types, from which we extracted discourse relation annotations using already existing parsers. Utilizing the data obtained, we have studied various ways in which discourse relations and text-types are linked in an effort to better understand how discourse schemas can be identified and subsequently utilized in the automatic extraction of discourse relations. Our experiments have shown that the classification of documents within our seven text-types is still better performed with a bag-of-words approach, but the results obtained with the automatically extracted discourse relations suggest that there is in fact a link between text-types and the use of specific discourse relations. We also found that the various text-types are identified with varying accuracy, with text-types such as *explanation* and *report* being harder to identify, regardless of the methods used. Finally, our results also show that the cue phrases used to identify *explicitly* stated discourse relations are amongst the more informative features of our better performing bag-of-words model, and can be utilized to reduce the feature space of this particular model.

Acknowledgments

The thesis presented here is the result of two years of studies and research during which I was privileged to work and collaborate with a number of friendly, helpful, and knowledgeable people. I must first express my most sincere thanks to my supervisor, Dr. Leila Kosseim, who I believe was throughout this degree, and beyond, more than helpful. Her excellent feedback and guidance at every step of the process which resulted in this thesis have been invaluable. At the conclusion of these two years of work, I believe I have learned a great deal from this excellent researcher and teacher.

In addition, I wish to extend my thanks to the my examining committee: Professors Sabine Bergler and Olga Ormandjieva, as well as the chair: Professor Brigitte Jaumard. I wish to extend particular thanks to Dr. Bergler whose interest in my research subject has allowed for many helpful discussions during my time within the department.

Finally, I wish to thank the various members of the CLaC lab who have always been more than willing to lend a helping hand. My work within the lab would certainly not have been as pleasant if it was not for the excellent members that accompanied me on this journey.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Consider the simple discourse: *Obtaining a Master's degree takes time, I have taken two years to complete mine.* In a coherent text, textual units are not understood in isolation but in relation with each other through discourse relations that may or may not be explicitly marked. The fact that "I" have taken two years to complete my Master's *illustrates* that obtaining such a degree takes time. Research on discourse analysis tries to model the coherence relations that hold between textual units, and these allow us to interpret the text and understand the communicative purpose of its units. This, in turn, is useful for many Natural Language Processing (NLP) applications such as automatic summarisation, question answering and text simplification. The objective of our work is to uncover ways that would allow for better performing automatic extraction of discourse level rethorical structures.

The task of automatic discourse relation extraction is a particularly difficult one. One important difficulty stems from the need for the system to be aware of the rhetorical purpose of the discourse on several levels. The rhetorical structure of a document can be divided into several levels of abstraction, from the general, down to the more specific. Discourse parsers available today (eg. (Soricut & Marcu, 2003; Hernault *et al.* , 2010a; Feng & Hirst, 2012; Lin *et al.* , 2012; Faiz & Mercer, 2014)) attempt to extract rhetorical relations between Elementary Discourse Units (EDUs) without trying to build structures to the highest level of discourse relations schemas. The notion of schema is based on the description of the Rhetorical Structure Theory described in (Mann & Thompson, 1987). For our purpose, we consider the highest level of abstractions related to rhetorical structures: the **text-type**. We argue that in order to extract discourse relations effectively, a system should consider the higher level rhetorical structures that we describe here as text-types, or at the least, that the text-type provides some indications on the highest level of discourse structures which are useful to the overall extractions of the rhetorical structures studied. By text-types we mean that texts can have a variety of communicative goals (Swales, 1990). Examples of text-types include: instructional texts, reviews, reports, etc. Our claim is that each text-type makes a particular usage of discourse schemas. When dealing with a document from a particular text-type, we expect that the usage of

discourse relations should be similar to those seen in other documents of the same text-type, while documents of different text-types should vary more widely. This is in line with the hierarchical view of discourse analysis presented in (Mann & Thompson, 1987). Schemas, which can be seen at a higher level as being related to text-types are therefore important features to consider when performing automatic extraction of discourse relations. In our work, we attempted to construct a corpus that contains documents that share the same text-type, while varying as much as possible in genre[1]. This was done in order to argue that the structure of the text itself should suffice in identifying the text-type, no matter what specific vocabulary is used, which we believe to be defined by the genre. For example, a newspaper article, which we would classify as a document of the *recount* text-type, could be political in nature, or describe a sporting event. It should not matter which of these two genres is at play. Instead we are mainly interested in how the clauses created from these terms are related to each other through discourse structures. It is our belief that discourse structures should be studied at various levels, from the more abstract concept of a document's overall schema, described here as the text-type, down to the relationship between single clauses composed of phrasal structures, in order to fully understand textual structures on a semantic level.

Currently, most work on the automatic extraction of discourse relations focuses on the lowest level of discourse relations, that is, discourse relations holding between two clauses. We believe, however, that an important part of obtaining a semantic understanding of documents on a structural level requires the structures to be extracted on several levels, from the lowest level to the highest, namely the text-type of a given document. Our claim is that by first identifying the highest discourse structure of a text, we can subsequently improve the identification of the finer grained discourse relations which hold between the various clauses that make up the document. Overall, the question this thesis attempts to answer is whether the identification of text-types can be used as a first step towards detecting larger discourse schemas which in turn should improve the extraction of the lower levels of the overall discourse structure of a document. For example, knowing that a document describes a procedural discourse, we would expect certain large schemas to occur, such as a list of items required for the procedure, and the steps of the procedure itself.

1.2 Key Linguistic Concepts

1.2.1 Syntactic Structures

With the first publication, in 1957, of Noam Chomsky's work on syntactic structures (Chomsky, 1957), the scientific field of linguistics as we know it today was created (Chomsky, 2002). An important point introduced by Chomsky is the distinction between semantics and syntax. With the now famous example of the sentence "Colorless green ideas sleep furiously", he demonstrates that grammatically correct sentences do not equate sense or meaning. In other words, the sentence itself makes a correct use of syntax, but it does not provide any intelligible semantics. Still, syntactic structures are employed in our everyday usage of natural languages in order to convey the ideas and

[1] See Section 1.2.3 for a more formal description of these concepts.

concepts we wish to communicate. The syntactic structures we create, as we communicate, link concepts together by forming larger and larger structures which eventually form complete presentations of our ideas. Consider the syntactic tree structure of Figure 1.1 created by parsing the example sentence:

(1) I like this food because it is tasty.

with the Stanford parser (De Marneffe *et al.* , 2006). The Stanford parser uses the Penn Treebank corpus (Marcus *et al.* , 1993) set of tags, which are listed in Appendix A.

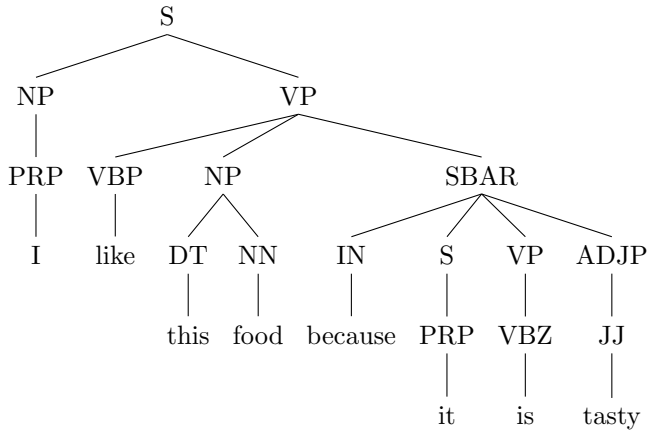

Figure 1.1: Example Syntax Tree Generated Using the Stanford Parser (De Marneffe *et al.* , 2006)

Figure 1.1 shows that the sentence provided as an example can first be broken down into a noun phrase (**NP**) and a verb phrase (**VP**). The **NP** simply contains the pronoun "I", which becomes the subject that attaches to the **VP**. Further down the tree, we find that the object of the **VP** is realized as an embedded **NP**. For our present purpose, however, we are mainly interested on how the two spans of text "I like this food" and "because it is tasty" are connected to each other. We will see in more detail, in Section 2.3.2.1, how the information provided by such a syntax tree can help us better identify similar structures (tree structures as with the syntactic ones), but at the discourse level. For now, suffice to say that the appearance of the **SBAR**[2] structure within the **VP** is interesting considering that it happens to occur at the beginning of a span of text that denotes causality.

1.2.2 Semantics

The study of semantics focus on the concept of *meaning*. In order to understand a text, we must first understand its parts. Because of this, semantics is interested in meaning on several levels:

1. word level

[2]the concept of $SBAR$ comes from x-bar theory presented in Jackendoff (1977) and serves to create a new syntactic structure containing the Sentence node and which can be extended with specifiers. That is, the $SBAR$ is the root of an embedded clause with a sentential structure.

2. phrasal level

3. discourse level

For the purpose of our current research, we are interested in discourse level semantics. That is, the meaning conveyed by a combination of clauses. While a clause is understood as the smallest possible portion of a discourse that expresses an idea that can stand on its own, the combination of such clauses creates more complex meaningful units we call discourse structures. In order to gather some understanding of this higher level of semantics, however, we must rely on our understanding of the smaller parts. In order to better explain how these various levels of meaning are used, consider again the following simple sentence:

(2) I like this thesis because it is interesting.

If we first consider the meaning of the words of the sentence, independently of each other, several interpretations could be found for some of these words. Much like the higher level phrasal structures, the words themselves can be composed of several parts. Take for example the word "interesting", which is itself the noun "interest" with an affix "-ing" which turns the original noun into an adjective. However, the words, and the parts of which they are composed, are sometimes insufficient to produce an accurate interpretation of the meaning that is being communicated. For example, the word "like", without any more context could be understood as a preposition meaning "similar to" or as a verb to signify an affinity for something. If we take the context surrounding the word, we can look into the phrase structure in which the term "like" is used. By using this specific term with the first person pronoun to produce "I like...", the meaning becomes clearer. Using the term "like" to mean "similar to" in such a context is ungrammatical and therefore rejected as a possible meaning of the word. Moving further up in our structures, if we look into the constituents where the term is found, we see, as mentioned in Section 1.2.1, that the verb phrase headed by "like" embeds another constituent, the **SBAR** from the right-hand side of the syntax tree of Figure 1.1. As we have mentioned in Section 1.2.1, we are interested in the relation between both phrasal structures which in turn form a complete sentence. The way by which such phrasal structures are related to each other form a larger entity which we can describe as the discourse structures. We will call these larger phrasal structures Elementary Discourse Units (EDU). Once these phrasal structures are joined into larger discourse structures, the EDUs, these EDUs relate to each other, thus creating even larger discourse structures. Eventually, all of the EDUs of a document join together to form the overall discourse schema. Much like how words are composed of smaller parts, and phrasal structures are composed of words, which themselves turn to larger phrasal structures or sentences, a discourse is composed of these discourse level constituents, or EDUs, which are connected to each other. Similar to syntactic trees, we can build tree structures that show the relation holding between these EDUs. Figure 1.2 depicts such a representation of discourse level schemas. Our present work is concerned with the existence of the relations holding between EDUs, namely *attribution* and *enablement* in Figure 1.2.

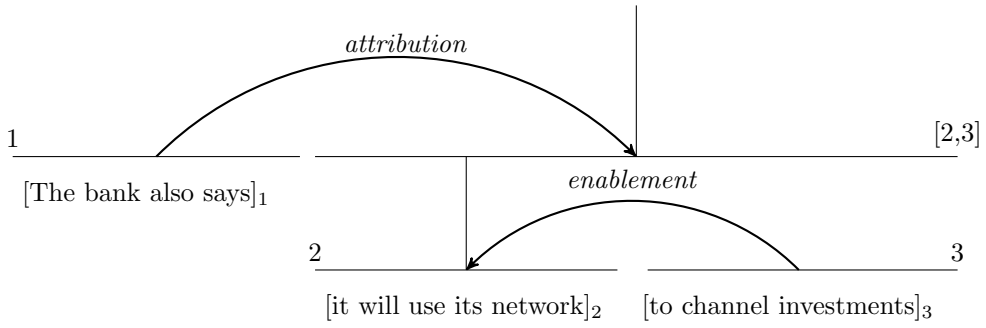

Figure 1.2: Example of Three EDUs and Their Relationships Using the RST Framework (Soricut & Marcu, 2003)

1.2.3 Text Types vs. Genres

The notions of **text-type** and **genre** are highly related, and often seem to be the source of some confusion. For this reason, we feel the need to define both concepts in more detail at this point. According to (Lee, 2001), **text-type** is defined as the communicative purpose of the document, and how that purpose is achieved through linguistic constructs that are detectable at the discourse level. Examples of **text-types** are: procedure, recount, narrative, etc. On the other hand, **genre** takes the definition of the subject matter of the document. As such, a document has a particular purpose, identified by its **text-type**, and deals with a particular subject, identified by its **genre**. For example, different **genres** could be: food, biology, politics, sports, etc. To define intuitively these two concepts, consider the following simple examples: consider two narrative texts, each detailing the adventures of a hero of their own respective genre. A very common way of building narratives is based on a three act structure (Lavandier, 2007). The first act typically depicts the hero content with his situation, until tragedy strikes. The second act shows the hero's fall following this tragedy and struggle to recover his position. The final and third act shows the hero's final battle against adversity and subsequent victory. Using this same basic template, it does not matter if the hero is an astronaut, as befitted for science-fiction genre, a cowboy, as expected of an adventure narrative, or a private detective, typical of mystery stories, the same basic structure can be used. The genre will define what type of hero the narrative calls for (e.g. science-fiction, adventure narrative, etc.), while the text type will define the overall structure of the document (i.e. our three acts structure.) As another example, compare two procedural texts on different genres: a cooking recipe, and instructions on assembling a piece of furniture. The first of these two will typically start by giving a list of ingredients required for the completion of the recipe, while the second will provide a list of parts that should be packaged in order to assemble your brand new piece of furniture. Typically, the recipe will then provide a list of steps on how to mix the ingredients, while the instruction manual will similarly give a series of steps to be performed in the assembly process. Again, the genre defines the type of tools that are to be listed (e.g. ingredients, parts, etc.), while the text type provides a template composed first of a list of those tools followed by the steps requiring those tools in order to obtain the final product.

David Lee's attempt at explaining this distinction (Lee, 2001) is used as the basis of the construction of our working corpus. The important distinction noted by Lee which distinguishes both concepts are *internal* and *external* criteria. The *external* criteria are expected to give insights on the genre of a given document. These criteria include: intended audience (e.g. a research paper is intended to be read by scholars of a specific field), purpose (e.g. the same research paper would serve to present experiments and findings), and activity type, (e.g. if the experiments were properly performed in a lab with specific guidelines or done more informally as an early investigation on a subject). It is expected that these criteria will manifest themselves through the usage of specific vocabulary, rather than through the form of a given document. For example, the term "bread" might appear in a restaurant review or a cooking recipe. Both documents are within the same **genre**, but differ in their **text-type** (the first is of the *recount* text-type, and the second is of the *procedure* text-type. More detail on the various text-types we considered can be found in Section 3.1). What we mean by this is that in essence, the genre is determined by the specialized vocabulary used. For example, given two documents, both fictional narratives, one in the adventure genre, the other in the science-fiction genre, it seems more likely that the word "stampede" would occur in the first of these documents, while "spaceship" seems more appropriate in the latter. On the other hand, the form of a document is related to the *internal* criteria. These are described as the linguistic characteristics of a text. For our purpose, we consider that discourse structures should form an adequate basis for this classification and therefore consider this as our *internal* criteria.

For the purpose of our thesis, we created a corpus that contains documents from various **text-types**, with various **genres** covered within these different **text-types**. What we wish to show through this is that the vocabulary used in these documents is generally shared within **genres**, while the structures are shared within **text-types**. As such, we hope that the **genres** are represented widely enough within the various **text-type** classification so that it will no play a role in the classification tasks performed, putting more importance on the discourse structures associated with **text-types** instead. Although some corpora do exist with discourse level annotations, we found that these were insufficient to properly gather the type of data needed to investigate the phenomenas at hand, for this reason we opted to build our own corpus.

1.3 Methodology and Results

In order to evaluate our claim that text-types are closely related to discourse relations and can therefore be used in the process of automatic extraction of those discourse relations, we have designed a classification task. In order to do achieve this, we have put together a corpus comprised of 3,769 documents from seven different text-types. We then extracted information related to discourse relations using two automatic discourse relation parsers currently available (Lin *et al.* , 2012; Faiz & Mercer, 2014). Using the information obtained from the outputs of these parsers, we attempted to classify the documents into our seven text-type categories (exposition, explanation, recount, report, response, procedure, narrative), using ten-fold cross validation. We have performed the classification task using three well known classifier algorithms, namely Multinominal Naïve Bayes,

Decision Trees, and Support Vector Machines. With each of these classifiers, we have obtained results of our text-type classification task using a number of features sets: bag-of-words, explicitly stated discourse relations, implicitly stated discourse relations, both types of discourse relations at once, discourse connectives, and all discourse relations and discourse connectives. The results obtained from these various experiments show that, although the baseline produced by studying the documents through the bag-of-words model produces the best overall results, the much more computationally manageable approach (in the sense that smaller feature spaces involve less factors to include in our calculation) of concentrating on discourse relations and their related vocabulary items (i.e. discourse connectives) allow for results that point to a relationship between text-types and discourse relations.

1.4 Contributions

A number of practical and theoretical contributions are made within this thesis:

1. the creation of a corpus of documents classified across seven text-types,
2. the creation of evaluation methods that can be used in classification tasks based on this corpus,
3. insights on the relation between the highest level of discourse structures (i.e. the text-type) and the lower level of such discourse structures (i.e. the relations between clauses),
4. the evaluation of a number of features for the automatic classification of documents according to the overall schema of the document.

In this chapter, we have introduced the motivation behind our work, provided some context related to the linguistic concepts that are important to our research, and introduced the methodology used and the results obtained related to our claims, and we have outlined the scientific contributions presented by this thesis. The remainder of this thesis is organized in four Chapters. Chapter 2 covers some of the previous work published on the subject of discourse relations. We begin by presenting the theory of discourse rhetoric from the level of discourse units and how such units related to one another, thus creating discourse structures. We then discuss some of the resources currently available that were helpful to us in our research. Namely, we discuss the various corpora annotated for discourse relations, as well as the different discourse relation parsers currently available. Finally, we discuss some of the work currently available through various publications which pertain to the subjects of discourse rhetoric and text-types. In Chapter 3, we provide a detailed explanation of the steps undertaken in order to build and perform our experiments. To do so, we begin by describing the process of building our corpus by using a number of corpora that are currently available, secondly, we discuss briefly the discourse relations framework we have decided to use for our experiments, we then describe how we used this framework to annotate our corpus for discourse relations, and finally, we discuss the various classification tasks performed by enumerating and describing the various feature

sets and classifiers used. In Chapter 4, we present the results of our experiments. We begin by providing a short overview of the final results of the various experiments. We then go into further details by studying the results obtained with each of our feature sets. We finally provide further discussions on the most informative features uncovered throughout our experiments. Finally, in Chapter 5, we provide a discussion of the findings provided by the analysis of our experiments in relations to the influence of the feature sets used, the influence of the classifiers used, the variations in performance observed across text-types, and the reasoning behind the most informative features observed. We conclude by discussing the avenues that now present themselves to us as possible future projects.

Chapter 2

Previous Work

In this chapter, we explore the various research efforts that have been published in the past in relation to the problems we have undertaken to address. We begin in Section 2.1 by defining some of the commonly used metrics that will become useful in subsequent chapters. In Section 2.2, we give an account of the theories of discourse rhetoric and describe the frameworks that have been used for the purpose of studying these. We then provide further details, in Sections 2.2.1 and 2.2.2, about some of the theories used to describe discourse rhetoric, more specifically, the base unit of discourse structures used, and how these base units can be used to form an intelligible picture of discourse structures. In Section 2.3, we explore the resources currently available for the purpose of our experiments. These include, in Section 2.3.1, an overview of the corpora available with annotation pertaining to discourse relations, and in Section 2.3.2, an overview of the state if the art parsers available for the automatic extraction of discourse relations. In Section 2.4, we give an overview of previously published works which deal with some of the problems related to text-types and discourse rhetoric.

2.1 Metrics Used

Several metrics are used to describe various implementations and classification tasks throughout this thesis. We will describe the more commonly used metrics here, while less common metrics will be described when appropriate within the thesis. For now, we provide a description of the following metrics: *precision*, *recall*, *accuracy*, and *F-score*.

In order to understand the metrics used for the evaluation of the various systems presented in Chapters 2 and 4, we must first describe a few key concepts. During a task such as classification, four outcomes are possible for every item that is classified: *true position*, *false positive*, *true negative*, and *false negative* (Olson & Delen, 2008). If an item is correctly classified, it is considered a *true positive*, while if an item is wrongly classified, it is considered a *false positive* for that class. Likewise, if an item is correctly identified as **not** being part of a class, it is considered a *true negative*, while an item being wrongly associated with a class is considered a *false negative*. Consider the simple data presented in Table 2.1.

Class	Expected	True Positive	False Positive	True Negative	False Negative
A	10	8	8	12	2
B	10	5	4	16	5
C	10	3	2	18	7

Table 2.1: Example Distribution for a Classification Task

Each class contains 10 items. For class A, we correctly identify 8 items; for class B, 5 items are classified correctly; and 3 of class C's items are classified correctly. These values are our *true positive*. The *false positive* column indicates the number of items of a certain class that is categorised as a different class. For example, class A has 8 *false positive* instances. This could mean, for example, that 3 items of class A were classified as class B, and 5 more items of class A were classified as class C. The sum of all *true positives* and *false positives* is the total of all items, which in our example is 30 items. If an item is not classified as a certain class, and it is not expected to be classified as such, it counts towards *true negative* values. For example, our example shows that 12 items are *true negative* for class A. This means that 12 items of class B or C were **not** classified as A. If, on the other hand, an item is expected to be a different class but is classified of that class anyway, it is a *false negative* value. The count of *false negative* values for class A is of 2, meaning that two items that should have been classified as A were misclassified as either B or C.

Out of these values, we can obtain our first three metrics: *precision*, *recall*, and *accuracy*. *Precision* represents the number of items classified that are relevant, that is, how many of the items were classified correctly. In order to calculate this metric, we simply apply the formula presented in Equation 2.1.

$$precision = \frac{\sum true\ positive}{\sum true\ positive + \sum false\ positive} \tag{2.1}$$

Given the example of Table 2.1, we find that 16 items are *true positives* $(8 + 5 + 3)$. We obtain the *precision* simply by dividing this value by the total number of items in our population, which is 30, or $(10 + 10 + 10)$. Therefor, the *precision* of our example is $(16/30) \times 100 = 53.\bar{3}\%$.

Our second metric is the *recall*, which represents the number of items that were correctly identified in respects to how many were expected to be identified. In order to obtain this metric, we use the formula presented in Equation 2.2.

$$recall = \frac{\sum true\ positive}{\sum true\ positive + \sum false\ negative} \tag{2.2}$$

Once again, the example of Table 2.1 shows a total of 16 *true positive*. The sum of *true positive* and *false negative* gives us a total of 30. Applying the formula provided in Equation 2.2, we obtain a *recall* score of $53.\bar{3}\%$.

Accuracy is simply the number of correctly classified values, whether it be correctly identified as being of a certain class or as correctly identified as **not** being part of a certain class, over the total number of values and can be obtained with the formula presented in Equation 2.3.

$$Accuracy = \frac{\sum true\ positive + \sum true\ negative}{\sum true\ positive + \sum false\ positive + \sum true\ negative + \sum false\ negative} \quad (2.3)$$

In the case of the example given in Table 2.1, we would obtain a total of 16 *true positive*, a total of 46 *true negative*, and total of 90 (which is the sum of all *true positive*, *false positive*, *true negative*, and *false negative*). The final *accuracy* measure is then 69%.

Finally, a last commonly used metric is the *F-measure*, or *F-score*. Such a metric combines both the *precision* and *recall* measures typically through an harmonic mean. Typically, the simple F_1*-score* is used to evaluate systems and classification tasks. With the *F-score* metric, it is possible to give more weight to either *precision* or *recall*. Using the F_1*-score*, we give equal amount of weights to both of these metrics. Equation 2.4 presents the formula used to obtain the F_1*-score* (note that throughout the rest of this thesis, all F-scores should be assumed to be the F_1-score).

$$F = 2 \times \frac{precision \times recall}{precision + recall} \quad (2.4)$$

Using this formula, we can obtain the F-score of the data presented in Table 2.1. Our precision and recall were both calculated at approximately 0.53. We can simply divide the product of these two values by the sum of these same values, and obtain an *F-score* of 0.265.

2.2 Theories of Discourse Rhetoric

The idea of discourse rhetoric is an important one when it comes to parsing a document in an attempt to understand it. In fact, an interesting problem with semantics is the apparent need to understand the documents on several levels, such as discourse types, topics, phrasal structures, and lexical items. What makes this problem particularly challenging is the interdependance of these various levels. For example, the meaning of particular words are better identified within their context, as John Rupert Firth famously stated: "you shall know a word by the company it keeps" (Firth, 1957). The problem here is that each word is thought to find its meaning based on the context and the context itself is defined by the combination of each word. Following a similar logic, we believe that semantics at the discourse level functions in a similar fashion. A document holds a specific intended meaning, but this meaning can be identified by looking at smaller portions of the documents, such as sections or topics. By sections or topics, we mean the various portions of the document, such as an abstract, a methodology section and a discussion section in a scientific article. These sections themselves can be broken down into smaller portions, each having a specific rhetorical purpose. For the time being, let us simply think of those smaller portions as the phrasal structures of sentences. These phrasal structures are typically related to one another which allows us to better identify their rhetorical meaning based on context, much like with lexical semantics. For example, consider the following simple statement:

(3) I enjoy my work because I find it fulfilling.

The first part of this sentence, "I enjoy my work", is supplemented by the second, "because I find it fulfilling", through a *causality* relation. Being able to identify such rhetorical relationships not only allows us to identify which phrasal structures are related to one another, but also gives us information on how these are related. In turn, this information allows us to build larger structures which will, once again, be related and contextualized within a greater scheme. Based on these observations, a number of researchers, notably (Mann & Thompson, 1988) have attempted to build frameworks to be utilized in the study of such problems. Bill Mann, Sandy Thompson, and Christian Mathiessen, of the University of Southern Carolina, have been working on the problem of the computational approach to discourse rhetoric since 1983.

2.2.1 Elementary Discourse Units

In order to discuss how discourse is structured, we must first segment the discourse into units which can be related to one another. A first choice for such units might be sentences, but using such a basis, we quickly realize that discourse structures can exist within sentences themselves. A more fine grained analysis could then make use of phrasal structures, but (Mann & Thompson, 1988) find this to be insufficient as well. For this reason, rather than limiting ourselves to phrasal structures, texts can be understood in terms of Elementary Discourse Units (EDU). EDUs cover spans of text which serve a specific rhetorical purpose. In their description of such units, (Mann & Thompson, 1988) claim that their sizes are arbitrary. The constraint they rely on instead is that each of these units should hold independent functional integrity, that is, each unit should hold a particular independent meaning which can be understood from the unit alone. For their purpose, these authors simply state that the basis of their units is essentially clauses. On the other hand, identifying discourse units has been noted time and time again to be a rather difficult task. Daniel Marcu of the *Information Science Institute* at the University of Southern California, and his research team, have been studying the problem of discourse rhetoric since the late 1990's. In the first portion of the annotation guidelines of Discourse Structure by (Marcu, 1999), this difficulty is attributed to the fact that the boundaries between syntactic, semantic and rhetorical information is blurry. This difficulty makes it so that properly identifying discourse units can easily become a matter of controversy. Fortunately, there is some level of agreement to be found. The idea of clauses being the most frequent basis for discourse units is shared by Mann and Thompson, and Marcu. In Marcu's guidelines, it is stated that fully fleshed clauses are always discourse units. The minimal requirements for a span of text to be considered a clause is to contain a verb and its obligatory argument. The following example, borrowed from the guidelines, demonstrates this:

(4) [As the play ended,][the body would return to its reliance on the aerobic metabolic system][while the capacities of the other energy systems regenerated themselves.]

This example features three discourse units separately bracketed. Each of these units contain a clause with a verb and their associated obligatory argument (as stated by (Marcu, 1999)). A series of 12 rules are also described in the guidelines in order to properly identify EDUs (Marcu, 1999). It should be noted, however, that some of these rules allow for EDUs to be identified even in cases

where they are not fully fledged clauses. For example, titles and headings are to be treated as EDUs by (Marcu, 1999), restrictive and non-restrictive modifiers (modifiers that add information that is not essential to the clause it modifies) that use a finite verb are embedded EDUs, appositive clauses using finite verbs are also embedded EDUs, and so on. For the purpose of understanding the theory presented here, understanding EDUs as being essential clauses, should be sufficient. For essential clauses, the description provided by (Marcu, 1999) explains such clauses convey a single idea and can stand on their own, without the need to be associated with another.

2.2.2 Discourse Theories

As we have mentioned in Section 2.2, our understanding of a text relies on the fact that EDUs are related to one another in some manner. In order to achieve some sort of coherence, authors put together several EDUs which are logically linked in some fashion or other, forming a single coherent structure. Detecting discourse relations has become an important step in many Natural Language Processing (NLP) tasks over the years. Several applications, for instance summarization and machine translation systems, rely on the automatic extraction of these Elementary Discourse Units and their relationships. For example, (Marcu, 2000) describes such a system for the purpose of text summarization and (da Cunha & Iruskieta, 2010) study discourse structures in relation to machine translation. However, there exists different theories of discourse structures. In this section, we provide an account of the *Rhetorical Structure Theory* of (Mann & Thompson, 1987, 1988) and the *Penn Discourse Tree Bank Framework* described by (Prasad *et al.* , 2007, 2008) while noting the current state and use of both of these approaches.

2.2.2.1 Rhetorical Structure Theory

A number of observations are made by (Mann & Thompson, 1987) leading to the creation of the Rhetorical Structure Theory (RST). (Mann & Thompson, 1987) first find that texts are constructed through hierarchically organized clauses related to one another in various ways. They also note that the most common of these organizations is the one they decided to call the *nucleus-satellite* relation. Such an organization is described as follows: given two distinct non-overlapping text spans, the *nucleus* and its *satellite*, a relation can be noted as the source of the coherence between the two. For example, a span of text A can be noted as denoting evidence for the claims from a span of text B. They observe that such relations are for the most part asymmetric. That is to say, the fact that text span A is evidence for the claims of text span B does not mean that B is evidence to A. Because of this, it can be assumed that certain text spans are to be considered more central, thus creating a hierarchy within the relations observed. Once relations can be observed, the authors then define a **schema** as sets of relations. The simplest possible schema is a single relation, while the most complete schema is the set of all relations found throughout the text. For example, the first half of this sentence forms a simple schema, while the entire thesis, composed of clauses organized in a specific fashion form the highest schema. With these basic principles, Mann and Thompson suggest that all texts can be described within this framework. The schemas which are implicitly perceived by human readers serve as identifying the various functions of the observed text spans.

To better clarify what relations and schemas are exactly, consider the following examples borrowed from Daniel Marcu's annotation guidelines (Marcu, 1999). Our first example denotes an *antithesis* relation between two elementary units:

(5) [He tried hard,][but he failed.]

The leftmost EDU presented here is the *nucleus*, while the rightmost EDU is the associated *satellite*. It should seem obvious and intuitive to readers how the two units are linked to each other showing that there is in fact a coherence that exists between such EDUs. The *nucleus* EDUs are generally capable of standing by themselves, regardless of the presence of their associated *satellite* EDUs. On the other hand, these associated *satellite* EDUs do not need to make sense by themselves. In other words, the core idea expressed by the *nucleus* is independent of its *satellite* which itself serves to provide further details related to the core idea of the *nucleus*. This can be observed by simply changing the *satellite* of our example:

(6) [He tried hard,][but he came in second.]

The core idea remains the same, but the added information from this new *satellite* provides different details to our *nucleus*. It should also be noted that relations are hierarchical in nature. For example, consider the following graphical representation of three EDUs and their relations from (Soricut & Marcu, 2003):

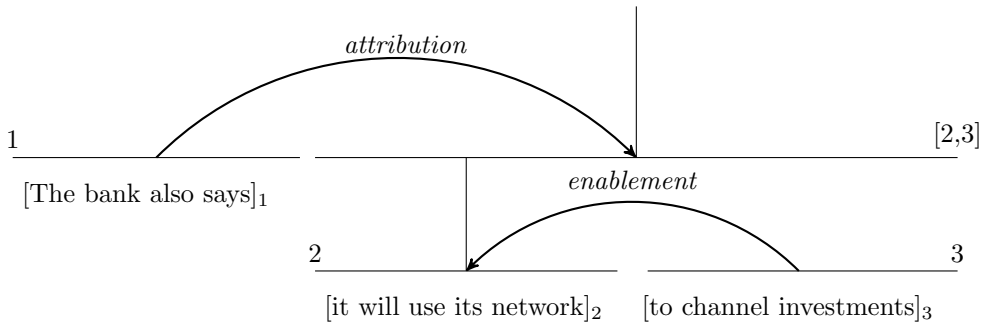

Figure 2.1: Example of Three EDUs and Their Relationships Using the RST Framework (Soricut & Marcu, 2003)

In this example, we find two relations. The *enablement* relation holds between EDU_3, the *nucleus*, and EDU_2, its *satellite*. The other relation represented here, *attribution*, holds between EDU_1, in this case the *nucleus*, and the projection of EDUs 2 and 3, which are now the *satellite*. As can be seen graphically, there is a hierarchy between these relations since the *enablement* relation links EDUs 2 and 3 together while the *attribution* relation links EDU_1 with the combination of the remaining EDUs and their relation.

Relations, within the RST framework, were originally grouped into 3 classes (Mann & Thompson, 1987). We provide here these 3 classes with their associated relations and a short description on why they are grouped together.

Presentational Relations (Antithesis, Background, Concession, Enablement, Evidence, Justify, Motivation, Preparation, Restatement, Summary)
With these relations, the *satellite* EDU presents new information about the *nucleus* EDU.

Subject Matter Relations (Circumstances, Condition, Elaboration, Evaluation, Interpretation, Means, Non-volitional Cause, Non-volitional Result, Otherwise, Purpose, Solutionhood, Unconditional, Unless, Volitional Cause, Volitional Result)
With these relations, the *satellite* EDU provides some context on the subject matter of the *nucleus* EDU.

Multinuclear Relations (Conjunction, Contrast, Disjunction, Joint, List, Multinuclear Restatement, Sequence)
With these relations, multiple EDUs are linked without putting the focus on any particular one.

For our purpose, however, RST's set of relations has been somewhat expanded. This expanded version forms the basis of the annotation guidelines used to produce the RST Discourse Treebank corpus (Carlson *et al.* , 2002). The guidelines described in (Marcu, 1999) will be used as the basis of our description of what is referred to as the RST framework. Another competing set of guidelines, following a very similar approach to the same problem but resulting in a different set of discourse relations, was created by, and is described in (Prasad *et al.* , 2007, 2008). We describe both of these frameworks in the following section.

2.3 Resources

2.3.1 Manually Annotated Corpora

A number of corpora manually annotated for discourse relations have been produced in the past few years. Two frameworks have typically been used in the creation of these corpora, the Rhetorical Structure Theory framework (RST), and the Penn Discourse Tree Bank framework (PDTB). Each of these frameworks make a number of assumptions that differ, but the overall idea of discourse relations is shared between the two. Sections 2.3.1.1 and 2.3.1.2 describe corpora using the RST framework, while Sections 2.3.1.3 and 2.3.1.4 describe corpora using the PDTB framework.

2.3.1.1 Rhetorical Structure Theory - Discourse Treebank

The Rhetorical Structure Theory - Discourse Treebank (RST-DT) (Carlson *et al.* , 2002) is a corpus of 385 documents in American English. The texts were selected from the Wall Street Journal articles of the Penn Treebank (Marcus *et al.* , 1993). The documents range from 31 to 2124 words. The objectives of the creation of this corpus was to provide annotations grounded within a specific framework, in this case Rhetorical Structure Theory (Mann & Thompson, 1988), and to create a corpus large enough to allow for linguistic analysis, training of statistical models of discourse, and other such computational linguistics applications. In total, 21,789 EDUs are tagged within the

corpus, with an average of 56.59 EDUs per document. EDUs contain on average 8.1 words. Within the RST framework, the discourse structures of this corpus are described as trees describing the following four aspects:

1. The leaves of the tree are the EDUs, which are the minimal units of discourse.
2. The internal nodes of trees are continuous text spans.
3. Each node is either a *nucleus*, containing the more essential unit, or a *satellite*, containing the supporting unit.
4. Each node is characterized by a rhetorical relation.

Other characteristics worthy of mention are that the annotations are designed to be read from left-to-right. This means that EDUs and their relations are to be considered sequentially. Because of this, only consecutive spans of text can be linked together though rhetorical relations.

To better illustrate these annotations, consider the a simple tree taken from the corpus presented in Figure 2.2.

```
( Root (span 1 3)
  ( Nucleus (leaf 1) (rel2par span)
    (text _!Spencer J. Volk, president and chief operating
            officer of this consumer and industrial products
            company, was elected a director._!)
  )
  ( Satellite (span 2 3) (rel2par elaboration-additional)
    ( Nucleus (leaf 2) (rel2par span)
      (text _!Mr. Volk, 55 years old, succeeds Duncan Dwight,_!)
    )
    ( Satellite (leaf 3) (rel2par elaboration-additional-e)
      (text _!who retired in September._!)
    )
  )
)
```

Figure 2.2: Example of the RST Annotation Style

In the example shown in Figure 2.2, the two sentences are split into three distinct leaves containing the smallest EDUs. The first leaf is of type *Nucleus* and is connected to the EDU containing leaves 2 and 3 with a relation *elaboration-additional*. This EDU itself is split into two smaller EDUs, leaf 2, the *Nucleus* and leaf 3, the *Satellite*. The last of these leaves is related to leaf 2 with the relation *elaboration-additional-e*, where the "E" signifies "Embedded"[1].

[1]Note that the relations provided in Figure 2.2 are some of the finer grained relations, as opposed to the 18 meta-relations discussed.

In total, 118 discourse relations are defined, but they are used to form 18 more broadly defined relations. The 18 meta-relations are described in Table 2.2. The meta-relations were based on those provided in (Mann & Thompson, 1987), but have since then evolved into those described in Table 2.2. These modifications lead to the creations of the guidelines described in (Marcu, 1999). The full set

Relation	Description
Attribution	The information provided by an EDU is attributed to a given speaker
Background	An EDU provides background information related to another EDU
Cause	A situation presented in an EDU is caused by a situation presented in another EDU
Comparison	Two situations from two EDUs are compared
Condition	An EDU presents conditions necessary to achieve the situation of the related EDU
Contrast	Two EDUs are compared in a contrasting manner
Elaboration	An EDU is provided to elaborate on the situation described in another EDU
Enablement	An EDU presents a situation enabling the situation of another EDU
Evaluation	An EDU provides an assesment of the situation presented in another EDU
Explanation	An EDU provides explanation on the situation presented in another EDU
Joint	Two or more EDUs are linked in list, such as an enumeration
Manner-means	An EDU provide details on the manner or the means by which the situation of another EDU was achieved
Same-unit	Pseudo-relation used to link spans of text that form an EDU but are not continuous
Summary	An EDU serves as a summary of the situation presented in another EDU
Temporal	EDUs are presented in relation to the order of the occurrences of situations in time
Textual-organization	Pseudo relation used to connect a title or sub-heading to a text
Topic-change	Marks a shift in subject between EDUs. Usually between large spans of text
Topic-comment	An EDU answers or comments on questions presented in another EDU

Table 2.2: List of the 18 Meta-Relations of the RST Framework (Carlson *et al.* , 2002)

of 118 discourse relations and their associated meta-relations are provided in Appendix B.

The annotation of the Rhetorical Structure Theory Discourse Treebank was performed manually by nine different annotators. These annotators were selected on the basis of being professional language analysts who had prior experience in other types of data annotation. Each of these annotators went through a training period prior to their performance of the annotation task. During this training period, the annotators followed three training phases. First, they were introduced to the Rhetorical Structure Theory framework and to the annotation tool provided for the task (Marcu *et al.* , 1999). Second, the annotators were asked to independently tag a short document and subsequently compare and discuss their results. The sessions during which the annotators compared their results served to enhance and improve the annotation guidelines. During these sessions, inter-annotator agreement was tracked regularly. Finally, during a final training phase, the annotators settled on heuristics for handling higher levels of discourse structure.

After the initial training period, the actual annotation task went underway. The first step to building the RST-DT was to segment the texts into EDUs. The guidelines for the annotations used for the creation of the RST-DT asks annotators to treat clauses as the basis for EDUs. This follows the recommendations made in (Mann & Thompson, 1988). In order to allow for a balance between granularity of tagging and consistency of tagging on a large scale, some further instructions were provided to the annotators:

1. Clauses that are subjects, objects or complements of a main verb were not to be treated as EDUs.

2. Relatives clauses, nominal post-modifiers, or clause that break up another EDU were treated as embedded discourse units.

3. A small number of phrasal EDUs were allowed (that is, an EDU composed of a single constituent such as an Noun Phrase or Verb Phrase), so long as the phrase began with a strong discourse marker (e.g. "Because", "In spite of", "As a result of", "According to").

Once the EDUs have been identified, the second step involved linking adjacent spans via rhetorical relations. There are two possible types of such relations: mononuclear and multinuclear. In the case of mononuclear relations, one span is annotated to be the *nucleus* and holds the more important information, while the other is annotated to be the *satellite* and holds the supporting information. In the case of multinuclear relations, two or more spans are noted as *nucleus* and hold information of equal importance in the discourse structure. The annotation guidelines provided to the annotators featured 53 mononuclear relations and 25 multinuclear relations.

The relation annotation process was performed in three phases: First, for about four months, the annotators were tasked to annotate a first set of 100 documents. Once these documents were annotated, the team went through a reassessment phase during which agreement was measured and the annotation guidelines were refined. Finally, the first 100 documents were annotated again with the new improved guidelines, subsequently the rest of the corpus was annotated in the same manner.

It was noted in (Carlson *et al.* , 2002) that the various annotators opted to use a number of tagging strategies during the annotation process. The two most frequently used strategies can be described as follows: The first strategy involves first segmenting the text into EDUs, one unit at a time. The annotators then incrementally built the discourse trees by immediately attaching the current node to the previous node. It was noted that this method required annotators to anticipate the upcoming discourse structure. This need for anticipation grows as the text grows, making this method ill suited for larger texts. It was noted that annotators did in fact prefer the second strategy for such texts. The second strategy involves segmenting the text multiple units at a time. Annotators would then build the discourse sub-trees for each sentence. Once the sentences have been graphed, they are linked together in larger trees, and finally these large chunks are linked together to build the final tree representation of the text. It was noted that this approach allows to see the emerging discourse structure more globally. What can be noted from this is that it seems that approaching discourse structures from the top down is more effective than starting from a specific EDU, especially given larger texts. In the end, an annotated text span for example (7) below can be represented as in Figure 2.3.

(7) [Still, analysts don't expect the buy-back to significantly alter per-share earning in the short term]^{16} [The impact won't be that great,]^{17} [said Graem Lidgerwood of First Boston Corp.]^{18} [This is in part because of the effect]^{19} [of having to average the number of shares outstanding,]^{20} [she said.]^{21} [In addition,]^{22} [Mrs. Lidgerwood said,]^{23} [Norfolk is likely to draw down its cash initially]^{24} [to finance the purchases]^{25} [and thus forfeit some interest income.]^{26}

Using the annotation software provided to the annotators (Marcu, 1999), the previous example is stored in the corpus in a LISP-like format to the short example provided in Figure 2.2.

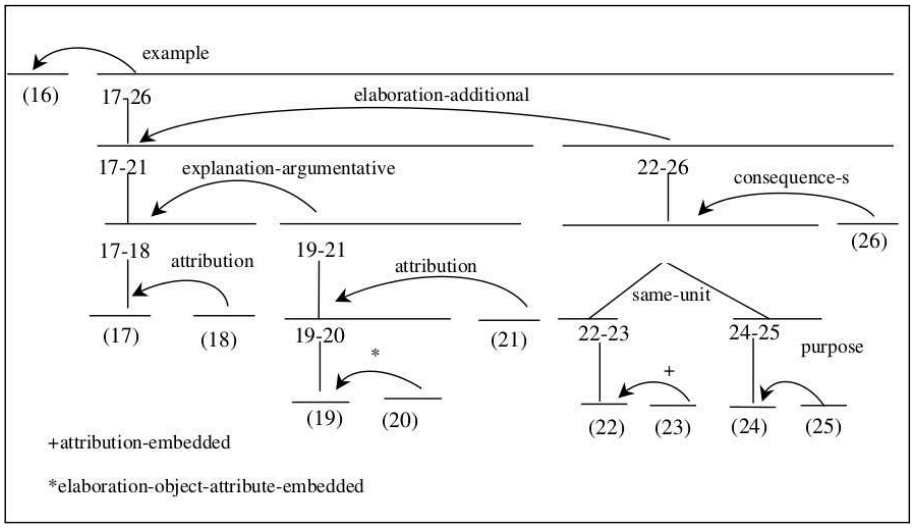

Figure 2.3: Annotated Text Span Example (7) from (Carlson *et al.* , 2001)

Quality assurance was performed on the corpus in two ways: by checking the structural validity of the resulting trees and by tracking inter-annotator consistency. The tree validation process was performed using a discourse parser and a tree traversal program in order to verify that the entire text was part of the tree, that it contained a single root node, that nuclearity was assigned properly and that relations attached to the proper types of nodes. Inter-annotator consistency was checked using Kappa statistics (Carletta, 1996) over hierarchical structures. Using the method described in (Marcu, 1999), an agreement measure larger than 0.8 is considered to represent very high agreement, while a measure between 0.6 and 0.8 is considered to represent good agreement. Table 2.3 details the inter-annotator agreement from the various annotation tasks performed by six of the nine annotators.

Annotators	Units	Spans	Nuclearity	Relations	Fewer-Relations	No. of Docs	Avg. No. EDUs
B,E	0.96	0.89	0.85	0.78	0.81	7	88.29
A,E	0.98	0.90	0.84	0.76	0.78	6	57.67
A,B	1.00	0.93	0.88	0.79	0.82	5	58.20
A,C	0.95	0.84	0.78	0.68	0.71	4	116.50
A,F	0.95	0.78	0.69	0.60	0.62	4	26.50
A,D	1.00	0.87	0.80	0.72	0.77	4	23.25

Table 2.3: Inter-Annotator Agreement – Final Result for Six Taggers From (Carlson *et al.* , 2002)

Table 2.3 reports the final inter-annotator agreement for all pairs of annotators who double-annotated four or more documents. Out of the 385 documents, 53 of these were double-annotated. As can been seen in these results, inter-annotator agreement is generally quite high. Identifying

EDUs and the spans of text of the leave nodes appears to yield very high agreement overall. The exception here is found between annotators A and F, with an average Kappa measure slightly under the 0.8 cutoff. Relations extraction appears to be the more difficult task when it comes to consistency. For this reason, the "Fewer-Relations" measure shows agreement between annotators using the 18 broad classes of relations of the RST framework. In the end, it is noted that the greatest sources of inconsistency came from the fact that the RST relation set is quite large with 118 distinct relations, and the concept of nuclearity is somewhat interpretive. Because of these reasons, the annotators received more leeway in their interpretation of discourse structures. This is noted to be the case especially with larger text spans where building the tree structures were noted to be a cognitive challenge for those involved.

In the end, the RST-DT was developed by over a dozen people working on full-time or part-time basis for a period of a year.

2.3.1.2 Rhetorical Structure Theory Review Corpus

The Rhetorical Structure Theory (RST) Review Corpus was created by and described in (Taboada *et al.* , 2006; Taboada & Grieve, 2004) for the purpose of using discourse relation identification for the task of sentiment analysis. It comprises 400 documents obtained from *epinions.com*, an online resource of user generated reviews on various products. Eight categories of products were selected and 50 reviews, with 25 negative and 25 positive reviews, on each subject were chosen. These categories are: books, cars, computers, cookware, hotels, movies, music, and phones. On average, these reviews have a length of 1615 tokens. Discourse relations were annotated by Maite Taboada and Montana Hay using the **RST Tool** created by Michael O'Donnel (O'Donnell, 1997). The annotation were made using the set of discourse relations from the RST framework. Only relations found within sentences were noted, those holding between sentences or through larger portions of the text were ignored.

2.3.1.3 Penn Discourse Tree Bank

The Penn Discourse Tree Bank (PDTB) (Prasad *et al.* , 2007) is a corpus of 2159 documents annotated for discourse relations which was released in February 2008. It covers the entire span of the *Wall Street Journal* articles found in the Penn Treebank (Marcus *et al.* , 1993). These documents, once again, are written in American English and range from 31 to a little over 2000 words in length. The authors of the PDTB created a framework which they claim to be theory neutral when it comes to the task of discourse annotations. Within this framework, relations are meant to be held between two and only two arguments (or what we have thus far referred to as EDUs).

One major difference between the RST framework and the PDTB framework is the emphasis made on *explicit* and *implicit* relations. Within the PDTB corpus, relations are noted to be either *explicit* if they are made through the use of a cue phrase, and *implicit* in the absence of such a cue phrase. For example, a *cause* relation could be made explicitly as:

(8) I like this shirt, **because** it is blue.

The appearance of the cue phrase "because" makes this relation *explicit*, and therefore much easier to identify. On the other hand, simply removing this cue phrase from our sentence to create the new sentence:

(9) I like this shirt, it is blue.

still denotes the same relation of *cause* between the two EDUs of the sentence. This relation, however, is considered to be *implicit*, as no cue phrase appears to identify the relation itself. Instead, the reader is expected to understand the causality implied by the juxtaposition of both clauses found in the sentence. It should come as no surprise that *implicit* relations are generally much harder to identify. Another important note to consider is that cue phrases are not limited to single word expressions like "because", although many are, but could also be composed of expressions such as: *only because*, *if and when*, *either [...] or*, etc. In total, the PDTB corpus denotes 100 different cue phrases, some of which have modified forms. The annotations made on the Penn Treebank corpus generated 18,459 instances of *explicit* discourse relations and 16,224 instances of *implicit* discourse relations. This goes to show that the distribution of these relations on the basis of this distinction splits discourse relations fairly evenly. This suggests that identifying cue phrases alone will not be sufficient for properly identifying discourse relations in an adequate fashion.

The PDTB framework defines four broad classes of discourse relations, which themselves are split into two to six more fine grained relation types, some of which are further refined into even more specific relation subtypes. These classes and discourse relations types are shown in Figure 2.4.

A particular relation that can also be found is the *attribution* relation, not mentioned in Figure 2.4. Unlike with the RST framework, the PDTB framework gives a special importance to this particular discourse relation. The authors argue that the nature of the discourse relation holding between two arguments is, in part at least, determined by the attribution of these arguments. That is, having an argument attributed to a source other than the author of a document means that the discourse of this external source should be considered separately during the identification of discourse relations (Prasad *et al.* , 2008).

The annotation produced for the PDTB corpus provides information on *explicit* and *implicit* relations, as well as what is referred to as *AltRel*, *EntRel* and *NoRel* relations. In the case of *explicit* relations, both the arguments (or EDUs), the sense of the relation and the connective (or cue phrase) are noted by the annotators. In the case of *implicit* relations, both the arguments (or EDUs), the sense of the relation and an **implicit** connective are noted by the annotators. The **implicit** connective is a cue phrase that can be inserted in the original text to turn the *implicit* relation into an *explicit* one without changing its sense. In the cases where such a connective cannot be added by the annotators, *AltRel*, *EntRel* and *NoRel* relations where identified. In the case of *AltRel*, the insertion of such a cue phrase would have been redundant as an non-connective alternate expression was used for this purpose. Consider the following example from (Prasad *et al.* , 2008):

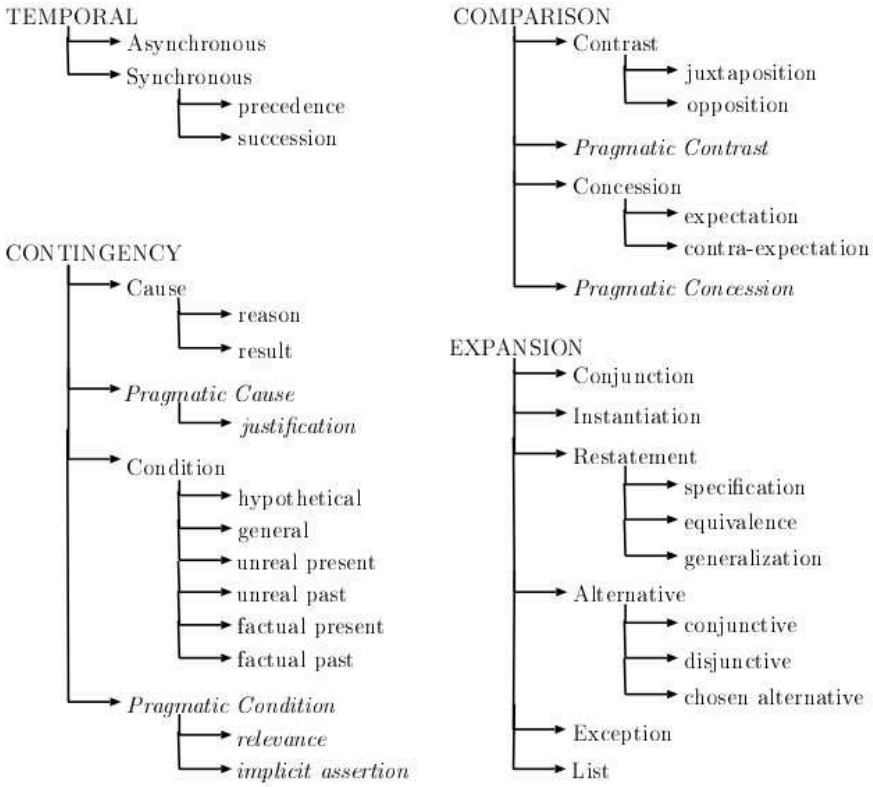

Figure 2.4: List of Relations in the PDTB Framework from (Prasad *et al.* , 2007)

(10) Ms. Bartlett's previous work, which earned her an international reputation in the non-horticultural art world, often took gardens as its nominal subject. **Mayhap this methaphorical connection made** the BPC Fine Arts Committee think she had a literal green thumb.

In this last example, the bolded portion of the text is perceived as the non-connective alternate expression (it restates what is described in the portion of the document that preceeds it). In the case of *EntRel*, only entity based coherence can be perceived between the two arguments (or EDUs). That is, a relation between the two arguments seems to exist, but no relation (as defined within the framework) add to the semantics of this relationship. In the case of *NoRel*, neither a discourse relation or an entity based relations could be perceived between the two arguments (or EDUs).

The PDTB corpus was annotated by two annotators and inter-annotator agreement was computed for all three levels of discourse relations: classes, types, and subtypes (see figure 2.4). Disagreement was calculated at the class level when the two annotators picked a subtype, type or class of different classes, at the type level when they picked different types within the same class, and at subtype level when both annotators picked different subtypes. The accuracy for each of these levels of agreement is shown in Table 2.4.

As mentioned earlier, the *attribution* relation is given a special consideration whithin the PDTB

Level	% agreement
Class	94%
Type	84%
Subtype	80%

Table 2.4: PDTB Corpus Inter-Annotator Agreement

framework. Unlike the RST framework which features such a discourse relation no different than the rest, the authors of the PDTB framework decided to treat the *attribution* relation as a special case, outside of the discourse annotation itself. Given the special considerations made to *attributions*, these were treated separately by a single expert.

2.3.1.4 Biomedical Discourse Relation Bank

The Biomedical Discourse Relation Bank (BioDRB) was created by and described in (Prasad *et al.* , 2011). It consists of 24 open-access full-text articles in the field of biomedicine obtained from the GENIA corpus (Kim *et al.* , 2003). Each article is formatted in the way research papers are typically expected to be structured. They can be separated into sections such as: abstract, methodologies, discussions, results, etc. The guidelines used for the annotation task of these articles is based on the ones created for the creation of the PDTB corpus. Some changes were made to the definitions of a few relations while others were rearranged within the hierarchy of senses defined by the original PDTB framework. First, the authors of the BioDRB corpus decided to completely eliminate the **class** level, originally defined as the four classes: *contingency*, *temporal*, *comparison*, and *expansion*. Only the *temporal* class was kept, but now as a **type**. With this change, the hierarchy of the PDTB framework used for the annotation of the BioDRB corpus is on two levels, rather than three. A second change noted by the authors is the collapsing of certain subtypes. For example, the original PDTB framework defined a *factual-present* and *factual-past* relation, the BioDRB version of the framework label both these relations as *factual*. A third change is the addition of a few new discourse relation senses: *purpose*, *similarity*, *continuation*, *background*, and *reinforcement*. In the case of *background* and *continuation*, these are noted by the authors as being reformulations of the *EntRel* type of relations. *AltLex* and *NoRel* still exist within this new version of the PDTB framework. The other three new senses were created to give better precision to certain discourse relations that were believed to be compounded with these new ones. For example, relations noted to be *purpose* would have been identified as *result* in the original PDTB framework. Finally, *pragmatic cause* became a subtype of *cause*, while *pragmatic condition* became a subtype of *condition*.

The BioDRB corpus contains the annotations for 5859 discourse relations, including all relation types: *explicit*, *implicit*, *AltLex*, and *NoRel*. The distribution of these relations is shown in Table 2.5.

For the purpose of annotating the 24 articles with discourse relations, annotators were first given the original list of discourse connectives (or cue pharses) from the original PDTB corpus, but were also encouraged to identify new instances of such connectives. The task of annotating the discourse

Relation Type	No. of Tokens	Distribution
Explicit	2636	45.0%
Implicit	3001	51.2%
AltLex	193	3.3%
NoRel	29	0.5%
Total	**5859**	**100%**

Table 2.5: Distribution of Relation Types in the BioDRB Corpus

relations was then performed in five steps, for every sentence:

1. Identify, if it exists, discourse connectives relating this sentence to the context via a discourse relation and mark it as *explicit*.
2. If no *explicit* relation can be identified, attempt to insert a connective in order to infer the *implicit* relation and mark it as such.
3. If the insertion of an *implicit* connective leads to redundancy, note the relation as *AltLex*.
4. If the sentence does not seem to relate in a coherent manner to the context, mark the relation as *NoRel*.
5. After relating the sentence to the previous context, identify and annotate any discourse relations found within the sentence itself.

The annotation task was performed by two pre-medical students at the University of Pennsylvania. Given the highly specialised nature of the documents annotated, the expertise provided by these annotators was a crucial point in their selection. The authors of (Prasad *et al.* , 2011) then proceeded to train the two annotators on linguistic theories of syntax, semantics, and discourse. Following this initial training, they were also given a training on the guidelines for this specific annotation task. The annotation task took over three years of work. Agreement between the two annotators was calculated on the basis of connective (or cue phrase) identification, argument (or EDU) identification, and sense labelling. The authors note an agreement of 82% on the identification of discourse connectives. For the identification of arguments (or EDUs), the authors calculated the accuracy of the exact match between the two annotators for relations that are *explicit* or *AltRel* and for relations that are *implicit*. The accuracies were then calculated for both arguments of each annotated discourse relations. Table 2.6 gives some details on the accuracy of these agreements.

Finally, agreement of sense labelling is given through Kappa scores. As noted in (Prasad *et al.* , 2011), the use of Kappa scores is used since the categorization task is multinominal, with several sense labels to chose from when annotating discourse structures. The results of these calculations is provided in Table 2.7.

After these agreement measures were calculated, the disagreements found were re-evaluated and fixed by an expert. Further reviews were also performed by the authors to fix remaining guideline related errors.

Relation Type	Argument	Agreement
Explicit, AltLex	First Argument	81%
	Second Argument	85%
Implicit	First Argument	75%
	Second Argument	88%

Table 2.6: Agreement on the Identification of EDUs in the BioDRB Corpus

Relation Type	Kappa
Explicit, AltLex	0.71
Implicit	0.63

Table 2.7: Sense Labelling Agreement in the BioDRB Corpus

2.3.2 Discourse Relation Parsers

Over the past decade, several discourse relation parsers have been developped. Some of the earlier ones were trained using the RST framework and associated corpora, while others appearing later made use of the PDTB framework and associated corpora. Both frameworks are still actively used in the creation and improvement of automatic discourse parsers.

2.3.2.1 SPADE

Radu Soricut and Daniel Marcu were the first to present two probabilistic models for the purpose of identifying EDUs and building sentence-level discourse parse trees (Soricut & Marcu, 2003). These models were subsequently used to build the SPADE (Sentence-level PArser for DiscoursE) parser. The first of these models is used to split a text into leaf nodes representing the smallest EDUs. The second of these models attempts to build sentence-level discourse trees using these EDUs. In the end, the authors report what they describe as "near-human levels of performance" in extracting sentence-level parse trees from unannotated text. We now describe the two models proposed by the authors.

The first model, the discourse segmentation model, takes a raw text and splits it into spans that will become the leaf nodes of the parse trees, the EDUs. It does so by identifying discourse boundaries. The first step of the process of identifying EDUs is to split the text at the sentence level. In order to select the appropriate location to insert these discourse boundaries, the authors find that this is best achieved by considering lexicalized syntactic structures. The approach first takes into account words and determines if it is a *boundary* word or a *non-boundary* word. In order to reach such a goal, the approach works in two steps: sentence segmentation and sentence-level discourse segmentation. The first step relies on already established approaches described in (Palmer & Hearst, 1997) and (Ratnaparkhi, 1998). Using the methods described by these authors, texts can effectively be split into sentences. The sentence boundaries are always considered to be EDU boundaries. The later step of splitting these sentences into the actual EDUs which will become the

leaf nodes is performed by two components. The first of these components is a statistical model which calculates the probability of a boundary to occur after every word of the sentence. The second component selects the most likely location determined by the first model and inserts the boundary separating the text into EDUs. To do so, the method relies on creating a syntactic parse tree for the sentence. This is achieved using Charniak's syntactic parser (Charniak, 2000). Using the syntactic trees obtained as the output of the Charniak parser, canonical lexical head projection (Magerman, 1995) is applied in order to retrieve the syntactic trees in a lexicalized form. Every word that is the lexical head of a constituent and that has a right sibling constituent is a possible candidate for a EDU boundary. In order to better illustrate this, consider the parse tree presented in Figure 2.5, reproduced from (Soricut & Marcu, 2003).

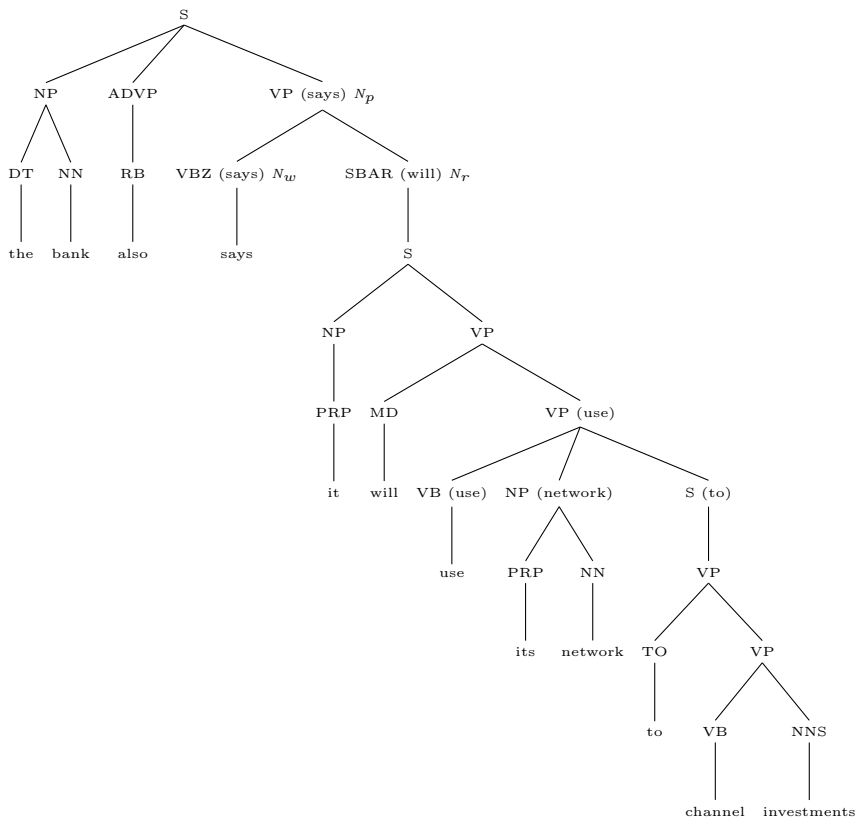

Figure 2.5: Example Syntactic Tree from (Soricut & Marcu, 2003)

The approach determines that the text span "The bank also says" constitutes an EDU while "it will use its network to channel investments" constitutes another. It should also be noted that this second EDU can be further split into two smaller EDUs: "it will use its network" and "to channell investments". The beginning of the first EDU is determined by the sentence boundary. "The" is therefore classified as a *boundary* word. In order to determine that "says" is the last word of this EDU and should therefore be classified as a *boundary* word, the system looks at the following features: the node $\mathtt{N}_w$, its parent node $\mathtt{N}_p$ and its sibling node $\mathtt{N}_r$. Given that the nodes presented here are $\mathtt{N}_w$ = VBZ(says), $\mathtt{N}_p$ = VP(says) and $\mathtt{N}_r$ = SBAR(will), the system calculates the likelihood

of a boundary to occur based on the lexical and syntactic information provided. In other words, we are interested in the probability of a syntactic structure VBZ with a lexical head "says" being a boundary word considering that it is part of a larger syntactic structure `VP` with lexical head "says" and has a right sibling with a syntactic structure `SBAR` with a lexical head "will". The probability is computed by these authors by training their system on the RST-DT corpus (Carlson *et al.* , 2002).

Table 2.8 shows the results obtained by the SPADE algorithm in terms of precision, recall and F-score when it comes to identifying EDU boundaries. Note that in order to asses the impact of

Syntactic Parse	**Recall**	**Precision**	**F-score**
Using Charniak	82.7	83.5	83.1
Using Penn Treebank	85.4	84.1	84.7
Human Performance	98.2	98.5	98.3

Table 2.8: Evaluation of the SPADE Discourse Segmenter

errors produced by the Charniak parser, the authors also carry an evaluation based on the gold standard syntactic trees provided from the Penn Treebank. The results show a slight improvement. Finally, the human performance is evaluated using the doubly annotated texts from the RST-DT (see Section 2.3.1.1), that is, texts that were annotated by two annotators and then compared.

Once discourse boundaries have been successfully identified, that is agreement between annotator was deemed satisfactory, both these and the lexicalized syntactic trees (trees in which constituents are labelled with the lexical items found at their heads) are used as input to the discourse parser. This is done in two steps: the *parsing model* assigns the probability of each possible parse tree given the specified input, while the *discourse parser* is an algorithm used to determine which of these possible parse trees is the better choice.

The *parsing model* extracts a set of tuples from the input. These tuples can be described as $R[i, m, j]$ where:

- R is the relation holding between two spans of text,
- i is the first EDU found in the text span considered,
- m is the first EDU found in the second of our text spans, and
- j is the last EDU of the second of our text spans.

In other words, the tuple $R[i, m, j]$ denotes the relation R between the text span from EDUs i to m and the text span from EDUs m to j. For example, take sentence 11 showing a text span made of three EDUs.

(11) [The bank also says]$_1$ [it will use its network]$_2$ [to channel investments]$_3$.

In this example, an *attribution* relation was annotated to hold between EDU_1 and the span from EDU_2 to EDU_3. The tuple would then be: $attribution[1, 2, 3]$. Given the input, every possible pair of text spans is to be considered for the basis of one of these tuples. For example, consider a text separated into four EDUs: $(1, 2, 3, 4)$. The following spans should be considered as candidates for the

basis of a relation: $R[1,2,2], R[1,2,3], R[2,3,3], R[2,3,4], R[3,4,4]$ This method allows to identify rhetorical relations between leaf nodes as well as between larger spans of text. On the other hand, it assumes that all relations are binary relations. This assumption is noted by the authors as being justified by the fact that 99% of the nodes in the RST-DT are binary nodes. Using two methods $rel()$ and $ds()$ which can be applied to tuples $R[i,m,j]$ and yielding the relation R and the discourse structure $[i,m,j]$ respectively, the authors suggest the formula of Equation 2.5.

$$P(DT|\Theta) = \prod_{c \in DT} P_s(ds(c)|\Theta) \times P_r(rel(c)|\Theta) \tag{2.5}$$

This formula should be understood as computing the probability of a discourse structure DT given a set of parameters Θ. This is determined by calculating the product of the probability of every tuple c in DT in terms of discourse structure and relation given the parameters Θ. The first of these probability measures (P_s) aims at finding the most likely hierarchical structure for the discourse tree. For example, given a text containing three leaves, it determines which of the following two structure is more likely: $(1,(2,3))$ or $((1,2),3)$. The second of these probability measures (P_r) identifies the type of relation that is most likely to appear in the leaves.

The features used to determine the probabilities P_s and P_r are based around what is described as the "attachment points" of EDUs. First, for each EDU, a head word is selected. Given the lexicalized parse tree used as input, the head word H is defined as the word with the highest occurrence as a lexical head within the lexicalized syntactic structure of the EDU. The node where the head word appears highest in the parse tree of the given EDU is called the *head node* of the EDU and is denoted N_H. The EDU in which the root node is present is called the *exception EDU*. For every N_H that is not part of the *exception EDU*, there exists a node that is the parent of N_H. This parent node is called the *attachment node* and is denoted N_A. For every EDU to be considered, the feature set $\{H, N_H, N_A\}$ is used to build the D set, called the *Dominance Set*. An example of such a set is shown in Equation 2.6, which shows the *Dominance Set* of the sentence of Figure 2.5.

$$D = \{(2, SBAR(will)) < (1, VP(says)), (3, S(to)) < (2, VP(use))\} \tag{2.6}$$

The set D should be understood as: there is a head node of type $SBAR$ with lexical head "will" in EDU 2 which is dominated by a head node of type VP with a lexical head "says" in EDU 1 and a head node of type S with a lexical head "to" in EDU 3 which is dominated by a head node of type VP with a lexical head "use" in EDU 2. The discourse parse tree (DT) is computed based on the dominance set D with Equation 2.7.

$$P(DT|D) = \prod_{c \in DT} P_s(ds(c)|filter_s(c,D)) \times P_r(rel(cs)|filter_r(c,D)) \tag{2.7}$$

Equation 2.6 should be understood as the probability of a discourse parse tree DT given the dominance set D as the product of the probability of every tuple c in DT in terms of discourse structure and relation given the dominance set D. In both of these two probabilities, a filter is applied in order to ensure that only the information in D related to the EDUs in the current c are considered. With $filter_s$, only the syntactic labels and an identifier of the EDU in which they are found are considered, and the lexical heads are discarded. With $filter_r$, both the lexical heads and syntactic

labels are considered, but only for the nodes which connect the two EDUs. To better illustrate this, consider the example provided in Figure 2.6.

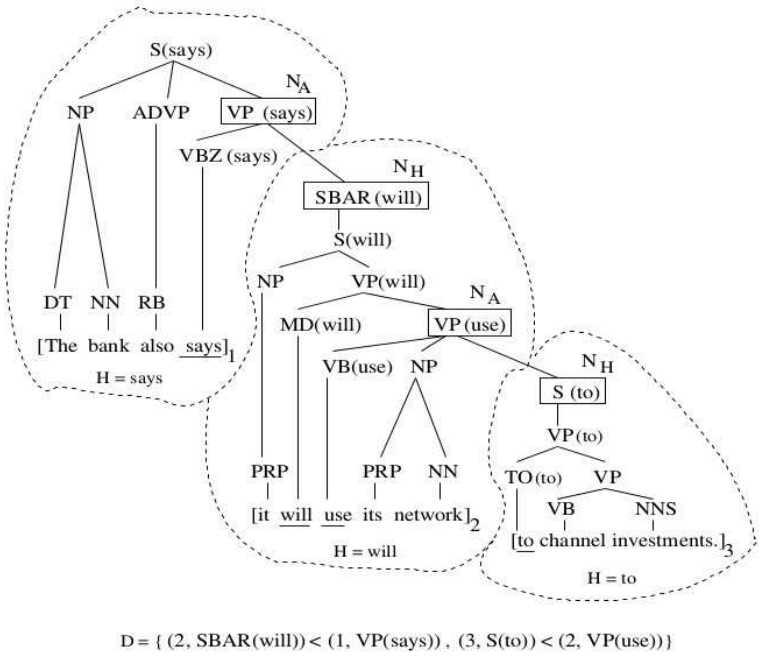

Figure 2.6: EDU Segmentation at the Syntactic Level

c =*enablement-ns*$[2, 2, 3]$ is a potential discourse parse for the example sentence of Figure 2.6. Recall that this structure means that the text "it will use its networks to channel investments" can be parsed as $R[i, m, j]$ =enablement-ns$[2, 2, 3]$ reflecting the relation of *enablement-ns* between the second and third EDUs of the full sentence. For $filter_s(c, D)$ we obtain the set $\{(2, SBAR) < (1, VP), (3, S) < (2, VP)\}$. In simple terms, the $SBAR$ node at the head of EDU_2 is dominated by the VP node of EDU_1 and the S node at the head of EDU_3 is dominated by the VP node of EDU_2. For $filter_r(c, D)$ we obtain the set $\{S(to) < VP(use)\}$ which indicates the node S lexicalized by "to" is dominated by the node VP lexicalized by "use".

In the case of relation extraction, SPADE's discourse parser obtains the results shown in Table 2.9 in terms of F-score based on a standard Parseval metric (Abney *et al.* , 1991). The scores given in Table 2.9 show the F-score when leaving the relations unlabelled, by using the set of 18 higher level relation labels described in Section 2.3.1.1 and finally by using the 118 relation labels found in the training corpus. As a basis of comparison, the human performance measures are based on the doubly-annotated texts of the RST-DT.

Although human performance is clearly better than what can be achieved with SPADE, at the time it was first published, SPADE was the most complete and best performing system. Since then, a few others have been developed using richer sets of features that allow for better performance.

	SPADE	Human Performance
Unlabelled	70.5%	92.8%
18 Labels	49.0%	77.0%
118 Labels	45.6%	71.9%

Table 2.9: F-scores for SPADE Discourse Parser Evaluation Using the Parseval Metric, from (Soricut & Marcu, 2003)

2.3.2.2 HILDA

In 2010, (Hernault *et al.* , 2010b,a, 2011) presented another discourse parser, HILDA (HIgh-Level Discourse Analyser) which reuses some of the methods described by (Soricut & Marcu, 2003) and used in the SPADE parser. One distinct improvement of HILDA over SPADE is that it performs text-level discourse extractions as well, as opposed to sentence-level extraction alone.

A few working assumptions are considered by the authors. First, they acknowledge the specific characteristic of RST as being presented in a left-to-right linear order. Second, they chose to use the simpler set of 18 meta-relations (see Section 2.3.1.1) as opposed to the 118 relations available in the RST-DT. Finally, although the RST allows for multi-nuclear relations, the authors chose to treat those as nested binary trees. For example, the *list* relation which would normally be represented as multi-nucleic relation becomes several nested trees. Figure 2.7 illustrates how such a change is produced.

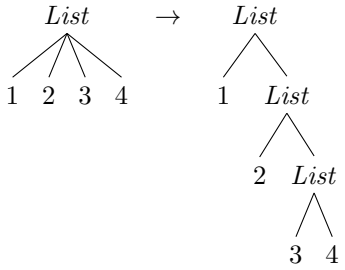

Figure 2.7: Binarization of Multi-Nuclear Relations

The reason why this conversion became necessary is due to the fact that the HILDA system uses a Support Vector Machine (SVM) classification approach (Vapnik, 1999). Using such a classifier, HILDA proceeds in four steps:

1. Segment the text into EDUs.
2. Find the most likely relation between each pair of consecutive EDUs, select the one with the highest probability and insert the new span with its relation to replace the two original EDUs.
3. Relabel all consecutive EDUs and repeat the previous step.
4. Continue until all text spans have been merged into one.

Much like SPADE, HILDA separates the tasks of discourse structure extraction into two broad steps: discourse segmentation and relation labelling. Both steps make use of an SVM classifier with different sets of features extracted from the input. For the purpose of detecting boundaries between EDUs, the authors opt to use a method similar to SPADE and use the same feature set: lexical items, part-of-speech tags and lexicalized heads.

The second task of HILDA is to perform relation labelling. Again, a number of features are extracted or derived from the input. (Hernault *et al.* , 2010a) use the following types of features in their labelling process:

1. textual organization features,
2. lexical features,
3. dominance sets as described by (Soricut & Marcu, 2003), and
4. structural features.

Textual organization features are summarized in Table 2.10, where they are classified as either (S)pan features which are extracted from both right and left spans of text separately and (F)ull which are extracted from the entire span of both EDUs as a single text span.

Feature Name	**Scope**
Belong to same sentence	F
Belong to same paragraph	F
Number of paragraph boundaries	S
Number of sentence boundaries	S
Length in tokens	S
Length in EDUs	S
Distance to beginning of sentence in tokens	S
Size of span over sentence in EDUs	S
Size of span over sentence in tokens	S
Size of both spans over sentence in tokens	F
Distance to beginning of sentence in EDUs	S
Distance to beginning of text in tokens	S
Distance to end of text in tokens	S

Table 2.10: Features Encoding Textual Organization Used in HILDA

Lexical features are noted as being good indicators of discourse relations (Soricut & Marcu, 2003; Hernault *et al.* , 2010a, 2011). The appearance of specific cue phrases provide powerful clues pertaining to the underlying discourse relations used. For example, the use of the term "because" is a good indicator of relation of causality between two clauses. The presence of specific cue words makes the use of pre-defined dictionaries of cue words a possible approach. With HILDA, the authors instead opt to use 3-grams built from the tokens (found in the RST-DT corpus) at the beginning

and end of every EDU. (Hernault *et al.* , 2010a) note that their approach performs less well than the dictionary approach, but has the advantage of not being reliant on a cue-phrase dictionary, making it more flexible (that is, new cue phrases that were not found in the training data do not cause an issue but remaining unidentified). In an effort to improve the performance of this approach, HILDA uses part-of-speech tags on the tokens of these 3-grams, allowing a slight increase in performance.

The third set of features considered by the HILDA system during relation extraction is dominance sets. Again, these are built from the lexicalized heads of the syntactic tree of EDUs and concentrate on the point where EDUs are linked together syntactically. A detailed list of these features is presented in Table 2.11 where, again, the scope is noted to hold across both spans separately or across the full text span of both EDUs.

Feature Name	**Scope**
Distance to root of the syntax tree	S
Distance to common ancestor in the syntax tree	S
Delta of distance to common ancestor	F
Dominating node's lexical head in span	S
Common ancestor's POS tag	F
Common ancestor's lexical head	F
Dominating node's POS tag	F
Dominating node's lexical head	F
Dominated node's POS tag	F
Dominated node's lexical head	F
Dominated node's sibling's POS tag	F
Dominated node's sibling's lexical head	F
Relative position of lexical head in sentence	S

Table 2.11: Features Encoding Dominance Sets Used in HILDA

Finally, structural features are considered by HILDA during the labelling process. (Hernault *et al.* , 2010a) notice that there seems to exist a correlation between relations at different levels of the tree. A trivial example would be the appearance of a *list* relation which, now that these have been converted to binary relations, is more likely to be followed by the same type of relation. The structures of the discourse trees from the RST-DT training set are encoded as features in a breath-first, flat list representation of the binary tree. This can then be used to better identify a series of relations. Although the features used by SPADE and HILDA are the same, it appears that HILDA outperforms SPADE due to its use of SVM when it comes to identifying EDU boundaries. Table 2.12 shows the F-score recorded by (Hernault *et al.* , 2010a). Similarly to the results shown in Table 2.8 (see Section 2.3.2.1), note that the authors record these measures by using both the syntactic trees obtained from the Penn Treebank and by deriving those tress using the Charniak parser. Again, the human agreement measures are provided as a basis of comparison.

Finally, (Hernault *et al.* , 2010a) note the performance measures using the set of 18 discourse

System	Trees	Precision	Recall	F-score
SPADE-Seg	Penn	84.1	85.4	84.7
HILDA-Seg	Penn	95.5	94.5	95.0
SPADE-Seg	Charniak	83.5	82.7	83.1
HILDA-Seg	Charniak	94.7	93.4	94.0
Human Agreement		98.5	98.2	98.3

Table 2.12: Performance of Discourse Segmenters from (Hernault *et al.* , 2010a)

relations, evaluated on 21 documents from the RST-DT. This is shown in Table 2.13.

	Structure	Nuclearity	Relations
Precision	76.0	61.4	51.2
Recall	75.6	61.2	50.6
F-score	75.8	61.3	50.9

Table 2.13: Performance of HILDA from (Hernault *et al.* , 2010a)

2.3.2.3 Feng & Hirst Parser

More recently, the HILDA parser was improved by Vanessa Feng and Graham Hirst of the University of Toronto (Feng & Hirst, 2012). In order to achieve this, rich linguistic features were used to improve the tree building step of the discourse relation extraction process. Some of the new features incorporated are based on the work described in (Lin *et al.* , 2009) while others are novel. Four new approaches were investigated and used to improve the HILDA parser. Contextual features takes into account that a coherent text typically uses particular sequences of discourse relations. Given this observation, the new parser attempts to better identify discourse relations based on which relations precede and succeed the one currently observed. This is noted to be somewhat tricky as the RST framework does not produce linear series of features, but rather tree structures where different discourse relations are embedded inside one another. An idealized solution is described in (Feng & Hirst, 2012) to obtain the full context of any given discourse relation, but given the bottom-up approach implemented by HILDA, such an ideal situation is not always feasible. A second new feature investigated by (Feng & Hirst, 2012) is the use of discourse production rules. Much like syntactic production rules (e.g. $S \rightarrow NP + VP$), such rules take into account the tree like structures of discourse relation schemas within the RST framework. These production rules were derived from the RST-DT corpus which annotated multilevel discourse structures. A third new feature compares semantic similarities of verbs and nouns across the EDUs linked by a discourse relation. The similarity between these various tokens is computed using VerbNet and WordNet (Fellbaum, 1999; Schuler, 2005). Finally, cue phrases are used to better identify discourse relations by determining if the cue phrases appear in the begining, middle or end of a span of text. Another important strategy adapted from the work of (Lin *et al.* , 2009) is to perform feature

selection prior to classification in an effort to reduce the total number of feature dimensions. This was done by making an analysis of the most informative features through experimental work. By computing the information gain provided by each of the features used by the parser, whether they are part of the original features set of HILDA, or some of the new ones introduced by Feng & Hirst, the authors reduced the total number of selected features to 21,410 (the total number of features before applying feature selection is not mentioned in (Feng & Hirst, 2012), but those features did include all possible word pairs from all text spans and is therefore expected to be quite high). Table 2.14 shows a comparison of the performance of the Feng & Hirst parser against the HILDA results on the identification of discourse structures, assuming the ideal situation allowing for the contextual feature, in terms of accuracy, precision, recall, and F-score score. In all cases, the F-score recorded by (Feng & Hirst, 2012) shows an improvement over HILDA. In fact, the Feng & Hirst parser improves discourse relation extractions within sentences for all measures observed: accuracy, precision, recall and F-score. A drop in precision is noticeable when it comes to identifying discourse relations across sentences, but the recall rate for these same discourse relations is noted to be considerably higher. Table 2.15 shows a similar comparison on the performance of discourse relation classification in terms of *Macro-Average F-scores* (MAFS)[2], and *Weighted-Average F-scores* (WAFS)[3], as well as the accuracy. (Feng & Hirst, 2012) notes that MAFS is not influenced by the number of instances that exist in each relation class by equalling weighting those classes. That way, this measure is not biased by the much larger appearance of certain discourse relations in comparison to others. The WAFS measure weights the performance of each class by the count of instances. Using these metrics, the Feng & Hirst parser is shown in Table 2.15 to improve slightly on HILDA in all observed cases.

	Within sentences		**Across sentences**		**All structures**	
	Feng	HILDA	Feng	HILDA	Feng	HILDA
Accuracy	91.04	83.74	87.49	89.13	95.64	87.04
Precision	92.71	84.81	49.60	61.90	94.77	79.41
Recall	90.22	84.55	63.95	28.00	85.92	58.15
F_1	91.45	84.68	55.87	38.56	89.51	67.13

Table 2.14: Feng & Hirst vs HILDA Parser Performance on Structure Classification

[2]The Macro average F-score (MAFS) is the harmonic mean of the average of the recorded precision and recall measures. Given two precision values (P_1 and P_2), and two recall values (R_1 and R_2), the MAFS would be calculated as the standard F-measure (see Section 2.1) but using the average precision $P = (P_1 + P_2)/2$ and recall $R = (R_1 + R_2)/2$.

[3]The Weighted-Average F-score (WAFS) takes into consideration that not all classes might be equally represented. If, for example, class C_1 has 100 instances with a precision score of 30%, while class C_2 has 2 instances with a precision score of 100%, the average precision would become 65%. This is problematic as the score is inflated by class C_2. In order to avoid this misleading situation, the number of instances for each class determines how much each class weights on the overall score. To obtain the weighted precision used to calculate the WAFS, we would use $P = (100 \times 65) + (2 \times 100)/102 = 31.4\%$.

	Within sentences		Across sentences		All relations	
	Feng	HILDA	Feng	HILDA	Feng	HILDA
MAFS	0.490	0.446	0.194	0.127	0.440	0.379
WAFS	0.763	0.740	0.607	0.588	0.607	0.588
Accuracy	78.06	76.42	65.30	64.18	65.30	64.18

Table 2.15: Feng & Hirst vs HILDA Parser Performance on Relation Classification

Input: a text T
Output: a discourse structure of T

```
// Step 1: label Explicit relations
Identify all connective occurrences in T
for each connective occurrence C do
    Label C as disc-conn or non-disc-conn
    if C is disc-conn then
        Label Arg1 span and Arg2 span of C
        Label (C, Arg1, Arg2) as one of the Explicit relations

// Step 2: label Implicit, AltLex, EntRel, and NoRel relations
for each paragraph P in T do
    for each adjacent sentence pair (S_i, S_j) in P do
        if (S_i, S_j) is not labeled as an Explicit relation in Step 1 then
            Label (S_i, S_j) as EntRel, NoRel, or one of the Implicit/AltLex relations

// Step 3: label attribution spans
Split T into clauses
for each clause U do
    if U is in some Explicit/Implicit/AltLex relation from Step 1 or 2 then
        Label U as attr-span or non-attr-span
```

Figure 2.8: Pseudo Code for the Discourse Parsing Algorithm of the End-to-end PDTB Parser (Taken from (Lin *et al.* , 2012))

2.3.2.4 End-to-End PDTB Parser

The End-to-End parser created by and described in (Lin *et al.* , 2012) differs from those described above in the fact that it makes use of the PDTB framework (see Section 2.3.1.3), as opposed to the RST framework (see Section 2.3.1.1). The authors describe the parser as an attempt to mimic the procedure used by human annotators in the creation of the PDTB corpus. Figure 2.8 provides a simple pseudo-code explanation of the algorithm used by the parser described in (Lin *et al.* , 2012).

The pseudo-code shows that the extraction process is performed in three steps: first identifying *explicit* relations, then identifying *implicit*, *AltLex*, *EntRel*, and *NoRel* relations, and finally identifying *attribution* relations. In order to achieve the parsing task following the steps described in Figure 2.8, five components were created. First a connective classifier attempts to identify 100 cue phrases, and modified versions of these, described within the PDTB framework. This component also identifies the context in which cue phrases occur, as well as the part-of-speech tags associated with the cue phrases in order to improve performance. The second component is an argument labeler

which attempts to identify the spans of text covering both arguments, or EDUs, which are to be related. In order to achieve this, the cue phrases are again used as features, as well as some contextual features within the sentence and adjacent sentences. The argument extractor also attempts to determine whether the arguments, or EDUs, are ordered with the main argument first, second, or split in two by the second argument. A third component used is an *explicit* relation classifier. Using the discourse connectives, or cue phrases, identified by the first component, the *explicit* relations are identified. In order to do so accurately, the cue phrases, their part-of-speech tags, and the word preceeding them are used as features. A fourth component attempts to classify *implicit*, *AltLex*, *EntRel*, and *NoRel* relations. In order to achieve this, four features are used: the occurrence of surrounding relations in the context, constituent parse features, dependency parse features, and word-pair features. Finally, an *attribution* span labeler component is used to identify the *attribution* relations. This is done by splitting the text into clauses and determining which of these clauses are *attribution* spans. In order to classify clauses as being *attribution* spans or not, the current, previous and next clauses are evaluated in terms of unigrams, verbs, first and last terms, positions of the clauses in the sentence, and production rules extracted from the current clause.

In order to evaluate the overall system, each component is tested using input devoid of errors. This is acheived using gold standard annotations rather than the input of the previous component. Although the original paper describing the parser provides results obtained through various experimental settings (Lin *et al.* , 2012), Table 2.16 provides a summary of the highest and most relevant results, note that only the F**-score** is provided for all components: **precision**, **recall**, and **accuracy** are provided only when available.

Component	**F-score**
Connective classifier	93.57
Argument labeler	97.94
Arg_1 identifier	86.67
Arg_2 identifier	99.13
Explicit classifier	86.77
Non-explicit classifier	39.63

Table 2.16: End-to-End PDTB Parser Evaluation

As can be seen in Table 2.16, identifying connectives (or cue phrases) is achieved fairly efficiently. The argument labeler component, which determines which EDUs should be considered the nucleus, likewise seems to achieve excellent results. When it comes to identifying the EDUs themselves, it appears that the second argument is typically easier to identify. Still, the reported F-score suggests that this task can be achieved fairly well. Finally, the results associated with the identification of the discourse relations themselves shows, unsurprisingly, that identifying *explicit* discourse relations is much easier than *non-explicit* ones. Obviously, the presence of cue phrases in the case of *explicit* relations makes the task immensely more obvious to the classifier. In the end, it seems that we can rely on the extraction of *explicit* relations to be performed automatically, while *non-explicti*

relations still pose a number of problems. Finally, (Lin *et al.* , 2012) reports overall system F-scores for partial matching of 46.80% with gold standard parses, and of 38.18% with full automation.

2.3.2.5 Faiz & Mercer Parser

Yet another parser using the PDTB framework that we have decided to use for a set of complementary experiments was developed at the University of Western Ontario (Faiz & Mercer, 2013, 2014). A few key elements need to be addressed in order to make proper usage of this particular parser. Firstly, this particular parser only performs automatic extraction of *explicit* discourse relations, and will not identify *implicit*, *AltLex*, and *NoRel* discourse relations. The parser itself relies highly on the appearances of cue phrases in order to achieve this task. One interesting added feature of the Faiz & Mercer Parser, and the reason why we opted to use it, is that it provides models trained with the PDTB corpus, the same corpus used by the end-to-end PDTB parser, and one trained with the BioDRB corpus (see details on both of these corpora in Sections 2.3.1.3 and 2.3.1.4). This allows to perform the automatic extraction task of discourse relations using models trained with one or the other of these corpora. The parser itself is noted to make use of two sets of features for the purpose of extracting *explicit* discourse relations: surface level features, and syntactic features. Some of the more important surface level features are cue phrases and their neighboring terms, and chunk tags. It was experimentally determined by the authors of the parser that, in addition to identifying cue phrases themselves, the extraction of discourse relations was improved by observing the terms that occur directly before and after the cue phrases. It was noted that the appearance of certain terms before or after various candidates for cue phrases is helpful in avoiding to mislabel some occurrences of the cue phrase's string when used in a way that is not marking a discourse relation. For example, it was noted that if a candidate is followed by "of" or "to", it is less likely to be a discourse connective (e.g. "as a result *of*", or "in addition *to*" are noted to usually not be used as cue phrases). The chunk tags, that is phrasal level tags such as VP (verb phrase) or NP (noun phrase), are also noted to provide interesting information for the task at hand. Again, this is noted as being helpful in avoiding mislabeling cue phrase candidates in situations where the string of the cue phrase occurs, but is not used as such by the author of the document. For example, if "when" is directly followed by a verb phrase, it is noted to be less likely to be an actual cue phrase. A second set of features that are noted by (Faiz & Mercer, 2014) to be helpful in discourse relation extraction are syntactic features. Some of the more useful syntactic features noted consider the syntactic siblings of the constituent being considered as the candidate to be a span of text using a specific discourse relation. It was determined that the best results can be achieved by considering all the parent constituents of the one currently being considered, combined to their distance to this same constituent. Another feature use is the part-of-speech tag and syntactic head of the right sibling of the constituent. Finally, a category assigned to the cue phrase is used as a syntactic feature. The cue phrase can be categorize as: subordinating conjunction, coordinate conjunction, discourse adverbial, prepositional phrase, and a phrase taking sentence complements.

In our experiments (see Chapter 3), we have found that the performances, in terms of **accuracy**, **precision**, **recall**, and F**-Scores**, are comparable to those we have obtained by using only the

explicit discourse relations extracted with the end-to-end PDTB discourse parser (Lin *et al.* , 2012). The parser's use of a reimplementation of some of the same methods described in (Lin *et al.* , 2012), combined with our own observations of the output of the parser lead us to believe that the accuracy of both parsers are comparable for the task of *explicit* discourse relation extractions. In their evaluation of cross-domain discourse relation parsing, (Stepanov & Riccardi, 2014) mention this parser as being the only effort, to their knowledge, available with models trained on both the PDTB and the BioDRB corpora. Unfortunately, and as noted in (Stepanov & Riccardi, 2014), very little cross-domain evaluation is available for either of the models. Regardless, the assumption is that using models built from the data available in these two corpora should influence the performance of the parsers, relatively to the type of document that is being parsed. Our own assumption is that extracting discourse relations using the model created from the PDTB corpus should yield better results in the case of documents similar to those of this corpus (namely, newspaper articles), while using the model created from the BioDRB corpus should yield better results in the case of documents similar to those of this latter corpus (namely, research papers).

The results of testing the parser are presented in Table 2.17. Since it is possible to use data from either the PDTB or the BioDRB corpus, results are shown using the PDTB training data on the same corpus, the PDTB training data on the BioDRB corpus, and the BioDRB training data on the same corpus, using cross-validation when using the same corpus for training and testing.

	Accuracy	**Precision**	**Recall**	***F*-score**
PDTB to PDTB	97.53	95.11	96.52	95.81
PDTB to BioDRB	91.43	86.16	75.00	80.19
BioDRB to BioDRB	94.34	85.17	79.80	82.36

Table 2.17: Results of Cross-Validation Evaluation of the Faiz & Mercer Parser

The authors of the parser notice a slight improvement, specifically in terms of **recall**, when using the same corpus as the basis of their training and testing data (Faiz & Mercer, 2014). This results in a small improvement in terms of ***F*-score**. Based on this, they conclude that tailoring the training data used by the parser according to the type of documents from which discourse relations are to be extracted does in fact play a role in improving performance.

2.4 Discourse Rhetoric and Text-types

As it stands now, some work has been performed in order to evaluate the relationship between discourse level rhetoric and either genres or text-types, but, to our knowledge, no parsers have used implementations leveraging such relations. To some extend, the HILDA and Feng & Hirst parsers (Hernault *et al.* , 2010a; Feng & Hirst, 2012) estimate the influence of textual organization by using the distance between a relation and the beginning of the text. Still, a number of papers have already been published discussing the relationship between discourse relations and higher level discourse structures, such as text-types.

Bonnie Webber's investigation of the distinctions of texts of different genres (Webber, 2009) shows that genre does in fact appear to play a role in the distribution of discourse relations. This conclusion is reached through a frequency analysis of the distribution of discourse relations across the PDTB corpus (Prasad *et al.* , 2008). Since the PDTB corpus is composed of documents from the printed press, these were categorized by the author in four different text-types. The bulk of the documents used are labelled as *news*, with 1902 documents. The remaining 208 documents of the corpus are labelled as *essays*, *summaries*, and *letters*. The author observed that in the case of labelling *implicit* relations, especially when such relations appear in between sentences, the genre appears to be a worthwhile feature to investigate. This observation is based on the fact that *implicit* discourse relations are much harder to identify, as can be seen in the evaluation of the PDTB end-to-end Parser (see Section 2.3.2.4). Gaining knowledge about the context in which such discourse relations occur is expected by Webber to provide important clues that should help the extraction of such relations. The author also notes that the various genres appear to share discourse structures across documents. For example, a news article might start by giving an effective summary of its contents, while an essay is less likely to do so. This leads to the hypothesis that we should not only consider the distribution of relations one at a time, but we should also consider sequences of such relations and the influence of genre on the observed patterns.

Another research conducted by the creators of the RST review corpus described in Section 2.3.1.2 studies documents in terms of "stages" (Taboada, 2011). Using the sub-section of the RST review corpus dealing with film reviews, as well as a number of other similar documents from sources such as *Rotten Tomatoes* and *Epinions*, the authors found that such review texts make use of a typical structure. Such reviews are typically organized in five sections: subject matter, plot description, character descriptions, background and evaluation. The authors then argue that these sections could be segmented in two larger communicative goals: description and evaluation, that usually appear in this order. Through their analysis, the authors found that the evaluative sections tend to contain more evaluative and subjective words. The description sections typically contain more temporal connectives, as well as more causal-type connectives. Although they do not study the distribution of discourse relations themselves, the appearance of these particular types of connectives and cue phrases is a good indication that discourse relations are used in particular manners given texts of certain text-types or genres.

Yet another research conducted by (Cardoso *et al.* , 2013) argues that discourse relations can be used as a feature to segment documents on various topics. The conclusion reached in (Cardoso *et al.* , 2013) suggests once again a relationship between discourse relations and overall discourse structures. In order to evaluate their hypothesis, the authors constructed a corpus of 140 texts in Brazilian Portuguese picked from a number of sections of mainstream news agencies. These were selected from the CSTNews corpus (Cardoso *et al.* , 2011), where the same authors performed their own annotation using the RST framework. The 140 documents were then split into topics. To ensure proper annotation of these splits, groups of trained annotators indicated possible boundaries and the ones indicated by the majority of the annotators were selected. The conclusions reached indicate that some relations are more frequent around topic boundaries, while others were never

recorded to occur around these same boundaries. Based on this observation, the authors tested the influence of discourse relations as a feature in the task of automatic topic boundary detection and noted an improvement over the baseline.

In our own work (Bachand *et al.* , 2014), we performed an investigation on the relationship between discourse relations and both the genre and the topics found across documents. To do so, we compared, using log likelihood ratio (Rayson & Garside, 2000), the distribution of discourse relations across various corpora. In an effort to reduce errors as much as possible, we opted to use manually annotated corpora (Carlson *et al.* , 2002; Taboada *et al.* , 2006; Taboada & Grieve, 2004; Prasad *et al.* , 2008, 2011). Since the first three (Carlson *et al.* , 2002; Taboada *et al.* , 2006; Taboada & Grieve, 2004) use the RST framework, while the last two (Prasad *et al.* , 2008, 2011) use the PDTB framework, we also opted to compare corpora based on which of these frameworks they used. Using the original RST-DT corpus, and the RST review corpus, we compared documents of the *news* type to documents of the *review* type. Using the PDTB corpus and the BioDRB corpus, we compared, once again, documents of the *news* type but this time to *research* papers. Through our work, we found that certain discourse relations are more likely to occur in certain types of documents, and further down, in certain sections of these specific documents. For example, *news* items are more likely to contain *attributions* then documents that can be classified as *reviews*. On the other hand, we noticed that *background* relations are statistically more frequent within *reviews*. When performing a similar analysis using the PDTB corpus and the BioDRB corpus, we found that *circumstance* and *background* relations are more commonly used in research papers while *contrast* relations are typically used in *news* items to compare opinions. Overall, the analysis showed that the discourse relations used across various different text-types are used in a manner that can be described as intuitive. As such, we believe that the extraction of discourse relations should provide a good enough feature to properly identify various text-types.

At the moment, most focus on the automatic extraction of discourse relations is on the lowest level of the discourse schema, and attempts to extract discourse relations that hold between two single EDUs. We believe that the ultimate goal of discourse relation extraction should be to identify the entire schema of a document's discourse, rather than being limited to the lowest level, which most of the current research as so far focused on. What we are currently interested in is the possibility of identifying text-types using automatically extracted discourse relations, as opposed to manually identified discourse relations, as we studied in (Bachand *et al.* , 2014). By doing this, we wish to move towards the automatic identification of those larger discourse schemas. We believe that identifying the text-type, which represents the highest level of the discourse structure, can later help in the identification of finer grained discourse structures, all the way down to the lowest level of the overall discourse tree. We believe that this could be especially beneficial to the identification of *implicit* discourse relations, which have been shown to be much more difficult to accurately identify, both in the case of manually annotated corpora (Prasad *et al.* , 2008), and automatic extraction (Lin *et al.* , 2012). A method of automatically identifying discourse relations could, for example, begin by extracting *explicitly* stated discourse relations, which can be done fairly accurately with current discourse relation parsers (for example, the End-to-End PDTB Discourse

Relation Parser can identify them with an F-score of 86.77% (see Section 2.3.2.4)), and subsequently use this information to identify the text-type of the document in order to improve the accuracy of the extraction of *implicit* discourse relations, which the End-to-End Discourse Relation Parser can identify with an F-score of 39.63% (See Section 2.3.2.4 for further details).

In this chapter, we have described some of the commonly used metrics that are used throughout this thesis, discussed the theories of discourse relations, from the concept of EDUs to the way such units relate to one another forming discourse relation structures. We have then described some of the resources currently available in terms of corpora annotated with discourse relations, and automatic discourse relation parsers, and we concluded by discussing some of the currently available work on the relationship between discourse rhetorics and text-types. In the next chapter, we will present the steps undertaken in order to put together and perform the experiments necessary to discuss our claims. We will begin by describing the corpora used to build our own corpus, then we will shortly discuss the discourse relations framework we have decided to employ for our task, we will then describe how we have annotated our corpus with discourse relations as described by this framework, and finally we will describe the various classifying tasks we have performed, both in terms of the feature sets used, and the classifiers that have performed these tasks.

Chapter 3

Experimental Design

In this chapter, we describe our experiments in order to answer our research question of whether text-types can be used as a first step towards detecting larger discourse level schemas in the process of automatic extraction of discourse relations. In order to evaluate the relationship between discourse relations and a document's text-type, we designed a number of classification tasks. To do so, we have gathered documents from various sources in order to build a corpus of over 3,500 documents which are split into seven text-type classes. By using features related to discourse relations, we then attempted to accurately identify the text-type to which each document belongs. It should be noted that some documents may exhibit features that are shared across text-types (e.g. a document could be of the **response** text-type, provide a review, while doing so in a manner that gives it the appearance of a document of the **narrative** text-type, for stylistic reasons), for the sake of simplicity, we assumed that each document has a single classification. In Section 3.1, we describe the corpus we have built for our experiments by giving details related to the various sources used to build our final working corpus. In Section 3.3, we provide a short description of the set of discourse relations used for the annotation of our final working corpus. In Section 3.4, we describe the steps performed in order to obtain the annotations of our final working corpus. In Section 3.5, we describe the various classification tasks we performed for the purpose of determining the relationship between text-types and discourse relations. To do so, we first give a description of the feature sets used in Section 3.5.1, and then a description of the classifiers used in Section 3.5.2.

3.1 Text-Types

We defined seven distinct types to be classified during our experimentation. It should be noted that these text-types were inspired by the description of such concepts from (Rotter & Bendl, 1978), but were subsequently expanded to accommodate for various corpora considered during our research. The definitions of the text-types themselves are based on the work of (Lee, 2001). The text-types are as follows:

Explanation: The **explanation** text-type is composed of documents which give specialized explanations in a specific field of study. Documents such as research papers, academic theses, and studies are included in this category. These type of documents are typically segmented into distinct sections, such as abstract, methodologies, discussion, and results, which each make use of various discourse structures. For example, *temporal* relations are more likely to appear in the methodologies section, which would describe steps taken in a specific order, while *circumstance* relations are typical of the results section, which provide an explanation of the results obtained (Bachand *et al.* , 2014). As described in the next sections, for our investigation, we used the **learned** section of the Brown corpus (Francis & Kucera, 1979) and the BioDRB corpus (Prasad *et al.* , 2011) in order to represent this text-type.

Exposition: The **exposition** text-type is defined as documents used to argue a point of view on a given subject. Documents such as argumentative essays or political speeches are a good example of this particular text-type. In such documents, speakers are expected to state a number of facts in order to build a case over which they can later argue. To represent these, we used the American English portions as well as the A-Level essays from British pupils of the Louvain Corpus of Native English Essays (Granger, 2003).

Procedure: The **procedure** text-type is composed of documents which detail instructions on how to perform a task. Documents such as cooking recipes and instructional manuals are examples of this particular text-type. For this text-type, we have gathered cooking recipes freely available online.

Narrative: The **narrative** documents are those that follow a fictional story. These can span across any genre, while retaining the same general structure. It should be noted, however, that the structure of narrative fiction is not necessarily very strict. In fact, one only has to look at the works of almost any modernist author in English to find examples that defy the conventional "narrative structure" that can be expected from documents of the **narrative** text-type. For our purpose, we limit ourselves to more structurally traditional works of fiction. During our classification task, we used the *fiction*, *mystery*, *sci-fi*, *adventure*, *romance*, and *humour* sections of the Brown corpus (Francis & Kucera, 1979).

Recount: The **recount** text-type contains documents which retell series of events. These are typically news items, recaps of sporting events, or political editorials. Typically, we expect *attribution* relations to appear in such documents, as newspaper articles often report speech that is then attributed to it's authors (Bachand *et al.* , 2014). To account for this, we used the *news* section of the Brown corpus (Francis & Kucera, 1979) and a subsection of the Reuters corpus (Rose *et al.* , 2002).

Report: The **report** text-type is defined as documents which give an impartial account of facts. These are typically documents produced by governments or large international agencies such as the United Nations. Unlike with the **exposition** text-type, such documents should not make subjective claims, but rather state facts which can be subsequently tested or verified.

As described in the next sections, to account for these we use the *report* section of the Brown corpus (Francis & Kucera, 1979) and a subsection of the Open American National Corpus (Ide & Suderman, 2007).

Response: Finally, the **response** text-type contains documents which give a judgemental account on a given subject. Documents such as consumer reviews and art criticism are examples of this text-type. In such documents, we expect relations such as *background*, which would be used by the author to provide background information on claims made to review something either positively or negatively. For our purpose, we consider the *review* section of the Brown corpus (Francis & Kucera, 1979) and the RST review corpus (Taboada & Grieve, 2004; Taboada *et al.* , 2006).

Note that the distinction between text-types is sometimes blurry. For example, a thesis may exhibit characteristics of a *report* as well as an *exposition*. For the sake of simplicity, however, we have assumed a single-class classification for all the documents selected as part of our corpus.

3.2 Corpora

In order to account for our seven text-types, we have gathered seven different corpora which have been rearranged and distributed in our seven text-type categories. The next section provides details on each of these corpora, as well as details on which portions from these were used to be representative of our seven text-types in the final corpus used in our experiments.

3.2.1 Brown Corpus

The Brown corpus (Francis & Kucera, 1979) is composed of 500 documents of roughly 2,000 words each. This corpus was built using documents published in 1961 in the United States. The 500 documents totalling roughly one million words was then separated into 15 categories, each describing a certain genre. The original categories used are: news, editorial, reviews, religion, skill and hobbies, popular lore, belles-lettres, government, learned, fiction, mystery, science-fiction, adventure, romance, and humour. It should be noted that the first nine genres are non-fiction, while the remaining six are fiction. We used these categories in order to split sections of this corpus into the various text-types we are interested in. Five of our seven text-types are represented, in part or in full, by documents of this corpus. Details of which of these documents are used in each one of these categories are shown in Table 3.1.

It should be noted that the **narrative** text-type uses documents from six of the Brown corpus' original genres. Since our hypothesis is that discourse structures are what makes the text-type, while the genre should be established through the use of specialized vocabulary, merging these portions of the corpus together should pose no problem. Within some of the original split on genre from the Brown corpus, we can also witness the occurrence of sub-genres. For example, the *news* section contains items related to the genres of politics, sports, society, finance, culture, etc. Likewise, the *reviews* section contains documents on the subject of theater, music, books, and dance. It therefore

Corpus	Text-Type	No. of Docs	No. of Words
Brown learned	Explanation	80	181,888
Brown fiction	Narrative	29	68,488
Brown mystery	Narrative	24	57,169
Brown sci-fi	Narrative	6	14,470
Brown adventure	Narrative	29	69,342
Brown romance	Narrative	29	70,022
Brown humour	Narrative	9	21,695
Brown news	Recount	44	100,554
Brown government	Report	30	70,117
Brown reviews	Response	17	40,704
	Total	**3,593**	**2,070,638**

Table 3.1: Categorization of Brown Corpus into Text-Types

seems that the definition of genre used by the Brown corpus authors is somewhat unclear, showing once again a difficulty at defining genre against text-type. Thankfully, for our purpose, this seems to work to our advantage as we do not want to separate documents on the basis of genre. In fact, we would rather observe documents of given text-types spanning across a number of genres.

3.2.2 Reuters Corpus

In order to supplement our **recount** text-type, we use a subset of the Reuters Corpus (Rose *et al.* , 2002), specifically, the first 1,250 documents of the corpus. This corpus is composed of 10,788 documents totaling 1.3 million words. These documents were originally categorized in 90 different topics. Again, we choose to ignore the variance in topics as we are interested in the text-type, which we hypothesize should not be influenced by genres or topics. In that respect, the documents are all news items which we associate with the *recount* text-type, regardless of topic. We use the first 1250 documents of the Reuters Corpus in order to obtain adequately equal distributions in terms of numbers of words for each text-type category. Table 3.2 shows the number of documents and words considered in our investigation.

Corpus	Text-Type	No. of Docs	No. of Words
Reuters	Recount	1250	201,155

Table 3.2: Categorization of Reuters Corpus into Text-Types

3.2.3 Open American National Corpus

The Open American National Corpus (Ide & Suderman, 2007) is composed of documents of spoken and written English totalling over 14 million words. For our purpose, we are only interested in a

subset of the *government* section in order to complement our *report* text-type. The documents of this section were gathered online from various American governmental agencies. We used the first 50 documents of this section, as detailed in Table 3.3.

Corpus	**Text-Type**	**No. of Docs**	**No. of Words**
OANC Government	Report	50	252,852

Table 3.3: Categorization of OANC Corpus into Text-Types

3.2.4 Louvain Corpus of Native English Essays

The Louvain Corpus of Native English Essays (LOCNESS) (Granger, 2003) is composed of argumentative essays authored by British and American university level students, and British A-level students. It is composed of a total of 324,304 words from 322 essays on various topics such as literary analysis and argumentation on controversial subjects like abortion, homosexuality, and animal testing. For our purpose, we used a subset of this corpus in order to account for documents of the **exposition** text-type. In particular, we are using the "American Argumentative Essays", "American Literary Mixed Essays", and "British A-Level Essays" sections. Table 3.4 gives some details on the number of documents and words procured from this corpus.

Corpus	**Text-Type**	**No. of Docs**	**No. of Words**
LOCNESS US arg	Exposition	175	161,331
LOCNESS US mixed	Exposition	33	20,125
LOCNESS British A-level	Exposition	114	63,592

Table 3.4: Categorization of LOCNESS Corpus into Text-Types

3.2.5 RST Review Corpus

The RST Review Corpus, as described in Section 2.3.1.1, is composed of 400 documents gathered from *Epinions*, a website aggregating customer reviews on a variety of products. The corpus contains reviews on: books, cars, computers, cookware, hotels, movies, music, and phones. Each of these categories contains 50 documents, half being positive reviews, and half being negative reviews. Again, we assume that the overall structure associated with our **response** text-type should not be influenced by either the sentiment of the review or the product it is reviewing. For these reasons, using a corpus spanning across 8 genres and including equal portions of negative and positive reviews should help us avoid over-fitting issues in that respect. In fact, we once again rely on the variance in vocabulary caused by these aspects of the documents to argue that the structure of the text itself is sufficient in identifying the text-type. For this reason, we are using all 400 documents of this corpus in an effort to represent our text-type as abstractly as possible. Originally, this corpus was manually annotated for discourse relations within the RST framework. We do not use these annotations, but

instead extract them automatically in order to ensure that all information on discourse relations were obtained in the same fashion, regardless of the availability of humanly annotated information. All of the documents found in this corpus are used to represent the **response** text-type. Table 3.5 provides further details on the corpus.

Corpus	**Text-Type**	**No. of Docs**	**No. of Words**
RST Review Corpus	Response	200	303,289

Table 3.5: Categorization of RST Review Corpus into Text-Types

3.2.6 Biomedical Discourse Relation Bank Corpus

The Biomedial Discourse Relation Bank (BioDRB) is yet another corpus of documents manually annotated with discourse relations (Prasad *et al.* , 2011) (see Section 2.3.1.4). Unlike the RST review corpus (see Section 3.2.5), these documents were not annotated within the RST framework. Once again, however, for uniformity purposes, we rely on automatically extracted information to perform our work. The BioDRB corpus is composed of 24 open-access documents. These are biomedical articles acquired from the GENIA corpus (Kim *et al.* , 2003). Each of these documents are separated into sections. These sections are what we would typically expect from research oriented articles, such as: abstract, methodologies, results, discussion, and conclusions. For our purpose, we ignore these separations and take into account each document as a whole. Table 3.6 gives some statistics on this corpus.

Corpus	**Text-Type**	**No. of Docs**	**No. of Words**
BioDRB	Explanation	24	112,483

Table 3.6: Categorization of BioDRB Corpus into Text-Types

3.2.7 Online Recipe Corpus

In order to account for the **procedure** text-type, we have gathered recipes from the website `epicurious.com`. The documents obtained contain, on average, 135 words each. They are typically composed of two broad sections: the list of ingredients, and the steps required for the recipe. For our purpose, we use 1250 of these documents in order to obtain an adequately equal distributions in terms of numbers of words for this given category[1]. Table 3.7 provides statistics on this corpus.

3.2.8 Final Working Corpus

We provide in Table 3.8 a summary of the distribution of documents across our seven text-types. As can be seen in Table 3.8, each of our seven text-types contain a comparable number of tokens,

[1]We would like to thank David Gurnsey for his work on gathering this corpus while working in our research lab during the summer of 2013.

Corpus	Text-Type	No. of Docs	No. of Words
Recipes	Procedure	1250	261,362

Table 3.7: Categorization of Online Recipe Corpus into Text-Types

between 250,000 and 350,000.

Text-Type	Corpora	No. of Docs	Prop.	No. of Words	Prop.
Narrative	Brown	126	3.51%	301,186	14.55%
Recount	Brown, Reuters	1294	36.01%	301,709	14.57%
Report	Brown, OANC	80	2.23%	322,969	15.60%
Procedure	Cooking	1250	34.79%	261,362	12.62%
Exposition	LOCNESS	322	8.96%	245,048	11.83%
Response	Brown, RST Review	417	11.61%	343,993	16.61%
Explanation	Brown, BioDRB	104	2.89%	294,371	14.22%
	Total	**3769**	**100%**	**2,119,219**	**100%**

Table 3.8: Overall Distribution of Corpora per Text-Type

3.3 Discourse Relations

For the purpose of our current investigation, we opted to use the relations of the PDTB framework. The full set of these relations and a description of the framework can be found in Section 2.3.1.3. Using this particular set of relations and the associated framework, we are able to make a distinction between *explicit* and *implicit* relations. This seems like a worthwhile distinction to make as *implicit* discourse relations are much more difficult to identify and might create more noise than they are worth. We also considered the overall performance of the PDTB End-to-End parser (Lin *et al.* , 2012), described in Section 2.3.2.4. For our experiments, we used the 20 higher level discourse relations used in (Prasad *et al.* , 2008), as well as the list of cue phrases used by the parser, as the basis of our feature space. The complete list of cue phrases is listed in Appendix C.

3.4 Parsing Our Working Corpus

In order to build our working corpus, even as some of the corpora used were already annotated with relations from either the PDTB framework or the RST framework, we opted to rely on automatic extractions of discourse relations using the PDTB End-to-End parser (Lin *et al.* , 2012), described in Section 2.3.2.4. Since only some portions of our final corpus includes manual annotations for discourse relations, using these would have been troublesome. Instead, we relied solely on automatically extracted discourse relations in order to obtain a corpus that was produced in the same fashion throughout. Again, this was done in order to insure uniformity of the annotations used.

Table 3.9 shows the distribution of discourse relations automatically extracted with the End-to-End PDTB parser, both *explicit* and *non-explicit*. Overall, 46% of the relations extracted are *explicit*,

Relation	**Explicit**	**Non-explicit**	**Total**	**Distribution**
Alternative	1,013	154	1,167	1.43%
Asynchronous	5,485	1,155	6,640	8.18%
Cause	3,242	23,911	27,153	33.44%
Comparison	7	27	34	0.41%
Concession	982	55	1,037	1.28%
Condition	3,604	9	3,613	4.45%
Conjunction	10,912	7,659	18,571	22.88%
Contingency	0	1	1	0.00%
Contrast	7,025	2,846	9,871	12.16%
Exception	30	4	34	0.04%
Expansion	2	3	5	0.00%
Instantiation	275	2,150	2,425	2.99%
List	38	688	726	0.89%
Pragmatic cause	0	28	28	0.03%
Pragmatic concession	0	0	0	0.0%
Pragmatic condition	1	0	1	0.0%
Pragmatic contrast	0	0	0	0.0%
Restatement	234	5,003	5,237	6.45%
Synchrony	4,586	55	4,641	5.72%
Temporal	0	0	0	0.0%
Total	37,436	43,748	81,184	100%

Table 3.9: Distribution of Discourse Relations Across the Working Corpus

while the other 54% are *non-explicit*. This seems to mimic the distribution of discourse relations across the PDTB corpus which contains a distribution of 45% for *explicit* relations and 55% for *non-explicit* relations (see Section 2.3.1.3). The distribution of the relations themselves provide a few interesting points that are worth discussing. Table 3.10 shows the distribution of discourse relations according to text-types. First, some relations are clearly more commonly extracted than others. For example, *cause* relations account for a third of all relations extracted, while *conjunctions* account for another 22.88%. These two relations together therefore account for half of the overall distribution. Some other relations, on the other hand, are seldom extracted. All of the *pragmatic* relations (*pragmatic cause*, *pragmatic concession*, *pragmatic condition*, and *pragmatic contrast*) appear a total of 29 times overall. This is not surprising, and the rarity of these relations was even noted by the authors of the BioDRB corpus (Prasad *et al.* , 2011), who subsequently eliminated these relations altogether from their adapted version of the framework. Some other relations seem more problematic by their low distributions. The complete absence of extracted *temporal* relations is quite unfortunate

Discourse Relation	Explanation	Exposition	Narrative	Procedure	Recount	Report	Response	Average	Std Dev
Alternative	1.03%	1.09%	1.27%	3.08%	1.12%	2.18%	1.12%	1.56%	0.78%
Asynchronous	6.44%	3.98%	8.64%	25.81%	8.71%	4.92%	4.90%	9.06%	7.62%
Cause	30.82%	44.48%	34.45%	21.47%	17.97%	28.35%	37.77%	30.76%	9.19%
Comparison	0.09%	0.00%	0.00%	0.00%	0.00%	0.33%	0.00%	0.06%	0.12%
Concession	1.89%	1.46%	1.15%	0.13%	1.07%	1.13%	1.59%	1.20%	0.56%
Condition	3.73%	6.97%	3.06%	2.15%	4.53%	5.60%	4.96%	4.43%	1.62%
Conjunction	28.18%	19.25%	21.41%	20.96%	32.29%	27.36%	20.57%	24.29%	4.95%
Contingency	0.00%	0.00%	0.00%	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%
Contrast	13.55%	10.26%	12.03%	6.17%	16.57%	12.94%	13.98%	12.21%	3.29%
Exception	0.05%	0.02%	0.04%	0.00%	0.08%	0.08%	0.04%	0.04%	0.03%
Expansion	0.00%	0.00%	0.00%	0.00%	0.00%	0.03%	0.02%	0.01%	0.01%
Instantiation	2.62%	2.01%	2.88%	5.31%	3.54%	4.94%	2.00%	3.33%	1.34%
List	0.10%	0.11%	0.49%	5.53%	0.61%	0.50%	0.42%	1.11%	1.96%
Pragmatic cause	0.02%	0.02%	0.07%	0.01%	0.00%	0.08%	0.02%	0.03%	0.03%
Pragmatic concession	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic condition	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic contrast	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Restatement	5.48%	4.37%	7.95%	7.59%	6.13%	6.25%	6.59%	6.34%	1.22%
Synchrony	5.97%	5.98%	6.56%	1.79%	7.37%	5.30%	6.02%	5.57%	1.78%
Temporal	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Table 3.10: Distribution of Discourse Relations Across Text-Types

as we would expect this particular type of relations to be commonly used in the **procedure** text-type. This, however, is not too worrisome considering the fairly large distribution of *asynchronous* discourse relations, which as we have seen in Section 2.3.1.3, are a sub-relation of the meta-relation *temporal*. Unfortunately, since we are dealing with automatic extraction of discourse relations based on the PDTB corpus (Prasad *et al.* , 2008) for training, we have to take into account that the parser used was designed for a specific text-type, while we attempt to use its output to properly classify several other text-types. In fact, as mentioned in Section 2.3.1.3, the PDTB corpus is composed of documents from *The Wall Street Journal*, most of which are news items, with occasional editorials, letters, and essays (Webber, 2009). Overall, the vast majority of the documents used as a training set for our parser should be associated with the **recount** text-type. Due to such difficulties, we decided to not only use the extracted discourse relations as features during our experiments, but we also opted to attempt classification based on the set of cue phrases used by the End-to-End PDTB parser (see Appendix C). Although using these will only give us an idea of the usage of *explicit* relations, foregoing *non-explicit* relations altogether, we hoped that some of the errors attributed to the limitations of the parser would be eliminated by this methodology.

In order to evaluate how the text-types of the documents used to train the automatic discourse extractor influences the output of such extractors, we produced two more versions of the corpus using the Faiz & Mercer Parser (see Section 2.3.2.5). This particular parser allows us to perform the extraction of discourse relations with models based on the PDTB corpus or the BioDRB corpus (see Sections 2.3.1.3 and 2.3.1.4). It should be noted that the Faiz & Mercer Parser uses the discourse relations defined in (Prasad *et al.* , 2011), which were adapted from those found in the original PDTB framework (Prasad *et al.* , 2008) (see Sections 2.3.1.4 and 2.3.2.5 for details).

3.5 Classification Task

A number of experiments were designed to evaluate the influence of text-types on the usage of discourse relations. All the experiments performed were done by extracting data using the End-to-End PDTB parser (Lin *et al.* , 2012) described in Section 2.3.2.4, and various simple scripts written in Python. The data gathered was then used to create properly formatted input to be used with WEKA (Hall *et al.* , 2009), a machine learning workbench written in Java and typically used for such classification tasks. WEKA allowed us to perform various experiments using different feature sets and classifiers. In all cases, we performed 10-folds cross validation (Kohavi *et al.* , 1995) in order to gather as precise an evaluation as possible using all the available data. Sections 3.5.1 and 3.5.2 provide further details on the features and classifiers used in our experiments.

3.5.1 Feature Sets

We have used seven different feature sets in order to evaluate our classifiers and provide enough data to allow for discussion.

Bag Of Words: Our first set of features represents documents using the bag-of-words approach. For each document, we extracted tokens after performing a number of preprocessing steps. The preprocessing steps used were: case-folding, stemming, and digits and punctuations removal. Case-folding, and the removal of digits and punctuation marks was straight forward and does not warrant further explanation. Stemming was performed using the implementation of the Porter stemmer (Porter, 1980) found in the Python NLTK package (Bird, 2006). After running these preprocessing steps, we obtained a feature space of 32,346 word stems. This was used as our baseline experiment.

All Relations: The second set of features used in our experiments is the complete set of the 20 discourse relations extracted by the End-to-End PDTB parser (Lin *et al.* , 2012) described in Section 2.3.2.4. They include both the set of *explicit*, and *non-explicit* relations, which include *implicit*, and *altLex* relations, whenever an actual discourse relation was extracted. We rely solely on the identification of the relations, regardless of how these are realized in the documents (that is, regardless of the appearance of a cue phrase or not). This leaves us with 20 different features, as we do not distinguish whether the relations were explicitly stated or not. Given the reported performance of our parser, as noted in Section 2.3.2.4, we expect the use of *non-explicit* relations to create some amount of noise, as the reported performance shows an accuracy of a little under 40% for these particular relations.

Explicit Relations: Our next feature set follows the same idea as we selected only the discourse relations marked as *explicit*. As before, these result in 20 different features. We feel it is important to observe how the use *implicit* and *explicit* discourse relations exclusively differs from using both types in order to allow us to study the influence of the poorer performance recorded in (Lin *et al.* , 2012) when it comes to extracting *implicit* discourse relations.

Implicit Relations: In order to better understand the influence of this particular difficulty, we used the *implicit* discourse relations as the basis of our third feature set. Using the same extracted relations as with our previous feature set, we selected only the discourse relations marked as *implicit*. Once again, we obtain 20 different features, each being a discourse relation from the PDTB framework.

Cue Phrases: In an attempt to minimize the effects of mislabelled discourse relations, we selected the list of cue phrases used by the End-to-End PDTB parser as the basis of our next features set (see Appendix C). Some work has been published in identifying cue phrases automatically, for example, (Laali & Kosseim, 2014) presents a method that relies on parallel corpora and a number of filtering rules to induce a list of cue phrases. However, no method currently available seems to produce perfect lists of cue phrases. For this reason, we decided to rely on a manually created list of cue phrases in the creation of this features set, namely, those described in the PDTB corpus (Prasad *et al.* , 2008) and used by the End-to-End PDTB Parser (Lin *et al.* , 2012). Given that the 100 cue phrases described in the original PDTB corpus can take many alternate forms, in some cases, we ended up with a set of 374 features. Obviously, these cue phrases only give us an indication of the presence of *explicit* discourse relations without actually labeling the relation. In fact, some cue phrases could be associated with more than one discourse relation.

All Relations and Cue Phrases: Our final features set is simply the combination of all discourse relations, *explicit* and *non-explicit*, with the set of cue phrases used by the End-to-End PDTB parser. This gives us a final count of 394 features.

Faiz & Mercer Relations: An added feature set was produced in order to better support one of our claims. Using the Faiz & Mercer Parser (Faiz & Mercer, 2014) (see Section 2.3.2.5), we parsed our entire working corpus again twice, this time using the parser with models built from the PDTB corpus (Prasad *et al.* , 2008) and the BioDRB corpus (Prasad *et al.* , 2011) separately, thus producing two extra sets of automatically annotated discourse relations on the entire working corpus. We then used the discourse relations extracted with each of these models to perform our various classification tasks in order to evaluate the changes in performance recorded. The discourse relations obtained using this parser are limited to the *explicitely* stated discourse relations.

3.5.2 Classifiers

WEKA provides a number of classifiers which can be used to carry out experiments on our various sets of input data. For our purpose, we selected three classifiers which were used to classify documents on the basis of text-types using our six features set. The classifiers are *Multinominal Naïve Bayes*, *Decision Trees*, and *Support Vector Machine*.

3.5.2.1 Multinomial Naïve Bayes

Our first classifier is a multinomial Naïve Bayes classifier (McCallum & Nigam, 1998). This model is typically used in tasks of document classification and constitutes a good starting point for our experiments. In order to perform classification, the Naïve Bayes classifier follows Equations 3.1 and 3.2.

$$P(C_i|D) = P(C_i) \times P(C_i|F) \times P(C_i|F_2) \times ... \times P(C_i|F_n) \tag{3.1}$$

$$classify(D) = \arg\max_i P(C|i) \times P(C_i|F) \times P(C_i|F_2) \times ... \times P(C_i|F_n) \tag{3.2}$$

where:

C_i is class i

D is a document

F are features

More precisely, with a multinomial Naïve Bayes model, the probability of a given document to be of a certain class, in our case of a certain text-type, is given by the prior probability of the class ($P(C_i)$) and the occurrence of certain features independently of each other. For each feature studied within our classification task, the probability of such feature to occur at a certain frequency is calculated for each possible class ($P(C_i|F_n)$). The calculation for each of these features is done independently of any other feature. The probability of all of these features are combined, resulting in a final probability for a document to be associated with each different class, in our case text-types. The probability for each class to be associated with the given document are then compared and the most likely class is selected ($\arg\max$). Unlike with the classic implementation of the Naïve Bayes model where features are noted to appear or not in a given document, the multinomial model takes into account the number of times each feature occurs in the same document. This seems like an important distinction to take advantage of since certain text-types are expected to make use of particular discourse relations at varying frequencies. For example, our previous investigation (Bachand *et al.* , 2014) showed that the usage of *contrast* discourse relations occur in both documents from our newspaper corpus (Prasad *et al.* , 2008) and our biomedical text corpus (Prasad *et al.* , 2011), but such relations are statistically more likely to appear in the former. Given this observation, it would seem insufficient to evaluate the appearances of features on a boolean level.

3.5.2.2 Decision Tree

The second classifier we used in our experiments is WEKA's J48 decision tree classifier. The implementation provided by WEKA is based on the C4.5 algorithm developed by Ross Quinlan (Quinlan, 1993). A decision tree classifier relies on information gain ratio in order to build a tree model where, at each branch, information on the features is used to select how to split the samples provided. At each branch, a feature and a possible associated value is used to determine branching down based on the information gain ratio that the appearance of such a value given this particular feature provides.

For example, the high number of appearances of *cause* discourse relations, as opposed to very few instances of this discourse relation, in a given document could provide information on what the text-type of the document might be. Unlike with the Naïve Bayes model described in Section 3.5.2.1, the features observed are not assumed to be independent as the classification process makes its way down the decision tree, evaluating the influence of various features at every branch. In order to generate the tree used for the classification task, each feature is evaluated in terms of information gain ratio. For all the features which have not been tested yet, the information gain ratio is calculated and the feature with the highest gain ratio is selected as the basis of the next branching. During the classification task, the features observed in a given document are evaluated and used to navigate through the tree until a leaf node is encountered, assigning a given class, in our case the text-type, to the document.

3.5.2.3 Support Vector Machine

Our third classifier is a Support Vector Machine (Cortes & Vapnik, 1995). Support Vector Machines (SVM) are supervised learning algorithms that map data to hyperplanes in such a way that the separation between the data identifying the different classes considered is as wide as possible. An important feature of such classifiers allows to separate the data effectively even when it seems impossible from the placement of data points on the original vector space. This non-linear classification is achieved by applying what is referred to as a "kernel trick" (Aizerman *et al.* , 1964), allowing for non-linear classification. To better illustrate this, consider the following example: Ideally, we wish to separate the data we are classifying in a linear fashion, as illustrated in Figure 3.1. Imagine that

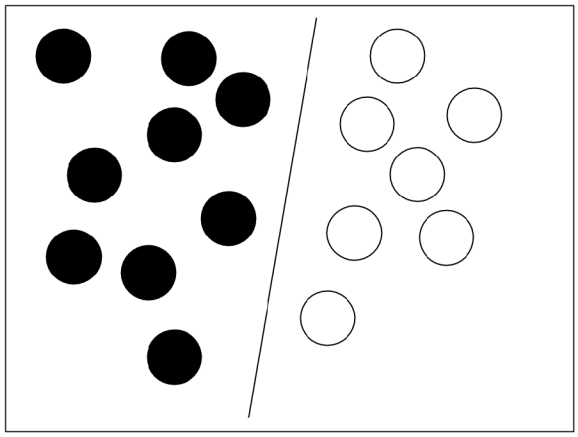

Figure 3.1: Example of a Linear Classification

Figure 3.1 represents data points, the black and white circles, placed on a two dimensional vector space. The line seen in the middle of Figure 3.1 determines which of two classes new data points should be associated with. If the new data lands on the left side of the line, we determine it is one of the black circles, if it lands on the right side of the line, we determine it is one of the white circles.

Unfortunately, such easy to differentiate data is not always available. A situation could arise

where the delimitation between the two groups of circles is not linear, as illustrated in Figure 3.2. In

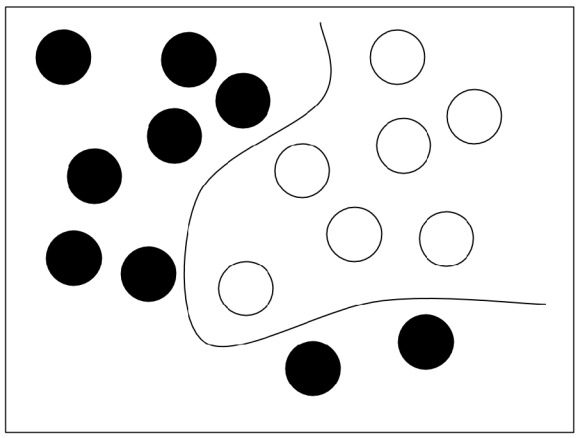

Figure 3.2: Example of a Non-Linear Classification

order to properly separate the two groups of circles, the line which was straight in Figure 3.1 is now curved around the white circles in Figure 3.2. The classification task is therefore no longer linear.

In order to get around this problematic situation, it is possible to map the data into high-dimensional feature spaces, that is, to extend our vector space in such a fashion that it allows for a linear classification. Figure 3.3 illustrates how such a thing is possible. In simple terms, we are

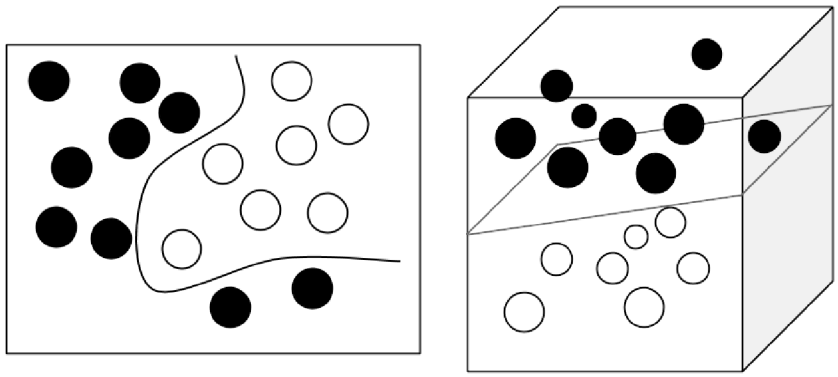

Figure 3.3: Example of Mapping Data Into a High-Dimensional Feature Space

adding a dimension to our feature space thanks to which we can once again separate our two groups

of circles, black and white, in a linear separation. Instead of separating the data against the line seen in Figures 3.1 and 3.2, we now delimit where our data is classified with a new hyperplane, seen separating our three dimensional vector space in Figure 3.3. From the perspective of the original two dimensional feature space, the separation is non-linear, but from the perspective of our high-dimensional feature space, it becomes once again a linear classification task, which is much easier to compute. In order to map our original data to their new locations in our high-dimensional vector space, we apply a function that allows for the separation of the data along our newly created separation hyperplane in a linear fashion without changing their location from the point of view of our original vector space. The data points that provide the most information in relation to our newly created separation hyperplane are the "support vectors" that the name of the classifier refers to. The purpose of using these data points with high information gain is to optimize the classification task, in other words, we only need to consider the most informative data points of our classification system and ignore the less informative data points which might not provide enough information to be worth the computational effort.

In this chapter, we have described our experimental setup. In particular, we described the corpora we have used to build our own text-type corpus, we then discussed the discourse relations framework we opted to use for our work, and we finally described the classification tasks performed in terms of feature sets and classifiers used. In the next chapter, we will discuss the results of our experiments. To do so, we begin by analysing the overall results of our experiments using the different feature sets and classifiers. We then focus our discussion on a detailed analysis of the results obtained using the different feature sets, and finally we discuss the most informative features found through our experiments.

Chapter 4

Results

In this chapter, we provide a discussion of the results obtained using the methodology detailed in Chapter 3. In Section 4.1, we provide an overview of the results obtained through all of our experiments. In Section 4.2, we provide a detailed analysis of some of the more interesting values observed in our experiments. We do so by analysing the results of our baseline experiment in Section 4.2.1, then the results obtained by using our 20 discourse relations in Section 4.2.2, and *explicit* and *implicit* discourse relations separately in Section 4.2.3, then the results obtained using cue phrases in Section 4.2.4, followed by the results of using both discourse relations and cue phrases in Section 4.2.5. Finally we study which features are noted to be the most informative in our various experiments in Section 4.2.7.

4.1 Experiments Overview

In order to perform the evaluation of the influence of a document's text-type on the use of discourse relations, we performed a number of classification tasks. Using the working corpus described in Section 3.2.8, we attempted to classify our documents among the seven text-types defined. To do so, we used the feature sets described in Section 3.5.1 in order to train the three classifiers described in Section 3.5.2 (Multinominal Naïve Bayes, Decision Tree, and Support Vector Machine). Table 4.1 describes the results obtained for each of these combinations in terms of accuracy, precision, recall, and F-score (see Section 2.1 for the definition of these measures).

As can be seen in Table 4.1, the best results are obtained using the bag of words set of features. Although the 25% drop in performance between the bag of words model compared to using discourse relations comes as somewhat of a disappointment, it is not altogether surprising. This particular set of features is much larger, especially compared to our relations feature sets (32,346 vs. 20). The relatively poor performance of our parser, especially with the extraction of *implicit* discourse relations is another factor. Some issues with the corpus used are also contributing to this situation. In the case of the Naïve Bayes classification experiments, we are in fact comparing vector spaces of over 30,000 dimensions, to some composed of 20 dimensions in the case of our relations feature sets. Similarly, it is not entirely surprising to find better results on the bag of words models over

Classifier	Features Used	Features Count	Accuracy	Precision	Recall	F-score
Naïve Bayes	BOW	32,346	0.974	0.975	0.974	0.973
	All relations	20	0.718	0.708	0.718	0.707
	Explicit relations	20	0.623	0.632	0.623	0.587
	Implicit relations	20	0.578	0.614	0.578	0.551
	Cue phrases	374	0.867	0.867	0.867	0.865
	Relations and Cue phrases	394	0.866	0.868	0.866	0.866
Decision Tree	BOW	32,346	0.866	0.955	0.955	0.954
	All relations	20	0.737	0.732	0.737	0.734
	Explicit relations	20	0.664	0.669	0.664	0.660
	Implicit relations	20	0.712	0.698	0.712	0.703
	Cue phrases	374	0.816	0.808	0.815	0.811
	Relations and Cue phrases	394	0.837	0.833	0.836	0.834
Support Vector Machine	BOW	32,346	0.941	0.945	0.941	0.938
	All relations	20	0.733	0.730	0.733	0.720
	Explicit relations	20	0.672	0.681	0.672	0.661
	Implicit relations	20	0.718	0.699	0.718	0.703
	Cue phrases	374	0.833	0.827	0.833	0.824
	Relations and Cue Phrases	394	0.865	0.874	0.865	0.862

Table 4.1: Overall Results of Classification Tasks Using the End-To-End PDTB parser

the classification tasks performed using cue phrases since the cue phrases themselves are part of the dimensions provided by the bag of words features. Some other cue phrases which appear in our corpus and that are not found in the list of cue phrases used for our experiments would also appear in our bag of words model. Again, we are comparing feature sets of different scales, namely over 30,000 features compared to just under 400 features, using all the cue phrases and their modified versions as described in Section 3.5.1. With these differences of size in mind, it seems that our classification tasks using discourse relations and cue phrases are not entirely a failure, even if they are outperformed by the bag of words model. The fact that we do achieve some relatively good results given that our feature sets are orders of magnitude smaller suggests that these particular features provide very interesting information when it comes to the link between discourse relations and text-types, but may not be enough. We will discuss the importance of these features in more detail in Section 4.2.7.

As far as which classifier performs best, it appears that the multinominal Naïve Bayes model either outperforms or is relatively close in terms of performance to both of our other classifiers. This is not too surprising as, although it is quite a simple algorithm by comparison, the Naïve Bayes model is noted to typically perform with the same level of accuracy as other more complex models (Huang *et al.* , 2003), and this without applying any amount of fine tuning. Still, our decision tree classifiers provide us with some important information that will allow us to better identify the features that are most informative, as we will once again see in Section 4.2.7. The Support Vector Machine classifier overall shows similar results as the Naïve Bayes classifier, with the noteable exception that, similarly to the decision tree classifier, it performs better with *implicit* discourse relations than with *explicit* discourse relations (the opposite is observed in the case of the

Naïve Bayes classification task). This is quite surprising. We believe that due to the unreliable performance of the discourse relation parser in the case of *implicit* discourse relation extraction, the results seen here themselves are unreliable. Both our more complex classifiers also fare better in the cases where only discourse relations are used. That is, we see a slight improvement in performance without the naïve hypothesis inherited from our first classifier. When performing the classification task using *explicit* relations, we see accuracy scores of 62.32%, 66.39%, and 67.24% for the Naïve Bayes, decision tree, and support vector machine classifiers respectively. When performing the classification task using *explicit* relations, we see accuracy scores of 57.81%, 71.16%, and 71.78% for the Naïve Bayes, decision tree, and support vector machine classifiers respectively.

4.2 Detailed Analysis

In this section, we provide a detailed analysis of some of the more interesting findings from our experiments. In order to perform an adequate analysis, we rely on the confusion matrices and the details of the calculations of the accuracy per classes provided by WEKA. We first take a look at the results of our baseline experiment in Section 4.2.1. We then discuss the results obtained using discourse relations in Section 4.2.2. In Section 4.2.3, we discuss the differences observed when comparing using *explicit* and *implicit* discourse relations as features. We then discuss the results of our experiments using cue phrases in Section 4.2.4. In Section 4.2.5, we discuss the results of our experiments using the set of features including both discourse relations, both *explicit* and *implicit*, and cue phrases. Finally, in Section 4.2.7, we make an analysis of the most informative features from our various experiments. In order to allow for better comparisons between experiments, for the time we concentrate on the results obtained using the Naïve Bayes classifiers with the various sets of features.

4.2.1 Bag of Words (BOW)

Our first analysis is based on the bag of words feature set, using the Naïve Bayes classifier. Tables 4.2 and 4.3 provide the confusion matrix and the details of the performance in terms of accuracy per class for this particular classifier using these features. It should be noted that we chose to use the results obtained with the Naïve Bayes classifier in the following sections in other to allow for better comparison between feature sets and concentrate on issues unrelated to the choice of classifier.

As can be expected given our final accuracy of 97.38%, the confusion matrix of Table 4.2 shows very good results overall, as can be noticed on the bolded results showing the number of accurately identified documents. Still, we can already notice, once we concentrate on some of the details, that certain text-types are more problematic than others, namely the *explanation* and *report* text-types. In the case of *explanation*, although the precision is quite high, it falls short on recall. It appears that this particular text-type received one of the lowest prior probabilities, with *report* coming in last, as can be seen in Table 4.4.

We can see in Table 4.3 that the *explanation* text-type receives the lowest **true positive** score. Looking into the details of how *explanation* documents are misclassified, we find that these are

	Observed Output						
Text-Type	Exposition	Explanation	Narrative	Procedure	Recount	Response	Report
Exposition	**310**	0	3	0	2	7	0
Explanation	20	**62**	3	0	3	3	13
Narrative	0	0	**125**	0	0	1	0
Procedure	0	0	0	**1248**	2	0	0
Recount	1	0	5	0	**1282**	3	3
Response	6	0	0	0	0	**411**	0
Report	9	1	0	0	8	1	**62**

Table 4.2: Confusion Matrix for Naïve Bayes Trained with BOW Features

Class	**True Positive**	**False Positive**	**Precision**	**Recall**	**F-score**
Exposition	0.963	0.011	0.896	0.963	0.928
Explanation	0.596	0.000	0.984	0.596	0.743
Narrative	0.992	0.003	0.919	0.992	0.954
Procedure	0.998	0.000	1.000	0.998	0.999
Recount	0.991	0.007	0.988	0.991	0.990
Response	0.986	0.005	0.965	0.986	0.975
Report	0.765	0.005	0.795	0.765	0.780
Weighted Avg.	**0.974**	**0.004**	**0.975**	**0.974**	**0.973**

Table 4.3: Detailed Performance by Class for Naïve Bayes Trained with BOW Features

Class	**Prior Probability**
Exposition	0.090
Explanation	0.029
Narrative	0.035
Procedure	0.347
Recount	0.360
Response	0.116
Report	0.022

Table 4.4: Prior Probability of Text-Types Using a Multinominal Naïve Bayes Classifier

most commonly mistaken for documents of the *exposition* or *report* text-types (see Table 4.2). The *explanation* text-type features documents such as research papers. On the other hand, the *exposition* text-type features argumentative essays while the *report* text-type features governmental reports (see Section 3.1 for further details). In both cases, the mistake seems somewhat logical. If we are to compare *explanation* documents to *exposition*, and *report* documents, in all three cases we expect some fairly specialized vocabulary which could easily be shared across all three text-types given

the same or similar genres. For example, classifying a research paper from the bio-medical field, a governmental report on the use of a certain drug, or an argumentative essay discussing the use of a drug could easily cause confusion using our bag of word features set. This problem stems from the fact that the parser's training data is small and only contains documents of the *recount* text-type. The *report* text-type itself is our second source of problems. We can see that such documents are typically misclassified, once again, as *exposition*, and in other cases as *recount*. The documents labelled as *recount* are, once again, newspaper articles. Again, it does not seem too far fetched that governmental reports and newspaper articles could share some of the same vocabulary, especially when dealing with similar subjects or genres.

4.2.2 Discourse Relations

We now turn our attention to the classification of text-types using the discourse relations extracted using the End-to-End Discourse Relation Parser described in Section 2.3.2.4. We begin by looking at the details of our experiment using all discourse relations, both *explicit* and *implicit*. Again, we provide the confusion matrix and details of the accuracy calculations in Tables 4.5 and 4.6.

	Observed Output						
Text-Type	Exposition	Explanation	Narrative	Procedure	Recount	Response	Report
Exposition	**174**	2	9	1	44	90	0
Explanation	19	**16**	8	1	29	26	5
Narrative	4	8	**72**	0	13	28	1
Procedure	6	0	1	**985**	219	13	0
Recount	7	2	2	160	**1112**	10	1
Response	117	10	11	7	78	**189**	5
Report	7	3	1	2	46	10	**11**

Table 4.5: Confusion Matrix for Naïve Bayes Trained with All Discourse Relations

Class	**True Positive**	**False Positive**	**Precision**	**Recall**	**F-score**
Exposition	0.544	0.049	0.521	0.544	0.532
Explanation	0.154	0.007	0.390	0.154	0.221
Narrative	0.571	0.009	0.692	0.571	0.626
Procedure	0.805	0.073	0.852	0.805	0.828
Recount	0.859	0.189	0.722	0.859	0.784
Response	0.453	0.056	0.516	0.453	0.483
Report	0.138	0.003	0.478	0.138	0.214
Weighted Avg.	**0.718**	**0.105**	**0.708**	**0.718**	**0.707**

Table 4.6: Detailed Performance by Class for Naïve Bayes Trained with All Discourse Relations

As expected, given the lower overall accuracy of 71.18%, the confusion matrix shown in Table 4.5

depicts a more noticeable number of problems. By looking at the details provided in Table 4.6, we once again notice that the most problematic text-types appear to be *explanation* and *report*. If we look at the prior probabilities for the text-types, shown in Table 4.4, we find once again that both these text-types rank the lowest, providing part of the answer. On the other hand, we can notice the *narrative* text-type has a similarly low independent probability, while still being identified far more accurately by our classifier. Another possible source of confusion could also stem from the fact that the distinction between text-types is blurry, and a document may in fact exhibit features that can be associated with several types. For example, a document of the *response* text-type could have the appearance of a document of the *narrative* text-type, for stylistic reasons.

If we once again investigate how our documents of the *explanation* text-type are misclassified, we notice that they tend to be labelled as *recount*, *response*, and *exposition*.

4.2.3 Explicit and Implicit Relations

We now discuss the differences in performance between our experiments using only the *explicit* and *implicit* discourse relations as our feature sets. By doing this, we hope to identify the problems encountered in the classification discussed in Section 4.2.2

We first begin by looking at the confusion matrix obtained through our experiment attempting to identify text-types with a Naïve Bayes model classifier using *explicit* discourse relations. These results can are shown in Tables 4.7 and 4.8.

	Observed Output						
Text-Type	Exposition	Explanation	Narrative	Procedure	Recount	Response	Report
Exposition	**122**	27	12	7	139	6	7
Explanation	19	**27**	19	1	26	7	5
Narrative	16	5	**62**	11	21	3	8
Procedure	4	1	0	**808**	411	0	0
Recount	17	8	12	69	**1184**	2	2
Response	65	50	35	8	236	**15**	8
Report	10	7	6	6	46	1	**6**

Table 4.7: Confusion Matrix for Naïve Bayes Trained with Explicit Discourse Relations

A first observation of one of the more problematic points presented by this data, is the high number of false positives obtained by the *recount* text-type. This is especially true in the case of the documents from the *response*, *procedure*, and *exposition* text-types. It appears to be especially problematic with the *response* text-type, where the difference between the number of properly identified documents (15) and documents wrongfully identified as being of the *recount* text-type (236) is quite high. Looking into the details of the classification task provided in Table 4.8, we see that in fact, the *recount* text-type receives the highest proportions of both **true positive** and **false positive**.

These results seem like they should be expected, to some degree, if we again consider the prior probabilities of the various text-types, given in Table 4.4. The *recount* text-type is, in fact, the most

Class	True Positive	False Positive	Precision	Recall	F-score
Exposition	0.381	0.040	0.482	0.381	0.426
Explanation	0.260	0.028	0.216	0.260	0.236
Narrative	0.492	0.024	0.425	0.492	0.456
Procedure	0.660	0.044	0.888	0.660	0.757
Recount	0.915	0.387	0.574	0.915	0.705
Response	0.036	0.006	0.441	0.036	0.067
Report	0.050	0.009	0.118	0.050	0.070
Weighted Avg.	**0.623**	**0.162**	**0.632**	**0.623**	**0.587**

Table 4.8: Detailed Performance by Class for Naïve Bayes Trained with Explicit Discourse Relations

probable class, but is followed closely by the *procedure* text-type. The *procedure* text-type, on the other hand, receives a much more reasonable amount of False Positive values, as can be seen again in Table 4.8. Even as the amount of False Positives for this particular class is the second largest, coming after the *recount* text-type, this more problematic class is the only one that truly seems to stick out in that respect. If we compare this to the numbers obtained in our experiment with all discourse relations, we find from Table 4.6 that the *recount* text type was similarly problematic, although not in such a pronounced fashion (compare a False Positive rate of 18.9% using all discourse relations to 38.7% using only *explicit* discourse relations). This can be attributed in part to the length of the documents in the *recount* section of our corpus. For example, document 14,929 of the Reuters corpus is a document of around 200 words on which the PDTB end-to-end Parser identified a single *explicit* discourse relation.

If we turn our attention to the classification task using a Naïve Bayes model classifier with the *implicit* discourse relations as a feature set, we discover a similar pattern concerning the *recount* text-type. This can be seen, once again, from a confusion matrix, given in Table 4.9. Again, we find from this data that the misidentification of documents as being part of the *recount* text-type is especially problematic in the case of documents from the *procedure* text-type.

	Observed Output						
Text-Type	Exposition	Explanation	Narrative	Procedure	Recount	Response	Report
Exposition	**174**	8	8	4	62	60	4
Explanation	25	**13**	6	7	37	12	4
Narrative	13	12	**31**	4	12	43	11
Procedure	105	0	0	**509**	588	22	0
Recount	17	2	1	40	**1218**	16	0
Response	156	11	10	24	92	**112**	12
Report	13	1	3	15	28	15	**4**

Table 4.9: Confusion Matrix for Naïve Bayes Trained with Implicit Discourse Relations

Likewise, the details provided in Table 4.10 show once again that the *recount* text-type obtains the highest amount of False Positives, which again contrasts with the results obtained with other text-types.

Class	True Positive	False Positive	Precision	Recall	F-score
Exposition	0.544	0.101	0.346	0.544	0.423
Explanation	0.125	0.010	0.271	0.125	0.171
Narrative	0.246	0.008	0.525	0.246	0.335
Procedure	0.416	0.040	0.844	0.416	0.557
Recount	0.941	0.361	0.598	0.941	0.731
Response	0.269	0.053	0.400	0.269	0.321
Report	0.050	0.009	0.114	0.050	0.070
Weighted Avg.	**0.578**	**0.161**	**0.614**	**0.578**	**0.551**

Table 4.10: Detailed Performance by Class for Naïve Bayes Trained with Implicit Discourse Relations

Seeing as that this particular text-type is more frequently falsely classified in all three of the experiments we are currently concerned with (all discourse relations, *explicit* discourse relations, *implicit* discourse relations, all using a Naïve Bayes classifiers), we are lead to believe that this problem is independent from our current concern. Instead, our first assumption is to believe that the root of this problem is with the automatic extraction of the discourse relations. Since our parser is trained on documents from *The Wall Street Journal* (see Section 2.3.2.4 for details), the vast majority of which are of the *recount* text-type (with a few notable exceptions, as noted in (Webber, 2009)), it seems likely that it would favor the type of discourse relations that are expected of this particular text-type during the automatic extraction process. The fact that documents of the *procedure* text-type are the most likely to be erroneously identified and confused for *recount* documents could also be explained by some of the problems encountered with our parser. Referring back to the distribution of discourse relations across our working corpus from Table 3.9 (in Section 3.4), we find that the *temporal* discourse relation is never extracted. This is very unfortunate, especially in the case of documents from the *procedure* text-type where we would expect such discourse relations to be used in a statistically significant manner. In fact, we would expect *procedures* to include a sequential list of steps. In fact, this was one of our conclusion in our previous effort at investigating the link between text-types and discourse structures (Bachand *et al.* , 2014). In order to see the effects of prior probabilities on the **false positive** rate, compare the results obtained using the Naïve Bayes classifier to the SVM classifier, show in Table 4.11. As can be seen, when using either a **decision tree** or **SVM** classifiers, the rate of **false positives** drops in the identification of the *recount* text-type. Although it remains somewhat high (in fact, the **false positive** rate for the *recount* text-type is still the highest with all three classifiers), it does lower by half with both *explicit*, and *implicit* discourse relations. This leads us to the conclusion that the prior probability, which is used with a Naïve Bayes classification, plays a fairly significant role in the misidentification of this particular text-type.

	Explicit		Implicit	
	True Positive	**False Positive**	**True Positive**	**False Positive**
Naïve Bayes	0.915	0.387	0.941	0.361
Decision Tree	0.848	0.213	0.872	0.120
SVM	0.806	0.232	0.896	0.126

Table 4.11: True and False Positive Measures on the Classification of the *Recount* Text-Type Using the Three Different Classifiers

4.2.4 Cue Phrases

As we have seen from Table 4.1 (see Section 4.1), using cue phrases as our set of features in order to train our models generally provides us with better results, in terms of accuracy, precision, recall and F-score, over the models trained with discourse relations themselves. We can clearly see an improvement overall in the classification from the data provided in Table 4.12, which provides us with the confusion matrix of the classification task using a Naïve Bayes model trained with cue phrases (see Sections 3.5.1 and 3.5.2.1).

	Observed Output						
Text-Type	Exposition	Explanation	Narrative	Procedure	Recount	Response	Report
Exposition	**78**	15	3	0	16	38	3
Explanation	12	**67**	0	0	5	11	9
Narrative	0	1	**111**	0	0	14	0
Procedure	0	1	0	**1228**	16	0	5
Recount	9	8	11	76	**1130**	33	27
Response	5	8	28	0	55	**314**	7
Report	3	10	1	0	23	4	**40**

Table 4.12: Confusion Matrix for Naive Bayes Trained with Cue Phrases

Although the bag-of-words model remains the most accurate, as can be seen by comparing the results detailed in Section 4.2.1, the numbers obtained using cue phrases are still telling. Considering that all of the terms used as "cue phrases" are also available in the bag-of-word set of features, it is interesting to see that the model does not perform considerably worse, even while using a very small subset of the features from the baseline model (32,346 words vs. 374 cue phrases, see Section 3.5.1). If we analyse the details of the accuracy measurements provided in Table 4.13 and compare them to those of our baseline model in Table 4.3 (see Section 4.2.1), we can identify which text-types are more problematic.

Given the data provided in Table 4.13, it appears that the *exposition* and *report* text-types are the main contributors to our overall loss in performance. We can notice a drop in the number of **true positive** for both of these text-types, when compared with the data gathered from the bag-of-word model (51% vs. 96% for *exposition* and 49% vs. 77% for *report*). With both text-types,

Class	True Positive	False Positive	Precision	Recall	F-score
Exposition	0.510	0.009	0.729	0.510	0.600
Explanation	0.644	0.013	0.609	0.644	0.626
Narrative	0.881	0.013	0.721	0.881	0.793
Procedure	0.982	0.035	0.942	0.982	0.962
Recount	0.873	0.054	0.908	0.873	0.890
Response	0.753	0.033	0.758	0.753	0.756
Report	0.494	0.015	0.440	0.494	0.465
Weighted Avg.	**0.867**	**0.039**	**0.867**	**0.867**	**0.865**

Table 4.13: Detailed Performance by Class for Naïve Bayes Trained with Cue Phrases

we also notice a loss in performance in terms of precision (73% vs. 90% for *exposition* and 44% vs. 80% for *report*) and recall (51% vs. 96% for *exposition* and 49% vs. 77% for *report*). Obviously, the F-score suffers from a similar loss given that it is directly dependent on these two previous metrics (see Section 2.1). One interesting difference when comparing the loss of performance with these two text-types is that the *exposition* class suffers more in terms of recall, while precision remains relatively high, while the *report* class suffers equaly in terms of precision and recall.

Considering our text-types and the documents which are identified by them, *exposition* being argumentative essays and *report* being governmental reports (see Section 3.1 for details), we can try to argue some of the factors that might influence the results obtained by our experiments. As we have noted, the *exposition* text-type suffers from a loss in performance mostly in terms of recall, while the precision still remains relatively high. If we investigate this a bit further by examining the data provided by the confusion matrix of Table 4.12, the highest false positive value obtained while classifying documents as being of the *exposition* class is obtained by documents of the *response* class. If we look at the prior probability for these classes, as provided in Table 4.4, we see once again the same patterns with an advantage given to the *procedure* and *recount* text-types. The *response* text-type comes in third place, behind those two, while the *exposition* text-type is part of our least probable classes, alongside *explanation* and *narrative*. For the case of documents of the *report* text-type, the highest number of false positive encountered while classifying *report* documents is seen with *recount* documents. Turning to Table 4.4, we find that the prior probabilities of the *response* and *recount* text-types give an advantage to such false classifications. However, given the similar data to the prior probabilities seen using our various feature sets, it does not seem like prior probability plays an important role in the issue at hand. It seems instead that when we trim down on the tokens used for our classification task, moving from the bag-of-word model to the cue phrases model, we remove too many of the distinctive tokens for these particular classes to be properly identified.

It seems that the identification of our text-types is influenced by the fact that many of the documents we use as part of our working corpus are arranged not only by text-type, but tend to share related genres as well (see Section 3.1 for details). In order to better support this claim, we would require a corpus composed of documents within the same genre but with different text-types

(such as restaurant reviews, cooking recipes, news item and governmental reports related to the food industry, research papers on food chemistry, and so on). Unfortunately, building such a corpus is a difficult task given the need for very specific types of documents. Instead, we opted at attempting to use as wide of a range of genres to be present in the documents for each text-type. Still, the documents that comprise our final working corpus tend to share similar genres across text-types which we believe to be the cause of some of the more striking loss in performance we notice with the model trained on cue phrases, compared to our baseline model. Overall, however, the relatively small loss in performance considering the drastic drop in the number of features considered when comparing both of these models suggests that cue phrases are an important feature to consider for this particular classification task.

4.2.5 All Relations and Cue Phrases

We finally come to our final experiment, using both cue phrases and discourse relations (both *explicit* and *implicit*) in order to classify our documents in classes representing text-types. Once again, we look at the details provided by our Naïve Bayes model. Looking at the confusion matrix built from using this model, shown in Table 4.14, the details related to the accuracy seen in Table 4.15, and the prior probability for each text-types, seen in Table 4.4, we notice that the performances observed compared with those seen in Section 4.2.4, using cue phrases alone, are very similar. When comparing the model using cue phrases alone, to the model using both cue phrases and discourse relations, we notice very few differences, in fact. The one more striking difference comes with the *exposition* text type. As we have seen in Section 4.2.4, using only the set of cue phrases as features cause a significant drop in the performance of our model. Adding the automatically extracted discourse relations in our feature space helps with obtaining better performance with this particular classification. Compare the results in terms of **true positives**, we obtain 96% using our baseline experiment, which drops to 51% once we use the cue phrases alone, but finally goes back up to 72% by adding the discourse relations to our model.

	Observed Output						
Text-Type	Exposition	Explanation	Narrative	Procedure	Recount	Response	Report
Exposition	**231**	18	7	0	9	48	7
Explanation	19	**59**	1	0	7	6	12
Narrative	0	2	**112**	0	0	12	0
Procedure	1	0	0	**1196**	19	0	8
Recount	14	11	3	84	**1153**	12	17
Response	42	15	37	0	17	**301**	5
Report	7	10	0	0	27	1	**35**

Table 4.14: Confusion Matrix for Naïve Bayes Trained with All Discourse Relations and Cue Phrases

If we look at the confusion matrix from Table 4.14, we can see that the errors obtained in relation to the *exposition* text-type are mostly associated with the *response* text-type. In fact, the highest

Class	True Positive	False Positive	Precision	Recall	F-score
Exposition	0.722	0.026	0.736	0.722	0.729
Explanation	0.567	0.016	0.513	0.567	0.539
Narrative	0.889	0.014	0.700	0.889	0.783
Procedure	0.977	0.036	0.934	0.977	0.955
Recount	0.891	0.035	0.936	0.891	0.913
Response	0.722	0.025	0.792	0.722	0.755
Report	0.438	0.014	0.417	0.438	0.427
Weighted Avg.	**0.866**	**0.031**	**0.868**	**0.866**	**0.866**

Table 4.15: Detailed Performance by Class for Naïve Bayes Trained with Cue Phrases and All Discourse Relations

number of **false negative** values when classifying *exposition* documents identifies them as *response*. Likewise, the highest number of documents wrongfully identified as being of the *exposition* text-type are documents which should have been classified as *response* text-types. If we compare this to the results obtained with some of our previous experiments, namely using *explicit* and *implicit* discourse relations (see Section 4.2.3), we see that similar patterns were noticeable. The appearance of **false positives** classified as *exposition* when *response* is expected occurs in a similar fashion in both cases. It is somewhat more pronounced in the case of *implicit* relations, but we can account this to the overall poorer performance of the classification task given this particular set of features. In the case of **false positives** classified as response when the *exposition* is expected, however, it appears that the use of *implicit* discourse relations alone is much more troublesome. This could account for some of the loss in performance when comparing our current experiment, using cue phrases and all discourse relations, to the baseline experiment. Again, the extraction of *implicit* discourse relations is difficult, as can be understood by the recorded performance of the End-to-End Discourse Relations parser (see Section 2.3.2.4). This explains in part the significant drop in performance in comparison to the baseline experiment. What appears to be the issue in this particular instance is that, although adding discourse relations to our model does in fact provide a boost in performance, the extraction of *implicit* discourse relations is too difficult of a task and generates some noise. Considering this, it seems that our experiment performs relatively well, suggesting that in fact discourse relations are helpful to our specific classification task.

4.2.6 Summary of the Classification Tasks

Finally, we provide in Table 4.16 an overview of the results described in the previous sections. We provide the F-score obtained in the classification of the seven text-types using the different feature sets considered. Table 4.16 shows that text-types are generally harder to identify accurately than others. For example, the **Explanation**, and **Report** text-types generally obtain lower F-scores. One particularly low score that is noted is related to the identification of **Response** documents using *explicit* discourse relations. While the F-score for this particular text-type is quite high using the

	BOW	All Relations	Explicit Relations	Implicit Relations	Cue Phrases	CP and Relations
Exposition	0.928	0.532	0.426	0.423	0.600	0.729
Explanation	0.743	0.221	0.236	0.171	0.626	0.539
Narrative	0.954	0.626	0.456	0.335	0.793	0.783
Procedure	0.999	0.828	0.757	0.557	0.962	0.955
Recount	0.990	0.784	0.705	0.731	0.890	0.913
Response	0.975	0.483	0.067	0.321	0.756	0.755
Report	0.780	0.214	0.070	0.070	0.465	0.427
Weighted Avg.	**0.973**	**0.707**	**0.587**	**0.511**	**0.865**	**0.866**

Table 4.16: Summary of F-scores for All Text-Types and Feature Sets Using Naïve Bayes Classifiers

BOW model (0.975), we notice that it has obtained the lowest F-score when it comes to classifying it using the *explicit* relations (0.067), followed closely by the **Report** text-type, which appears to cause difficulties across the board.

4.2.6.1 Classification Task Using Different Training Sets

In order to further support our claim that the text-types present in the training set used to model our parser will allow for a better extraction of discourse relations, we attempted the following experiment: using the Faiz & Mercer Parser (Faiz & Mercer, 2014), we performed the automatic extraction of discourse relations on our entire working corpus (see Section 3.2.8) with the parser's training data based on the Penn Discourse Treebank (Prasad *et al.* , 2008) and the BioDRB corpus (Prasad *et al.* , 2011). Based on the comparable performance in terms of accuracy (see Section 2.3.2.5) we assume its performance to be comparable to that of the extraction of *explicit* discourse relations using the End-to-End PDTB parser. We reach this conclusion after comparing the performances of our classifying tasks using all three data sets, the *explicit* discourse relations obtained from the End-to-end PDTB parser and those obtained from the Faiz & Mercer parser with the PDTB and BioDRB models, as can be seen in Table 4.17.

	Accuracy	**Precision**	**Recall**	**F-score**
End-to-end PDTB parser	0.623	0.632	0.623	0.587
Faiz & Mercer Parser (PDTB)	0.620	0.602	0.620	0.562
Faiz & Mercer Parser (BioDRB)	0.655	0.674	0.655	0.633

Table 4.17: Accuracies of Text-Type Classifications Using Naïve Bayes with *explicit* Discourse Relations Extracted from the End-to-end Parser and the Faiz & Mercer Parser

As can be seen from Table 4.17, the results obtained from using the End-to-End PDTB parser and the Faiz & Mercer parser with the PDTB model obtain very similar results, while using the BioDRB model for extraction of discourse relations results in a slight increase in accuracy in our classification task. Since the PDTB corpus is composed of documents of the *recount* text-type, while the BioDRB corpus is made up of documents of the *explanation* text-type, we propose that using these two models yield extraction results of varying performances. That is, we expect that extracting discourse relations using the PDTB training set to be done more effectively on documents of the *recount* text-type, while using the BioDRB training corpus allows for better extractions of discourse relations on the documents of the *explanation* text-type. Because of this, performing the subsequent classification tasks between our seven text-types using the relations extracted using the PDTB training model is expected to perform better on the documents of the *recount* text-type, while the data obtained using the BioDRB training set is expected to perform better on our *explanation* documents. Table 4.18 details the performances recorded using these two data sets.

Explanation	**Trained on PDTB**			**Trained on BioDRB**		
	Precision	**Recall**	***F*-score**	**Precision**	**Recall**	***F*-Score**
Naïve Bayes	0.000	0.000	0.000	0.621	0.173	0.271
J48 Decision Tree	0.272	0.269	0.271	0.438	0.471	0.454
SVM	0.387	0.279	0.324	0.412	0.519	0.460
Recount	**Trained on PDTB**			**Trained on BioDRB**		
	Precision	**Recall**	***F*-score**	**Precision**	**Recall**	***F*-Score**
Naïve Bayes	0.514	0.964	0.670	0.567	0.907	0.698
J48 Decision Tree	0.666	0.886	0.761	0.666	0.809	0.730
SVM	0.675	0.884	0.765	0.674	0.833	0.803

Table 4.18: Accuracy of *explanation* and *recount* Text-Types Classification Using *explicit* Discourse Relations Extracted with the Faiz & Mercer Parser Using Both Models

Table 4.18 shows the accuracy scores obtained using discourse relations extracted using the both the PDTB model and the BioDRM model. Again, the documents of the PDTB corpus are of the *recount* text-type. As expected, the classification of documents of the *explanation* text-type is much more difficult than it is for the documents of the same text-type as our training model. In fact, when it comes to using the **Naïve Bayes** classifier, our experiment was unable to correctly identify any of the documents from that particular text-type. This can be partially attributed to the low **prior probability** of this particular text-type, as can be seen in Table 4.19, detailing the probability distribution of the various text-types using the Faiz & Mercer parser trained on the PDTB model. Using our other two classifiers, the **J48 decision tree** and the **Support Vector Machine** (see Section 3.5.2), we still appear to obtain rather low results, with ***F*-Scores** of 0.271 and 0.324, respectively. In the case of classifying documents of the *recount* text-type, we obtain much better results, both in terms of **recall** and **precision**, and as a result, ***F*-Scores**.

Turning to Table 4.18 again, which details the accuracy of our classification task on documents of

Class	Prior Probability
Explanation	0.029
Exposition	0.090
Narrative	0.035
Procedure	0.347
Recount	0.360
Report	0.023
Response	0.116

Table 4.19: Prior Probability of Text-Types Using Multinominal Naïve Bayes Classifier with the Faiz & Mercer Parser Trained on PDTB

the *explanation* and *recount* text-types, we can find the results of the classification task in terms of **precision**, **recall**, and ***F*-score** obtained while using data trained on the BioDRB model. While the accuracy of the classification task of documents of the *explanation* text-type is still not that high, we do notice a clear improvement. The **Naïve Bayes** classification task improves from 0.000 to 0.271, while the **J48 decision tree** classification task nearly doubles its ***F*-Scores**, and the **Support Vector Machine** classification task sees an imporovement from 0.324 to 0.460. The classification of documents from the *recount* text-type, on the other hand, does not seem to vary much. In some cases, we notice a slight gain or loss in **precision** or **recall**, but there does not seem to be any statistically significant change.

Although an improvement on the classification of the documents of the *recount* text-type when using the discourse relations extracted with the PDTB training set would have better supported our claims, the clear improvement noted in the classification of documents of the *explanation* text-type using the discourse relations parsed with the BioDRB training set does show the results we were expecting. This leads us to believe, once again, that knowing which text-type the document from which discourse relations are extracted will improve the extraction process. As a result, thanks to a fewer errors in identifying discourse relations, the subsequent task of identifying the text-type of documents is done more accurately.

4.2.7 Most Informative Features

We now investigate in further detail the influence of the various features in terms of how informative they are in the process of our classification tasks. It should be noted that the data discussed in this section is once again based on the classification tasks made using the End-to-End PDTB parser (see Section 2.3.2.4 for details). For each feature evaluated, a **information gain ratio**[1] value allows to sort the features from most informative to least informative. The **information gain ratio** value is provided by WEKA's *information gain attribute evaluator* function (Hall *et al.* , 2009) It should

[1]Several other measures could have been used such as mutual information gain, or cross entropy, but we chose information gain ratio as it was readily available in WEKA, provides us with enough information to perform an analysis, and is typically used in Decision Tree classifiers, such as the ones we have used in some of our experiments.

also be noted that the **information gain ratio** provided and calculated by WEKA provides a very similar overall image as the one that can be seen by investigating the decision trees computed by the J48 decision tree algorithm, also provided by WEKA (Hall *et al.* , 2009) (see Section 3.5.2.2). This is not surprising, as decision trees rely on similar information gain metrics to determine the ordering, and therefore, the importance of the various features used in classification tasks (see Section 3.5.2.2 for further details). The calculation of **information gain ratio** we use in this section seemed like a better choice as far as evaluating the relevant information, compared to reproducing entire decision trees which can be rather difficult to interpret, especially given large numbers of features. In order to further discuss and justify some of the findings presented in Section 4.2, we look at the most informative features obtained when considering our baseline model and cue phrases in Section 4.2.7.1, and using discourse relations in Section 4.2.7.2.

4.2.7.1 Bag-of-Words and Cue Phrases

We began our detailed exploration of nost informative features by investigating the importance of the appearance of cue phrases in our classification tasks. In order to evaluate this, we considered the experiments involving the feature sets of cue phrases and our bag-of-words model (see Section 3.5.1). In Table 4.20, the 100 most informative cue phrases are listed, from most informative to least informative; while Table 4.21 provides the list of the top 100 most informative tokens using our bag-of-words model. It should be noted that some amount of preprocessing was used in the creation of our list of features for the bag-of-words model, such as stemming (see the full list of steps in Section 3.5.1), and that the bag-of-words model only considers single tokens, whereas some cue phrases contain expressions composed of several words. Still, it is possible to find within our bag-of-words model some of the cue phrases considered with our other feature set. Those particular tokens are identifiable as the values in bold in Table 4.21.

If we compare the values presented in Tables 4.20 and 4.21, we find that a number of cue phrases are present within our top 100 most informative tokens. Again, the features used in the bag-of-words model are single tokens, which means that some more cue phrases might still play an important role in the classification tasks performed. For example, the cue phrase "that is" is the 39^{th} most informative feature using our cue phrase model, while the single token "that" ranked 4^{th} in our bag-of-words model. It should also be noted that some tokens found to be very informative in our bag-of-words model seem to suggest, once again, that the working corpus used could benefit from having a more even distribution of particular genres. That is, each text-type category should have documents that are related to the same set of genres in order to avoid the classification to be influenced by genres. This problem explains the appearance in Table 4.21 of tokens such as "cup" in the 6^{th} most informative feature, "tablespoon" as the 7^{th}, "salt" as the 25^{th}, and "teaspoon" as the 26^{th} (the tokens related to cooking are displayed in italic throughout Table 4.21). It appears that these particular tokens where very informative in identifying documents related to the genre of food. As it turns out, our working corpus contained much more documents related to this particular genre in our *procedure* class, which contained cooking recipes (see Section 3.1).

Setting aside this issue, we find that a number of tokens that can be associated with cue phrases

Rank	Info. Gain	Cue Phrase	Rank	Info. Gain	Cue Phrase	Rank	Info. Gain	Cue Phrase
1	0.77	and	34	0.14	yet	67	0.06	whereas
2	0.75	as	35	0.14	however	68	0.05	indeed
3	0.69	or	36	0.14	as well	69	0.05	otherwise
4	0.65	so	37	0.12	rather	70	0.05	certainly
5	0.61	until	38	0.12	thus	71	0.05	now that
6	0.59	for	39	0.12	that is	72	0.05	anyway
7	0.55	not	40	0.11	therefore	73	0.05	unless
8	0.54	if	41	0.11	further	74	0.05	earlier
9	0.44	now	42	0.11	else	75	0.05	this time
10	0.44	but	43	0.11	although	76	0.05	just as
11	0.4	when	44	0.1	next	77	0.05	fortunately
12	0.39	only	45	0.1	second	78	0.04	at the same time
13	0.38	though	46	0.1	of course	79	0.04	as though
14	0.34	too	47	0.1	later	80	0.04	at that
15	0.33	where	48	0.1	true	81	0.04	obviously
16	0.29	then	49	0.09	in that	82	0.04	at once
17	0.29	even	50	0.09	actually	83	0.04	specifically
18	0.29	also	51	0.09	except	84	0.04	at first
19	0.27	back	52	0.09	last	85	0.04	in fact
20	0.26	just	53	0.09	you know	86	0.04	overall
21	0.26	because	54	0.08	finally	87	0.04	even though
22	0.25	nor	55	0.08	at least	88	0.03	merely
23	0.24	after	56	0.08	soon	89	0.03	the moment
24	0.23	well	57	0.08	for example	90	0.03	at last
25	0.22	once	58	0.08	so that	91	0.03	besides
26	0.22	again	59	0.07	suppose	92	0.03	much as
27	0.21	first	60	0.07	suddenly	93	0.03	lest
28	0.19	before	61	0.07	as if	94	0.03	after all
29	0.18	still	62	0.07	instead	95	0.03	even if
30	0.18	since	63	0.07	not only	96	0.03	in turn
31	0.18	till	64	0.06	clearly	97	0.03	third
32	0.17	either	65	0.06	in addition	98	0.03	with that
33	0.17	while	66	0.06	previously	99	0.03	whenever
						100	0.03	moreover

Table 4.20: 100 Most Informative Cue Phrases Ordered by Information Gain

are ranked in our top 100 tokens, within the bag-of-words feature set. This indicates that these cue phrases do play an important role in identifying our text-types. Our top cue phrase in Table 4.20, "and", is ranked second in Table 4.21, only below "the" which has an almost identical ranking score (note that the scores were rounded to two decimal points in both tables). The simple token "the" actually appears in some of our cue phrases ("at the same time", "the moment"), and so do many

Rank	Info. Gain	Token	Rank	Info. Gain	Token	Rank	Info. Gain	Token
1	0.77	the	34	0.45	on	67	0.36	more
2	0.77	**and**	35	0.46	one	68	0.36	*chop*
3	0.73	to	36	0.44	t	69	0.36	who
4	0.71	that	37	0.44	which	70	0.36	*mixtur*
5	0.7	of	38	0.44	all	71	0.36	than
6	0.68	*cup*	39	0.44	had	72	0.35	inch
7	0.66	*tablespoon*	40	0.44	would	73	0.35	do
8	0.65	in	41	0.44	had	74	0.35	heat
9	0.65	thi	42	0.43	been	75	0.35	**even**
10	0.62	have	43	0.43	**when**	76	0.35	lt
11	0.61	a	44	0.42	**or**	77	0.34	mani
12	0.61	with	45	0.42	said	78	0.34	out
13	0.6	is	46	0.42	an	79	0.34	get
14	0.59	i	47	0.41	no	80	0.34	hi
15	0.59	**until**	48	0.41	they	81	0.34	larg
16	0.59	wa	49	0.41	these	82	0.33	**also**
17	0.58	be	50	0.41	their	83	0.33	into
18	0.56	**as**	51	0.4	**if**	84	0.32	way
19	0.55	minut	52	0.4	what	85	0.32	he
20	0.55	it	53	0.4	onli	86	0.32	over
21	0.54	**not**	54	0.4	at	87	0.32	go
22	0.54	**for**	55	0.39	were	88	0.31	me
23	0.52	**but**	56	0.39	other	89	0.31	most
24	0.51	s	57	0.39	apo	90	0.31	**becaus**
25	0.51	*salt*	58	0.39	*stir*	91	0.31	were
26	0.49	*teaspoon*	59	0.38	my	92	0.3	**then**
27	0.48	there	60	0.37	can	93	0.3	up
28	0.47	you	61	0.37	time	94	0.29	how
29	0.46	**so**	62	0.37	*pepper*	95	0.29	see
30	0.46	are	63	0.37	add	96	0.29	look
31	0.46	*bowl*	64	0.37	peopl	97	0.29	**now**
32	0.46	like	65	0.37	from	98	0.28	some
33	0.45	by	66	0.36	could	99	0.28	**just**
						100	0.28	much

Table 4.21: 100 Most Informative Tokens Using the Bag-of-Words Model Ordered by Information Gain

of the most informative tokens presented in Table 4.21. The fourth most informative token, "that", appears in the cue phrases "that is" and "at that", the fifth most informative token, "of", appears in the cue phrase "of course", the eighth most informative token, "in", appears in the cue phrases "in that" and "in fact", and so on. In total, 16 of the 100 tokens are an actual cue phrases composed

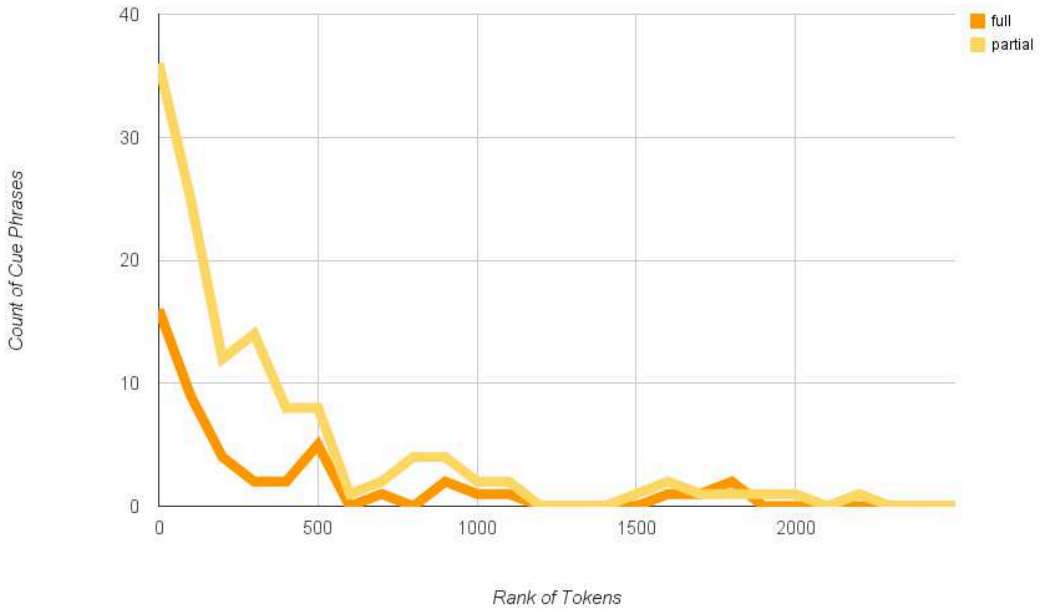

Figure 4.1: Count of Full and Partial Cue Phrases Found in the BOW Features, Ordered by Information Gain.

of a single word (or a stemmed version of that word), while another 36 are sub-strings found in our full list of cue phrases (such as "the", "that", "of", etc). On the other hand, we find that 8 of these tokens are clearly related to cooking. Looking over Table 4.21, it actually appears that, for the notable exception tokens clearly related to the genre of food, many of the tokens that are highly ranked in terms of information gain are words that could be used to mark discourse relations, within certain contexts at least. In fact, if we count the number of tokens that are full or partial cue phrases that appear within each slice of 100 most informative features, we notice that cue phrases are often very informative. Table 4.1 shows the count of the full and partial cue phrases found within the first 2,500 most informative features.

4.2.7.2 Discourse Relations

We provide the list of all discourse relations (see Section 3.3 for details), sorted according to their **information gain ratio**, the distribution of each discourse relations for each one of our text-types, and the **average distribution** across text-types, along with the **standard deviation**. These sets of values are presented for all discourse relations in Table 4.22, for *explicit* discourse relations in Table 4.23, and for all *non-explicit* discourse relations in Table 4.24. It should be noted that *non-explicit* discourse relations include *implicit*, and *altLex* discourse relations (see Section 3.5.1 for further details).

We begin by providing in Table 4.22 the distribution of each discourse relation, whether it be *explicit* or otherwise, for each of our text-types ordered by information gain. If we study the data provided in Table 4.22, we first find that *cause* is the most informative feature. This is not very

Discourse Relation	Info. Gain	Explanation	Exposition	Narrative	Procedure	Recount	Report	Response	Average	Std Dev
Cause	0.92	30.82%	44.48%	34.45%	21.47%	17.97%	28.35%	37.77%	30.76%	9.19%
Conjunction	0.67	28.18%	19.25%	21.41%	20.96%	32.29%	27.36%	20.57%	24.29%	4.95%
Contrast	0.60	13.55%	10.26%	12.03%	6.17%	16.57%	12.94%	13.98%	12.21%	3.29%
Synchrony	0.51	5.97%	5.98%	6.56%	1.79%	7.37%	5.30%	6.02%	5.57%	1.78%
Asynchronous	0.47	6.44%	3.98%	8.64%	25.81%	8.71%	4.92%	4.90%	9.06%	7.62%
Restatement	0.47	5.48%	4.37%	7.95%	7.59%	6.13%	6.25%	6.59%	6.34%	1.22%
Condition	0.39	3.73%	6.97%	3.06%	2.15%	4.53%	5.60%	4.96%	4.43%	1.62%
Concession	0.26	1.89%	1.46%	1.15%	0.13%	1.07%	1.13%	1.59%	1.20%	0.56%
Instantiation	0.25	2.62%	2.01%	2.88%	5.31%	3.54%	4.94%	2.00%	3.33%	1.34%
Alternative	0.16	1.03%	1.09%	1.27%	3.08%	1.12%	2.18%	1.12%	1.56%	0.78%
List	0.11	0.10%	0.11%	0.49%	5.53%	0.61%	0.50%	0.42%	1.11%	1.96%
Pragmatic_cause	0.02	0.02%	0.02%	0.07%	0.01%	0.00%	0.08%	0.02%	0.03%	0.03%
Comparison	0.01	0.09%	0.00%	0.00%	0.00%	0.00%	0.33%	0.00%	0.06%	0.12%
Exception	0.01	0.05%	0.02%	0.04%	0.00%	0.08%	0.08%	0.04%	0.04%	0.03%
Expansion	0.01	0.00%	0.00%	0.00%	0.00%	0.00%	0.03%	0.02%	0.01%	0.01%
Contingency	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%
Temporal	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic_concession	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic_contrast	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic_condition	0.00	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Table 4.22: Distribution of All Discourse Relations Across Text-Types Ordered by Information Gain

surprising when considering that the **standard deviation** for this particular discourse relation is the highest. It appears that the usage of *cause* discourse relations is very frequent in our documents of the *exposition* text-type, while they are much less frequent, with less than half the distribution, in documents of the *recount* and *procedure* text-types. A second observation that can be made by the data provided in Table 4.22 is that many of the discourse relations used seem to have very little influence on the classification tasks. It is not surprising for some of these relations which were rarely or never extracted by our parser (see Section 3.4 for details), which certainly explains the last nine discourse relations presented which all have average distributions below 1%, the last five being completely absent from our working corpus. Some of these discourse relations were in fact noted by (Prasad *et al.* , 2011) to be problematic and were subsequently reclassified or rearranged in the creation of the BioDRB corpus (see Section 2.3.1.4). A few more detailed observations can also be made using the data presented in Table 4.22 in relation to which discourse relations are likely the most helpful at identifying particular text types. A very clear example of this is seen in the distribution of the *asynchronous* discourse relation. It appears that this particular discourse relation is much more frequently used in the documents of the *procedure* text-type, with a distribution of over 25%, compared to a little under 9% in documents of the *recount* text-type, the class with the second highest frequency of this discourse relation. This suggests that particular discourse relations can be utilised to identify particular text-types, but might not be as relevant when attempting to classify documents with less statistically interesting usage of these particular discourse relations. This again follows the findings of our previous investigation (Bachand *et al.* , 2014). This is unfortunate, in a way, considering that some of the discourse relations we would expect to be very informative are badly indentified by our parser. For example, we would expect the *temporal* discourse relation to have a higher distribution in documents of the *procedure* text-type, but due to the performance of the End-to-End PDTB parser, such relations are never extracted. Another problem we have encountered with the End-to-End PDTB parser is that *implicit* relations are inherently more difficult

to identify. In order to better evaluate the influence of this drawback, the next paragraph will study the distribution of discourse relations and their associated information gain for discourse relations expressed *explicitly* and *implicitly* separately.

Table 4.23 provides the same set of discourse relations as Table 4.22, once again sorted according to their **information gain ratio**, with the most informative features on top, but this time we only consider *explicit* discourse relations. A first observation we can make from the data presented

Discourse Relation	Info. Gain	Explanation	Exposition	Narrative	Procedure	Recount	Report	Response	Average	Std Dev
Synchrony	0.50	10.51%	13.14%	15.88%	4.91%	11.25%	12.74%	12.19%	11.52%	3.38%
Contrast	0.46	20.75%	18.33%	20.19%	0.89%	23.54%	15.21%	21.60%	17.22%	7.66%
Condition	0.39	6.61%	15.31%	7.55%	6.37%	7.05%	13.51%	10.00%	9.49%	3.61%
Conjunction	0.38	29.06%	28.98%	32.33%	10.39%	32.22%	31.32%	30.57%	27.84%	7.81%
Asynchronous	0.33	9.64%	7.72%	15.09%	67.56%	12.08%	8.58%	8.64%	18.47%	21.79%
Concession	0.25	3.39%	3.15%	2.46%	0.04%	1.60%	2.72%	3.10%	2.35%	1.18%
Alternative	0.11	1.84%	2.25%	1.77%	8.86%	1.72%	5.03%	1.97%	3.35%	2.70%
Cause	0.09	16.18%	8.73%	3.64%	0.96%	9.21%	7.56%	10.80%	8.15%	4.92%
Instantiation	0.08	1.24%	1.88%	0.19%	0.00%	0.18%	1.44%	0.34%	0.75%	0.75%
Restatement	0.03	0.62%	0.39%	0.70%	0.00%	0.80%	1.54%	0.54%	0.66%	0.47%
Exception	0.01	0.06%	0.05%	0.11%	0.00%	0.06%	0.19%	0.09%	0.08%	0.06%
Pragmatic_cause	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic_condition	0.00	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic_contrast	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00	0.00%	0.00%
Expansion	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.03%	0.01%	0.01%	0.01%
Temporal	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Contingency	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic_concession	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
List	0.00	0.02%	0.07%	0.08%	0.04%	0.30%	0.06%	0.15%	0.10%	0.10%
Comparison	0.00	0.09%	0.00%	0.00%	0.00%	0.00%	0.06%	0.00%	0.02%	0.04%

Table 4.23: Distribution of *Explicit* Discourse Relations Across Text-Types Ordered by Information Gain

in Table 4.23 is that the *cause* discourse relation provides much less information when we only consider *explicitly* stated occurrences of this particular discourse relation. In fact, it appears that this discourse relation tends to be used *implicitly* far more often. The data in Table 4.23 shows that the *cause* discourse relation as an average distribution of 8.15% when *explicitly* stated, compared to 50.26% when *implicitly* stated, as can be seen from Table 4.24. Considering that the *cause* discourse relation is the most informative feature when using all discourse relations, regardless of whether or not they were *explicitly* stated, and that *implicit* discourse relations are noted to be far more difficult to identify (see Section 2.3.2.4 for details), the appearance of the *cause* discourse relation as the most informative overall (*explicit* or otherwise) is somewhat problematic. Given that *implicitly* stated *cause* discourse relation account for more than half of the *implicit* discourse relations obtained by parsing the documents with the End-to-End PDTB parser (Lin *et al.* , 2012), and that, according to (Prasad *et al.* , 2008), the manually annotated Penn Discourse Treebank only has around 25% of its *implicit* relation in the *contingency* meta-category, of which *cause* is only one of the discourse relations (see Section 2.3.1.3), it is clear that the output of the End-to-End PDTB parser is faulty to some some extent. Such errors are sure to create some difficulties in the task at hand, and can be argued to explain some of the loss in accuracy observed when comparing the results obtained from our baseline experiments to those made using discourse relations feature sets.

Investigating once again the data provided by Tables 4.22, 4.23, and 4.24, we can see how

Discourse Relation	Info. Gain	Explanation	Exposition	Narrative	Procedure	Recount	Report	Response	Average	Std Dev
Cause	0.83	49.21%	74.37%	55.41%	31.90%	33.76%	43.07%	64.10%	50.26%	15.61%
Restatement	0.42	11.60%	7.69%	12.88%	11.45%	15.76%	9.59%	12.49%	11.64%	2.55%
Conjunction	0.31	27.08%	11.11%	13.98%	26.33%	32.43%	24.55%	10.82%	20.90%	8.75%
Instantiation	0.19	4.35%	2.12%	4.71%	8.01%	9.62%	7.41%	3.62%	5.69%	2.69%
Contrast	0.18	4.52%	3.52%	6.48%	8.86%	4.01%	11.34%	6.54%	6.47%	2.83%
List	0.11	0.21%	0.15%	0.78%	8.32%	1.18%	0.82%	0.68%	1.73%	2.93%
Asynchronous	0.09	2.42%	0.86%	4.24%	4.59%	2.62%	2.34%	1.25%	2.62%	1.39%
Alternative	0.05	0.02%	0.11%	0.93%	0.14%	0.05%	0.16%	0.28%	0.24%	0.32%
Pragmatic_cause	0.02	0.05%	0.03%	0.12%	0.02%	0.00%	0.14%	0.04%	0.06%	0.05%
Concession	0.02	0.00%	0.04%	0.26%	0.18%	0.11%	0.00%	0.11%	0.10%	0.10%
Comparison	0.01	0.09%	0.00%	0.00%	0.00%	0.00%	0.52%	0.00%	0.09%	0.19%
Condition	0.00	0.12%	0.00%	0.00%	0.00%	0.00%	0.00%	0.04%	0.02%	0.05%
Pragmatic_condition	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Exception	0.00	0.05%	0.00%	0.00%	0.00%	0.11%	0.00%	0.00%	0.02%	0.04%
Synchrony	0.00	0.28%	0.00%	0.22%	0.20%	0.37%	0.02%	0.00%	0.16%	0.15%
Pragmatic_contrast	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Pragmatic_concession	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Contingency	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.02%	0.00%	0.00%	0.01%
Temporal	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Expansion	0.00	0.00%	0.00%	0.00%	0.00%	0.00%	0.02%	0.02%	0.01%	0.01%

Table 4.24: Distribution of Non-*explicit* Discourse Relations Across Text-Types Ordered by Information Gain

some discourse relations can be used to identify specific text-types. For example, considering all discourse relations (Table 4.22), it appears that the *cause* discourse relation is much more frequently used in documents of the *exposition* text-type than documents of the *recount* text-type (44.48% vs. 17.97%). This seems intuitively right considering that the *exposition* documents argue in favor or against a point of view and would naturally use causality as a means of proving a point, while documents from the *recount* text-type should be expected to state facts in a more straight forward fashion. This same observation can probably explain the distribution of the *conjunction* discourse relations which appear with a distribution of 19.25% in the *exposition* documents and 32.29% in the *recount* documents. The documents of the *procedure* text-type stand out with their high distribution of *asynchronous* discourse relations. The Penn Discourse Treebank annotation guidelines (Prasad *et al.* , 2008) describe this specific discourse relations as describing situations that are ordered through time. This is expected in a document of the *procedure* text-type, as the steps of such procedures should typically be performed in sequence. We can also notice a lower distribution of *contrast* discourse relations which do not seem to be very useful in the case of documents of the *procedure* text-type. Similarly, if we look at the data related to *explicit* discourse relations from Table 4.23, we find that *asynchrony* is significantly more frequently used in documents of the *procedure* text-type, while *contrast* discourse relations are significantly more rare compared to the text-types. When still considering *explicit* discourse relations, we also find the documents of the *procedure* text-type to stand out in their less frequent usage of *synchrony*, *conjunction*, and *concession* discourse relations. Yet another discoure relation, but this time stated in a *non-explicit* manner that helps identify documents of the *procedure* text-type is the *list* discourse relation. With a distribution of 8.32% compared to an average of 1.73% for documents of all text-types, as can be seen in Table 4.24, this discourse relation is used in an intuitive manner given this particular text-type. We would, in fact, expect lists to occur in *procedural* texts, with such discursive schemas as lists of ingredients

in the case of cooking recipes, or lists of tools and parts in assembly manuals, etc. Still looking at *implicit* discourse relations of Table 4.24, we find that *cause* discourse relations are significantly more frequent in documents of the *exposition* text-type. This explains the higher distribution we have already observed in Table 4.22 for this particular discourse relation. For this same text-type, we also find that *implicitly* stated *conjunction* discourse relations are more rare than with most other text-types, but particularly compared to documents from the *recount* text-type. This can be explained intuitively, once again, as documents of the first of these two text-types tend to argue in greater length, while documents of the *report* text-type are expected to state facts in a more straight forward fashion. Similar distributions can be seen as ways of better identifying text-types, although they might not stand out as much. For example, looking at the *explicit* distributions of Table 4.23, we find that the *condition* discourse relations are more common in documents of the *exposition*, and *report* text-types, while also more rarely used in documents of the *explanation*, *narrative*, *procedure*, and *recount* text-types. This distribution does not stand out as clearly as some of the others we have previously discussed, but should still provide information that is helpful to our classification tasks. Similarly, the *non-explicit* use of *conjunction* discourse relations appear more frequently in documents of the *explanation*, *procedure*, *recount*, and *report* text-types, while being noticeably less frequent in documents of the *exposition*, *narrative*, and *response* text-types.

Overall, our investigation of the distribution of discourse relations, and their ranking according to the amount of information gain they provide in our classification tasks, seem to indicate that, although some amount of errors are present due to a number of factors (genre is unevenly represented across the corpus, the automatic extraction of *implicit* discourse relations is difficult, the automatic parser used is only using training data from documents we classify in some of our text-types), discourse relations provide an interesting avenue in identifying a document's text-type. A more subtle selection of which discourse relations should be used as features might allow to reduce the feature space, while conserving the accuracy achieved with all the discourse relations. If, for example, we were to select discourse relations for which the information gain ratio is above 0.09, based on the data provided in Table 4.22, it seems we could limit ourselves to using only the top eleven most informative features. Looking back at the numbers provided by Tables 4.23 and 4.24, we find that the top seven and six features are above this same threshold. Obviously, some further investigation should be performed in order to determine exactly where the threshold should lie.

In this chapter, we have discussed the results of our experiments. More precisely, we began by providing a general overview of the results obtained throughtout the different classification tasks performed. We then focused on a detailed analysis of the results obtained with each of the feature sets we chose to employ, and finally we discussed the most informative features discovered through our investigations. In the next chapter, we will discuss the conclusions and findings in some more details, namely, we will discuss the influence of the feature sets used on the classification tasks, the influence of the classifiers used on those same classification tasks, the variance in performance observed across text-types, and finally discuss the most informative features. We will then conclude by discussing the possible future avenues that now present themselves to us in light of our current

research.

Chapter 5

Conclusions and Future Work

In this final chapter, we summarize in Section 5.1 the findings and claims made from studying the data acquired during our various experiments, and subsequently discuss in Section 5.2 the future work that could be undertaken in light of these findings.

5.1 Findings

Our various classification experiments (see Chapter 4) allow us to state a number of findings. These findings can be separated in four broad categories. We first present findings related to the influence of using our various feature sets over the classification tasks, we then present findings related to the influence of the classifier used to perform these experiments, we then present the findings related to the influence of the text-types themselves on those same classification tasks, and finally we present the findings uncovered based on the most informative features in terms of information gain.

5.1.1 Influence of Feature Sets Used in the Classification Tasks

We now discuss the findings related to the feature sets used in our various experiments. A first of these findings, and perhaps the most unfortunate, is that the text-type classification task is performed with the highest accuracy when using the baseline bag-of-words model. This can be clearly seen as the accuracy recorded throughout our various experiments is consistently higher with the our bag-of-words model. Still, we find that the bag-of-word model provides the most informative features for our classification task at this point. This is not exactly surprising, however, as using such a model not only covers, at least in part, the cue phrases used to identify discourse relations, but also allows to consider a number of other features that might play a role in our classification tasks. For example, the fact that the genre of the text from our working corpus varies widely across text-types, but not necessarily in an even fashion, is believed to influence the results observed. This influence can be seen in Table 4.21 as many terms directly related to cooking are noted to be very informational. Ideally, we would like to avoid such a situation altogether by representing genre across our different text-types more evenly. Once again, we believe that such terms are helpful

at identifying genre while the way the clauses in which those terms are found through discourse relations tell us about the text-type itself. The fact that we are also considering a feature space of over 30,000 features in the case of our bag-of-words model with one of 20 features when using discourse relations (*explicit* or otherwise), is sure to play a role in classification tasks. It is also important to acknowledge that the extraction of discourse relations is performed automatically, we are therefore certain that at least some noise is introduced to our systems, making the classification tasks more difficult. Knowing this, we find that discourse relations still provide an interesting amount of information related to text-types. Given that we are using a feature space that is several orders of magnitude smaller then the one used in our baseline experiments (20 features when using discourse relations vs. over 30,000 when using the BOW model), data that is inherently noisy due to the automatic extraction of discourse relations, and a discourse relation framework that has its own flaws, we find that the accuracy of our classification tasks are satisfying. We believe that the loss in accuracy observed, in light of these limitations, is justifiable, and we therefore believe that the data presented supports our claim of a link between the concepts of text-types and discourse structures. Given a feature space that is, by comparison, very limited, we find that the accuracy of above 70% in all of our experiments using discourse relations to classify text-types is very promising. Another finding we observed is how adding cue phrases directly into our feature space, as opposed to simply using them to identify *explicit* discourse relations, is helpful in regaining some of the accuracy lost compared to our baseline experiments. However, we find that the baseline still remains to be the most accurate of our experiments. We believe that the use of particular cue phrases within certain text-types, in our working corpus at least, makes these particular cue phrases very valuable in terms of information gain. In other words, two different cue phrases might be used to identify a single discourse relation, but one might be more commonly used within documents identified in our working corpus of a certain text-type, while the other is more commonly used in documents of another text-type. We also find that using a combination of discourse relations and cue phrases does not provide a noticeable change in the accuracy of our classification task. This is not surprising, once again, as the cue phrases themselves are instrumental in the identification of *explicit* discourse relations, and *implicit* discourse relations are noted to be much more difficult to identify automatically. For these reasons, we believe that whatever gain, in terms of information, is provided by adding discourse relations to our cue phrases feature space is offset by some of the issues that are introduced at the same time, such as mislabelled discourse relations which are noted to occur frequently when identifying *implicit* discourse relations. Finally, the corpus itself is problematic in the sense that the separation of documents across text-types also often follows a separation across genres. Since we expect particular genres to share vocabulary items, having documents of distinct genre in the various text-type categories gives an unfair advantage to the BOW model.

5.1.2 Influence of Classifiers in the Classification Tasks

We now discuss the observations made in relation to the types of classifiers used during our various experiments. As far as which of the three classifiers is better suited for our specific task, we find that the multinominal Naïve Bayes classifier yields competitive results, regardless of the fact that

it is the simplest method utilized in our experiments. It should be noted, however, that the prior probability of each text-type can influence the classification task using the multinominal Naïve Bayes classifier. This was noted to be the source of a few issues related to specific classes, for example the *recount* text-type obtained half as many false positives when using a Decision Tree or SVM classifier, compared to the use of the multinominal Naïve Bayes classifier. The difference in performances recorded using a Decision Tree classifier were, for the most part, not statistically significant. One observation that could be argued is that the use of a Decision Tree does show a slight improvement in accuracy, compared to the use of our Naïve Bayes classifier, when using discourse relations. This could be used to argue that discourse relations need to be considered as part of greater schemas, rather than by relying on the *naïve* assumption of conditional independence between features. However, a Decision Tree does not exactly identify such schemas, but only suggests that the appearance of certain discourse relations together, at some point in the document, and not necessarily in sequence, are likely to provide information that is useful to our classification task. In other words, although the fact that relation R_1 is more likely to appear in a document that also contains a higher number of relations R_2, it does not tell us anything about the specific sequence of these two relations, we still treat our documents as a *bag of relations*. Still, the improvement recorded could be argued to suggest that the identification of schemas is helpful in the automatic extraction of discourse relations. However, since the improvement recorded when using a Decision Tree classifier with discourse relations as a feature set is somewhat minimal (see Table 4.1), such a claim remains to be further supported. It would be much more interesting, in fact, to attempt the classification task using bigrams of discourse relations. Such bigrams should describe sequences of discourse relations that occur one after the other. On the other hand, building these bigrams can prove problematic as the discourse schemas are tree structures, as opposed to linearly represented, which makes things problematic in the case of embedded discourse relations. Another finding we observe is that the Support Vector Machine classification tasks we have performed show similar results to the Decision Tree classification tasks. The overall performance seems to be closer to those seen with the Naïve Bayes classification when it comes to the baseline experiment using the bag-of-words model, but the experiments using discourse relations outperform, once again, the Naïve Bayes classifier. The results for both the Decision Tree and Support Vector Machine classification tasks are very close.

5.1.3 Variance in Performance Across Text-Types

We now discuss our findings related to the differences observed in the classification of specific text-types, that is, how the text-type itself influences the performance of our various systems. Results show that certain text-types are harder to identify or differentiate from others, regardless of the classifiers or the feature sets. For example, our experiments have shown that documents of the *explanation* and *report* text-types are always more problematic to identify, as can be seen in their various precision, recall and F-score measures. For example, the documents of the *response* text-type appear to be significantly more difficult to identify using *explicit* discourse relations. It would be interesting at this point to test whether these same difficulties are shared amongst human annotators

in a manual task of text-type classification. On the other hand, some text-types see their classification affected more greatly by the use of specific feature sets. The same observation cannot be made based on the data gathered using *implicit* discourse relations, where the relative accuracy seems to follow the same pattern observed with our baseline experiments. As far as the automatic extraction of discourse relations is concerned, we find that knowing the text-type in advance of the extraction task can be helpful. By comparing the accuracy recorded while classifying documents of the *explanation* and *recount* text-types with discourse relations extracted with a parser that can be tailored to extracting discourse relations from documents associated with these text-types, we find that the classification of documents of the *explanation* text-type are more easily identifiable when obtaining the data based on a model built from documents of this same text-type. This leads us to claim that knowing which text-type the document falls under is helpful in improving the automatic extraction of discourse relations. This claim, however, relies on the assumption that discourse relations do in fact provide enough information to perform our text-type classification task adequately. We believe that some of our other findings justify this assumption.

5.1.4 Most Informative Features

We now discuss the findings related to the most informative features discovered throughout our various experiments. A first finding worth mentioning is that, when comparing the most informative features of our bag-of-words model to those of our cue phrases model, many of the words making up cue phrases rank in the top 100 most informative tokens of the bag-of-words experiments. Due to the various pre-processing steps employed to extract the tokens that make up our bag-of-words model, the cue phrases themselves seldom appear *as is* in our bag-of-words feature set, but they are visible enough to support the claim that *explicit* discourse relations, at least, are helpful in our task. For example, the token "that", which can be used to produce the cue phrase "that is" is noted as being significantly informative in our bag-of-words model. Some cue phrases that are unigrams, on the other hand, can be found in both the bag-of-words model and the cue phrase model. For example, the words "and" and "because" both appear as part of our top 100 most informative features, for the bag-of-words and cue phrase models. In the end, 16 cue phrases clearly represented in the top 100 tokens provide the most information gain, and an extra 38 that could become cue phrases if associated with other tokens. This suggests that, although the larger amount of information made available through the bag-of-words model helps in making the baseline experiments the most accurate of the experiments, it remains that the cue phrases themselves play an important role in properly identifying the text-types.

Another finding related to information gain is that some discourse relations are more likely to occur *explicitly*, while others are more likely to occur *implicitely*. This can be problematic, as we find that the overall most informative discourse relation is *cause*, but when comparing the information gain provided by this particular discourse relation as it is expressed *explicitely* or otherwise, we find that it is much more informative when done *implicitly*. This, in itself should not be too problematic if it was not for the fact that the automatic extraction of *implicit* discourse relations is a particularly difficult one, and we therefore expect a fair amount of errors to incur in the process.

As far as which of the discourse relations are the most informative, a number of interesting observations need to be noted. We found through our experiments that the usage of the *cause* discourse relation is very frequent in documents of the *exposition* text-type, while they are less then half as frequent in documents of the *recount*, and *procedure* text-types. We also found that many of the discourse relation types are not very helpful in identifying the text-type of a document. In fact, it appears that only considering the ten most informative of these features would suffice.

5.2 Future Work

In order to identify some of the possible future avenues that our work has identified, we considered the various findings detailed in Chapter 4 and summarized in Section 5.1.

5.2.1 Using Text-Types to Tailor Discourse Parsers

Our first set of findings are related to the influence of the feature sets used on the accuracy of the text-type classification task. Since the classification task itself performs better using our baseline experiment, as seen in Section 4.1, there seems to be little advantage in going through the task of extracting discourse relations for this particular classification task. On the other hand, the results recorded has shown a relation between the text-type of documents and the usage of discourse relations. For this reason, we believe that identifying the text-type of a document could become an important step in subsequently extracting the discourse relations. An approach that could be attempted is to identify the text-type through the use of cue phrases, which can be achieved fairly easily and accurately given the smaller feature space employed compared to the bag-of-words model, and subsequently using this information to better identify the discourse relations of the document. We feel that this information could be helpful, especially when identifying *implicit* discourse relations, which have been noted in Section 2.3.2 to be much more difficult to identify than *explicit* discourse relations.

5.2.2 Identifying n-grams of Discourse Relations Across Text-Types

We also believe that an interesting avenue of research, as far as feature sets used, is the appearence of n-grams of discourse relations. Currently, we have studied the distribution of discourse relations with what we would describe as a bag-of-relations model. We believe that the distribution of bigrams of discourse relations should show an even stronger difference in their usage across text-types. Given the tree structure of discourse schemas, however, defining how exactly these bigrams should be created remains to be studied.

5.2.3 Identification of Higher Level Discourse Schemas

Based on some of our observations, we believe that the automatic extraction of discourse relations can be improved through the identification of larger discourse schemas, as original described in (Mann & Thompson, 1987) with the original Rhetorical Structure Theory. We believe that, much

like with the problem of understanding the meaning of a word depends both on the word itself and the context in which it occurs, the full understanding of a discourse structure requires an understanding of the discourse schema in which it occurs. When it comes to the automatic extraction of discourse relations, we believe that classification should take into account context. That is, we once again believe that discourse schemas (such as n-grams of discourse relations) should be considered during the identification of discourse relations. Because of this, research could take into account the importance of these schemas during the automatic extraction of discourse relations. We believe, once again, that this would be helpful mostly in the case of *implicit* discourse relations, since *explicit* discourse relations are already fairly easy to identify, given an appropriate list of cue phrases. One important factor to consider for such a task, however, is that the discourse schemas are complex tree structures, with further schemas and discourse relations embedded within one another. Because of this, using methods such a Hidden Markov Models, as described in (Eddy, 1996), would most likely be inadequate, as it assumes a linear sequence of discourse relations. In reality, the tree structures formed with discourse relations are too complex to be properly identified through such a model, but the idea of considering context during the extraction process is a worthwhile avenue for future endeavors. Still, some of our own preliminary work on the topic shows that this is a promising avenue, as we noticed that (parent, child) bigrams of discourse relations tend to vary quite a bit across documents of various sources. Unfortunately, the corpora used at the time did not exactly consider text-type as the deciding factor for the classification of these documents, and the variances observed could not be used to further argue effectively the relation between discourse relations and text-types.

In a practical sense, the ability to identify larger discourse schemas could be useful in identifying portions of a discourse where relevant information to a task might occur. For example, if were to perform discourse analysis on a research paper, being able to identify sections such as the methodology section due to it's resemblance to a text of *procedure* text-type could help an application in honing in on relevant information.

5.2.4 Prior Identification of Text-Types

Concerning the variances in performances when comparing text-types and how effectively they are being identified, we feel that once again, the results observed suggest that the overall performance of automatic discourse relation extraction could benefit from having some understanding of the text-type of a document. We believe that ideally having an understanding of the larger discourse schemas at play within a document is really the key to achieving a full understanding of the document on a discourse level. The idea of being able to identify text-types is a step in that direction. Future work should then be aimed at utilizing this information, which seems reasonably easy to obtain, in order to move towards identifying larger discourse schemas. These discourse schemas should themselves be composed of discourse relations which appear with varying distributions. This information should then be used in the automatic extraction of the discourse relations. In other words, knowing that a document is of a given text-type should allow us to assume that the document is composed of some distinct discourse schemas, which themselves are more or less likely to be composed of specific

discourse relations. We believe that the Faiz & Mercer Parser (see Section 2.3.2.5) is taking a step in the right direction in that sense, as it allows to use training data from the two currently available manually annotated corpora for PDTB styled discourse relations. We also feel that an interesting avenue that should be tested with future discourse relation parsers is to consider sequences of discourse relations, as opposed to treating the documents as a *bag-of-relations*. By that, we mean that studying the occurrence of n-grams of discourse relations will most likely be helpful in the future as better performing automatic discourse relation parsers are created.

5.2.5 Corpus

Another source of issues that deserves to be addressed in the future is related to the corpus used. One of the reasons we believe the baseline bag-of-words model to be so effective is that the genres of the documents found in our corpus are not evenly distributed enough. That is, while we attempt to classify our documents according to text-types, the particular use of specific vocabulary terms, which we would associate with a particular genre, provides better clues during the classification task due to the fact that documents from the various text-type categories also share the same genre (e.g. many documents of the *recount* text-type category are news paper articles related to politics). In order to get around this issue, it would be necessary to either extend the corpus by adding documents from all possible genres to each text-type category, or limit the corpus to documents of a single genre. Ideally, to do this if our *procedure* text-type category contains cooking recipes, we should find restaurant reviews under the *response* text-type category, research papers related to diets in the *explanation* text-type category, and so on. This, unfortunately, is quite an undertaking, which we have attempted to achieve during the construction of our corpus, however, the task proved very difficult and we still believe that there is quite a bit of room left for improvement in this area. A future work should be to create a new corpus, either from scratch or by expending the one used in our investigation, in order to make sure all documents classified in the various text-types show a large variety of genres, alleviating the advantage given to the bag-of-words model. We also feel that the some problems could be caused by the assumption that all documents in our corpus can only be associated with a single text-type. In fact, it is possible that certain documents exhibit features that could be attributed to several text-types, for example, a document providing the *review* of a customer product could also be presented as a *narrative*, detailing the consumer's experience with the product.

Finally, our analysis of the most informative features in the task of classification of documents according to their text-types shows that cue phrases are still an interesting feature available when it comes to dealing with discourse relations. Given that, once a list of cue phrases has been obtained, finding occurrences of these cue phrases is a trivial task, it seems logical that future endeavors should utilize this inexpensive feature whenever appropriate. In the end, the identification of higher level discourse structures, such as the ones represented by text-types, in the task of automatically extracting discourse relations seem to be beneficial in that respect. However, this thesis has only explored this nascent field of study with the few currently available tools and resources available. However, much more work should be performed and much more data should be shared within

the scientific community for this fascinating field to be developed. This why we wholeheartily thank (Marcu *et al.* , 1999; Soricut & Marcu, 2003; Hernault *et al.* , 2010a; Feng & Hirst, 2012; Lin *et al.* , 2012; Faiz & Mercer, 2014; Carlson *et al.* , 2002; Taboada *et al.* , 2006; Prasad *et al.* , 2008, 2011; Taboada *et al.* , 2006) for their work, the tools, and the data they have made available publicly through the years.

Bibliography

Abney, Steven, Flickenger, S, Gdaniec, Claudia, Grishman, C, Harrison, Philip, Hindle, Don, Ingria, Robert, Jelinek, Fred, Klavans, Judith, Liberman, Mark, *et al.* . 1991. Procedure for quantitatively comparing the syntactic coverage of English grammars. *Pages 306–311 of: Proceedings of the Workshop on Speech and Natural Language*. HLT '91. Association for Computational Linguistics, Pacific Grove, California.

Aizerman, A, Braverman, Emmanuel M, & Rozoner, LI. 1964. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and remote control*, **25**, 821–837.

Bachand, Félix-Hervé, Davoodi, Elnaz, & Kosseim, Leila. 2014. An Investigation on the Influence of Genres and Textual Organisation on the Use of Discourse Relations. *Pages 454–468 of: Computational Linguistics and Intelligent Text Processing (CICLing 2014)*. Kathmandu, Nepal: Springer.

Bird, Steven. 2006. NLTK: The natural language toolkit. *Pages 69–72 of: Proceedings of the COLING/ACL on Interactive presentation sessions*. Association for Computational Linguistics, Sydney, Australia.

Cardoso, Paula CF, Maziero, Erick G, Castro Jorge, MLC, Seno, Eloize MR, Di Felippo, Ariani, Rino, Lucia HM, Nunes, Maria das Graças V, & Pardo, Thiago AS. 2011. CSTNews- A Discourse-Annotated Corpus for Single and Multi-Document Summarization of News Texts in Brazilian Portuguese. *Pages 88–105 of: Proceedings of the 3rd RST Brazilian Meeting*.

Cardoso, Paula CF, Taboada, Maite, & Pardo, Thiago AS. 2013. On the contribution of discourse structure to topic segmentation. *Pages 92–96 of: Proceedings of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*.

Carletta, Jean. 1996. Assessing agreement on classification tasks: The Kappa statistic. *Computational Linguistics*, **22**(2), 249–254.

Carlson, Lynn, Marcu, Daniel, & Okurowski, Mary Ellen. 2001. Building a discourse-tagged corpus in the framework of rhetorical structure theory. *Pages 1–10 of: Proceedings of the Second SIGdial Workshop on Discourse and Dialogue - Volume 16*. SIGDIAL '01.

Carlson, Lynn, Okurowski, Mary Ellen, & Marcu, Daniel. 2002. *RST Discourse Treebank*. Linguistic Data Consortium, University of Pennsylvania.

Charniak, Eugene. 2000. A maximum-entropy-inspired parser. *Pages 132–139 of: Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*. NAACL 2000.

Chomsky, N. 1957. *Syntactic structures*. Janua linguarum: Series minor. Mouton.

Chomsky, Noam. 2002. *Syntactic structures*. Walter de Gruyter.

Cortes, Corinna, & Vapnik, Vladimir. 1995. Support-vector networks. *Machine learning*, **20**(3), 273–297.

da Cunha, Iria, & Iruskieta, Mikel. 2010. Comparing rhetorical structures in different languages: The influence of translation strategies. *Discourse Studies*, **12**(5), 563–598.

De Marneffe, Marie-Catherine, MacCartney, Bill, Manning, Christopher D, *et al.* . 2006. Generating typed dependency parses from phrase structure parses. *Pages 449–454 of: Proceedings of LREC*, vol. 6.

Eddy, Sean R. 1996. Hidden markov models. *Current opinion in structural biology*, **6**(3), 361–365.

Faiz, Syeed Ibn, & Mercer, Robert E. 2013. Identifying explicit discourse connectives in text. *Pages 64–76 of: Advances in Artificial Intelligence*. Lecture Notes in Computer Science, vol. 7884. Springer.

Faiz, Syeed Ibn, & Mercer, Robert E. 2014. Extracting Higher Order Relations From Biomedical Text. *Proceeding of 52nd Annual Meeting of the Association for Computational Linguistics, Baltimore*, 112–113.

Fellbaum, Christiane. 1999. *WordNet*. Lecture Notes in Artificial Intelligence, vol. 10. Wiley Online Library.

Feng, Vanessa Wei, & Hirst, Graeme. 2012. Text-level discourse parsing with rich linguistic features. *Pages 60–68 of: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, Jeju Korea: Volume 1*.

Firth, J.R. 1957. *Papers in linguistics, 1934-1951*. Oxford University Press.

Francis, W Nelson, & Kucera, Henry. 1979. Brown corpus manual. *Brown University Department of Linguistics*.

Granger, Sylviane. 2003. The International Corpus of Learner English: A new resource for foreign language learning and teaching and second language acquisition research. *Tesol Quarterly*, **37**(3), 538–546.

Hall, Mark, Frank, Eibe, Holmes, Geoffrey, Pfahringer, Bernhard, Reutemann, Peter, & Witten, Ian H. 2009. The WEKA data mining software: An update. *ACM SIGKDD explorations newsletter*, **11**(1), 10–18.

Hernault, Hugo, Prendinger, Helmut, Ishizuka, Mitsuru, *et al.* . 2010a. HILDA: A discourse parser using support vector machine classification. *Dialogue & Discourse*, **1**(3).

Hernault, Hugo, Bollegala, Danushka, & Ishizuka, Mitsuru. 2010b. Towards semi-supervised classification of discourse relations using feature correlations. *Pages 55–58 of: Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. SIGDIAL '10. Association for Computational Linguistics, Tokyo, Japan.

Hernault, Hugo, Bollegala, Danushka, & Ishizuka, Mitsuru. 2011. Semi-supervised discourse relation classification with structural learning. *Computational Linguistics and Intelligent Text Processing*, 340–352.

Huang, Jin, Lu, Jingjing, & Ling, Charles X. 2003. Comparing naive Bayes, decision trees, and SVM with AUC and accuracy. *Pages 553–556 of: Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*. IEEE.

Ide, Nancy, & Suderman, Keith. 2007. The open american national corpus (oanc). *Corpus available at* `www.americannationalcorpus.org/OANC/index.html` *(Retrieved on February 6th, 2014)*.

Jackendoff, Ray. 1977. X-bar syntax. *The MIT Press*.

Kim, J-D, Ohta, Tomoko, Tateisi, Yuka, & Tsujii, Junichi. 2003. GENIA corpusa semantically annotated corpus for bio-textmining. *Bioinformatics*, **19**(suppl 1), i180–i182.

Kohavi, Ron, *et al.* . 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. *Pages 1137–1145 of: IJCAI*, vol. 14. San Francisco: Morgan Kaufmann Publishers Inc.

Laali, Majid, & Kosseim, Leila. 2014. Inducing Discourse Connectives from Parallel Texts. *In: The 25th International Conference on Computational Linguistics 2014*. Dublin, Ireland: COLING.

Lavandier, Yves. 2007. *La dramaturgie*. Le clown & l'enfant.

Lee, David YW. 2001. Genres, registers, text types, domains and styles: Clarifying the concepts and navigating a path through the BNC jungle.

Lin, Ziheng, Kan, Min-Yen, & Ng, Hwee Tou. 2009. Recognizing implicit discourse relations in the Penn Discourse Treebank. *Pages 343–351 of: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*. Association for Computational Linguistics.

Lin, Ziheng, Ng, Hwee Tou, & Kan, Min-Yen. 2012. A PDTB-styled end-to-end discourse parser. *Natural Language Engineering*, 1–34.

Magerman, David M. 1995. Statistical decision-tree models for parsing. *Pages 276–283 of: Proceedings of the 33rd annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics.

Mann, William C, & Thompson, Sandra A. 1987. Rhetorical structure theory: A framework for the analysis of texts. *IPRA Papers in Pragmatics*, **1**, 79–105.

Mann, William C, & Thompson, Sandra A. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, **8**(3), 243–281.

Marcu, D. 1999. Instructions for manually annotating the discourse structures of texts. *Available from www.isi.edu/~marcu*.

Marcu, Daniel. 2000. *The theory and practice of discourse parsing and summarization*. MIT Press.

Marcu, Daniel, Amorrortu, Estibaliz, & Romera, Magdalena. 1999. Experiments in constructing a corpus of discourse trees. *Pages 48–57 of: Proceedings of the ACL99 Workshop on Standards and Tools for Discourse Tagging*.

Marcus, Mitchell P, Marcinkiewicz, Mary Ann, & Santorini, Beatrice. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, **19**(2), 313–330.

McCallum, Andrew, & Nigam, Kamal. 1998. A comparison of event models for naive bayes text classification. *Pages 41–48 of: AAAI-98 Workshop on Learning for Text Categorization*, vol. 752.

O'Donnell, Michael. 1997. RST-Tool: An RST analysis tool. *Pages 253–256 of: INLG '00 Proceedings of the first international conference on Natural language generation*, vol. 14.

Olson, David L, & Delen, Dursun. 2008. *Advanced data mining techniques*. Springer.

Palmer, David D, & Hearst, Marti A. 1997. Adaptive multilingual sentence boundary disambiguation. *Computational Linguistics*, **23**(2), 241–267.

Porter, Martin F. 1980. An algorithm for suffix stripping. *Program: Electronic library and information systems*, **14**(3), 130–137.

Prasad, Rashmi, Miltsakaki, Eleni, Dinesh, Nikhil, Lee, Alan, Joshi, Aravind, Robaldo, Livio, & Webber, Bonnie L. 2007. The Penn Discourse Treebank 2.0 annotation manual. *Technical Report, Institute for Research in Cognitive Science, University of Pennsylvania. www.seas.upenn.edu/pdtb/PDTBAPI/pdtb-annotation-manual.pdf*.

Prasad, Rashmi, Dinesh, Nikhil, Lee, Alan, Miltsakaki, Eleni, Robaldo, Livio, Joshi, Aravind K, & Webber, Bonnie L. 2008. The Penn Discourse TreeBank 2.0. *Pages 2961–2968 of: Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC)*.

Prasad, Rashmi, McRoy, Susan, Frid, Nadya, Joshi, Aravind, & Yu, Hong. 2011. The biomedical discourse relation bank. *BMC bioinformatics*, **12**(188).

Quinlan, John Ross. 1993. *C4. 5: Programs for machine learning*. Vol. 1. Morgan Kaufmann.

Ratnaparkhi, Adwait. 1998. *Maximum entropy models for natural language ambiguity resolution*. Ph.D. thesis, University of Pennsylvania.

Rayson, Paul, & Garside, Roger. 2000. Comparing corpora using frequency profiling. *Pages 1–6 of: Proceedings of the workshop on Comparing Corpora*.

Rose, Tony, Stevenson, Mark, & Whitehead, Miles. 2002. The Reuters Corpus Volume 1 - From Yesterday's News to Tomorrow's Language Resources. *Pages 827–832 of: LREC*, vol. 2.

Rotter, Wilfried, & Bendl, Hermann. 1978. *Your companion to English texts: Comprehension, analysis, appreciation, production*. Vienna, Austria: Manz.

Schuler, Karin Kipper. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. Ph.D. thesis, University of Pennsylvania Philadelphia.

Soricut, Radu, & Marcu, Daniel. 2003. Sentence level discourse parsing using syntactic and lexical information. *Pages 149–156 of: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*. Edmonton, Canada: NAACL.

Stepanov, Evgeny A, & Riccardi, Giuseppe. 2014. Towards Cross-Domain PDTB-Style Discourse Parsing. *Pages 30–37 of: Proceedings of the 5th International Workshop on Health Text Mining and Information Analysis, A Workshop Co-Located with EACL*. Gothenburg, Sweden: Louhi.

Swales, John. 1990. *Genre analysis: English in academic and research settings*. Cambridge University Press.

Taboada, Maite. 2011. Stages in an online review genre. *Text & Talk-An Interdisciplinary Journal of Language, Discourse & Communication Studies*, **31**(2), 247–269.

Taboada, Maite, & Grieve, Jack. 2004. Analyzing appraisal automatically. *Pages 158 – 161 of: Proceedings of AAAI Spring Symposium on Exploring Attitude and Affect in Text*. Stanford University, CA: AAAI.

Taboada, Maite, Anthony, Caroline, & Voll, Kimberly. 2006. Methods for creating semantic orientation dictionaries. *Pages 427–432 of: Proceedings of the 5th International Conference on Language Resources and Evaluation*. Genova, Italy: LREC.

Vapnik, Vladimir. 1999. *The nature of statistical learning theory*. New York: Springer.

Webber, Bonnie. 2009. Genre distinctions for discourse in the Penn Treebank. *Pages 674–682 of: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2*. Suntec, Singapore: AFNLP.

Appendix A

Penn Treebank Tag Set

Appendix A provides the list of tags used in the Penn Treebank (Marcus *et al.* , 1993), and produced by the Stanford parser (De Marneffe *et al.* , 2006).

Clause level

S simple declarative clause, i.e. one that is not introduced by a (possible empty) subordinating conjunction or a wh-word and that does not exhibit subject-verb inversion.

SBAR Clause introduced by a (possibly empty) subordinating conjunction.

SBARQ Direct question introduced by a wh-word or a wh-phrase. Indirect questions and relative clauses should be bracketed as SBAR, not SBARQ.

SINV Inverted declarative sentence, i.e. one in which the subject follows the tensed verb or modal.

SQ Inverted yes/no question, or main clause of a wh-question, following the wh-phrase in SBARQ.

Phrasal Level

ADJP Adjective Phrase.

ADVP Adverb Phrase.

CONJP Conjunction Phrase.

FRAG Fragment.

INTJ Interjection. Corresponds approximately to the part-of-speech tag UH.

LST List marker. Includes surrounding punctuation.

NAC Not a Constituent; used to show the scope of certain prenominal modifiers within an NP.

NP Noun Phrase.

NX Used within certain complex NPs to mark the head of the NP. Corresponds very roughly to N-bar level but used quite differently.

PP Prepositional Phrase.

PRN Parenthetical.

PRT Particle. Category for words that should be tagged RP.

QP Quantifier Phrase (i.e. complex measure/amount phrase); used within NP.

RRC Reduced Relative Clause.

UCP Unlike Coordinated Phrase.

VP Verb Phrase.

WHADJP Wh-adjective Phrase. Adjectival phrase containing a wh-adverb, as in how hot.

WHAVP Wh-adverb Phrase. Introduces a clause with an NP gap. May be null (containing the 0 complementizer) or lexical, containing a wh-adverb such as how or why.

WHNP Wh-noun Phrase. Introduces a clause with an NP gap. May be null (containing the 0 complementizer) or lexical, containing some wh-word, e.g. who, which book, whose daughter, none of which, or how many leopards.

WHPP Wh-prepositional Phrase. Prepositional phrase containing a wh-noun phrase (such as of which or by whose authority) that either introduces a PP gap or is contained by a WHNP.

X Unknown, uncertain, or unbracketable.

Word level

CC Coordinating conjunction

CD Cardinal number

DT Determiner

EX Existential there

FW Foreign word

IN Preposition or subordinating conjunction

JJ Adjective

JJR Adjective, comparative

JJS Adjective, superlative

LS List item marker

MD Modal

NN Noun, singular or mass

NNS Noun, plural

NNP Proper noun, singular

NNPS Proper noun, plural

PDT Predeterminer

POS Possessive ending

PRP Personal pronoun

PRP$ Possessive pronoun (prolog version PRP-S)

RB Adverb

RBR Adverb, comparative

RBS Adverb, superlative

RP Particle

SYM Symbol

TO to

UH Interjection

VB Verb, base form

VBD Verb, past tense

VBG Verb, gerund or present participle

VBN Verb, past participle

VBP Verb, non-3rd person singular present

VBZ Verb, 3rd person singular present

WDT Wh-determiner

WP Wh-pronoun

WP$ Possessive wh-pronoun (prolog version WP-S)

WRB Wh-adverb

Appendix B

Full Set of RST Discourse Relations

Appendix B provides the set of 118 discourse relations from the RST framework described in Section 2.3.1. Note that certain relation names contain one or two affixes. Their meanings are:

-e embedded: the relation is embedded within another relation

-n nucleus: the EDU identified with this relation is the nucleus of the relation

-s satellite: the EDU identified with this relation is the satellite of the relation

For each of the 18 meta-relations described in Section 2.3.1, a number of lower level discourse relations are available. We provide here the meta-relations and their associated lower level relations:

Attribution attribution, attribution-e, attribution-n, attribution-negative

Background background, background-e, circumstance, circumstance-e

Cause cause, cause-result, result, result-e, consequence, consequence-n-e, consequence-n, consequence-s-e, consequence-s

Comparison comparison, comparison-e, preference, preference-e, analogy, analogy-e, proportion

Condition condition, condition-e, hypothetical, contingency, otherwise

Contrast contrast, concession, concession-e, antithesis, antithesis-e

Elaboration elaboration-additional, elaboration-additional-e, elaboration-general-specific-e, elaboration-general-specific, elaboration-part-whole, elaboration-part-whole-e, elaboration-process-step, elaboration-process-step-e, elaboration-object-attribute-e, elaboration-object-attribute, elaboration-set-member, elaboration-set-member-e, example, example-e, definition, definition-e

Enablement purpose, purpose-e, enablement, enablement-e

Evaluation evaluation, evaluation-n, evaluation-s-e, evaluation-s, interpretation-n, interpretation-s-e, interpretation-s, interpretation, conclusion, comment, comment-e, comment-topic

Explanation evidence, evidence-e, explanation-argumentative, explanation-argumentative-e, reason, reason-e

Joint list, disjunction

Manner-Means manner, manner-e, means, means-e

Topic-Comment problem-solution, problem-solution-n, problem-solution-s, question-answer, question-answer-n, question-answer-s, statement-response, statement-response-n, statement-response-s, topic-comment, comment-topic, rhetorical-question

Summary summary, summary-n, summary-s, restatement, restatement-e

Temporal temporal-before, temporal-before-e, temporal-after, temporal-after-e, temporal-same-time, temporal-same-time-e, sequence, inverted-sequence

Topic-Change topic-shift, topic-drift

Textual-organization textual-organization

Same-unit same-unit

Appendix C

Full List of Cue Phrases

Appendix C provides the complete list of cue phrases as used in the End-to-End PDTB Discourse Parser (Lin *et al.* , 2012):

I mean		as a corollary	
	all the same		at first
above all		as a result	
	also		at first blush
accordingly		as an alternative	
	alternatively		at first sight
actually		as if	
	although		at first view
additionally		as it happened	
	always assuming that		at last
admittedly		as it is	
	and		at least
after		as it turned out	
	and/or		at once
after all		as long as	
	another time		at that
after that		as luck would have it	
	anyway		at the moment when
afterward		as soon as	
	apart from that		at the outset
afterwards		as though	
	as		at the same time
again		as well	
	as a consequence		at which point
all in all		at any rate	

back	come to think of it	even if	following this
because	consequently	even so	for
before	considering that	even then	for a start
before and after	conversely	even though	for another thing
before long	correspondingly	even when	for example
before then	despite the fact that	eventually	for fear that
before...ever	despite this	ever since	for instance
besides	each time	every time	for one thing
but	earlier	everywhere	for one,
but then	either	except	for that matter
by all means	either..or	except after	for the simple reason
by and by	else	except before	for this reason
by comparison	equally	except if	fortunately
by contrast	especially because	except insofar as	from then on
by the same token	especially if	except when	further
by the time	especially when	failing that	furthermore
by the way	essentially, then	finally	given that
by then	even	first	having said that
certainly	even after	first of all	hence
clearly	even before	firstly	however

if	in so doing	initially	mainly because
if and when	in spite of that	insofar as	meantime
if ever	in sum	instantly	meanwhile
if not	in that	instead	merely
if only	in that case	it follows that	merely because
if so	in that respect	it is because	mind you
if..then	in the beginning	it is only because	more Xly
in a different vein	in the case of X	it might appear that	moreover
in actual fact	in the end	it might seem that	most Xly
in addition	in the event	just	much as
in any case	in the first place	just as	much later
in case	in the hope that	just then	much sooner
in conclusion	in the meantime	largely because	naturally
in contrast	in this way	last	neither is it the case
in doing this	in truth	lastly	neither..nor
in fact	in turn	later	nevertheless
in other respects	in which case	lest	next
in other words	inasmuch as	let us assume	next time
in particular	incidentally	likewise	no doubt
in short	indeed	luckily	no sooner than

nonetheless

on the one hand

or again

regardless of whether

nor

on the one hand..on the other hand

or else

second

not

originally

secondly

not because

on the one side

otherwise

seeing as

not only

on the other hand

overall

separately

not that

on the one hand..on the other hand

particularly because

similarly

notably

on the one side

particularly if

simply because

notwithstanding that

on the other hand

particularly when

simultaneously

notwithstanding that,

on the other side

plainly

since

now

on top of this

plus

so

now that

once

presently

so that

obviously

once again

presumably because

soon

of course

once more

previously

specifically

on balance

only

provided that

still

on condition that

only after

providing that

subsequently

on one hand

only because

put another way

such that

on one side

only before

rather

suddenly

on the assumption that

only if

reciprocally

summarising

on the contrary

only when

regardless

summing up

on the grounds that

or

regardless of that

suppose

suppose that

supposing that

sure enough

surely

that is

that is to say

that's all

that's how

that's when

that's why

the fact is that

the first time

the moment

the more often

the next time

the one time

the thing is

then

then again

thereafter

thereby

therefore

third

thirdly

this means

this time

though

thus

till

to be precise

to be sure

to begin with

to conclude

to make matters worse

to start with

to sum up

to summarise

to take an example

to the degree that

to the extent that

too

true

ultimately

undoubtedly

unfortunately

unless

until

until then

we might say

well

what is more

when

when and if

whenever

where

whereas

wherein

whereupon

wherever

whether or not

which is why

which means

which reminds me

while

whilst

with that

yet

you know