

Copyright
by
Robert Blaine Elwell
2008

**Robust Methods for Automated Discourse Connective
Argument Head Identification**

APPROVED BY

SUPERVISING COMMITTEE:

Jason Baldridge, Supervisor

Katrin Erk

**Robust Methods for Automated Discourse Connective
Argument Head Identification**

by

Robert Blaine Elwell, B.A.

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF ARTS

THE UNIVERSITY OF TEXAS AT AUSTIN

May 2008

Robust Methods for Automated Discourse Connective Argument Head Identification

Robert Blaine Elwell, M.A.
The University of Texas at Austin, 2008

Supervisor: Jason Baldridge

This thesis discusses the application of specialized models and richer features to the task of automated identification of the lexical heads of discourse connective arguments. Understanding how connectives and their arguments interact is an important task in modeling discourse and its coherency within a document. While previous work uses a single general model to make predictions for all connective arguments, building models which consider only specific connective string types or functional types shows an improvement in performance. Combining general and specific models offers even more improvement. The best model used here achieves 3.6% improvement in accuracy for selecting both arguments of a connective accurately compared to the current best model.

Table of Contents

Abstract	iv
List of Tables	vi
Chapter 1. Introduction	1
Chapter 2. Background and Data	4
2.1 Discourse and Data Structures	4
2.2 The Penn Discourse Treebank	9
Chapter 3. Models	16
3.1 Specialized Models	18
3.1.1 Connective Specific Models	20
3.1.2 Type Specific Models	21
3.2 Interpolated Models	22
Chapter 4. Features	25
Chapter 5. Experiments	32
5.1 Base model results: new features	33
5.2 Base model results: specialized models	35
5.3 Interpolated model results	37
5.4 Results by connective type	39
Chapter 6. Conclusion	41
Bibliography	45
Vita	51

List of Tables

2.1	Examples of connectives, grouped by syntactic function (from [14]).	5
4.1	Wellner & Pustejovsky’s features.	27
4.2	Additional features.	28
5.1	Accuracy scores of various models using gold parse data. . . .	36
5.2	Accuracy scores of various models on Bikel auto-parsed data. .	36
5.3	Breakdown of CONN scores by connective type using gold parse data.	38
5.4	Breakdown of CONN scores by connective type on auto-parsed data.	38

Chapter 1

Introduction

Automated analysis of discourse structure within a text document is a difficult high-level problem in the field of natural language processing. However, it has use in many valuable applications. For example, determining the full discourse structure of a text has uses in practical applications such as generation [13], text summarization [19], and automated essay evaluation [12]. This work explores a subtask of determining full discourse structure, identification of the arguments of explicit discourse connectives. Connectives are words with rhetorical functions such as *and*, *because*, and *however*; their arguments are the text spans that they relate. For example, in the sentence *John left **because** he was mad*, the first argument of *because* is *John left* and the second is *he was mad*.

Following Wellner and Pustejovsky (2007) [31] (henceforth, W&P), a machine learning approach is used to identify argument heads based on data from the Penn Discourse Treebank (PDTB, [22]), a layer of annotations for discourse connectives and their arguments over the Wall Street Journal portion of the Penn Treebank [21]. W&P use maximum entropy rankers combined with a reranking step to improve joint selection of the two arguments of each

connective. Their best model shows value for the approach, but leaves a great deal of room for improvement on identifying first arguments.

Here, I discuss improvements made on W&P's results by using models which consider specific connective string types along with specific connective functional types (*subordinating conjunctions*, *coordinating conjunctions*, and *discourse adverbials*). This approach also uses maximum entropy ranking, like the general model used by W&P. The strength of ranking on a general model is its ability to capture general behaviors between all connectives and their arguments. However, the approach is stunted in that there exist subgroups of connectives which behave differently than others. Generalizing upon these runs a risk of making false predictions on a sub-problem with regularities of its own. For instance, coordinating and subordinating connectives maintain constituency- based relationships between the connective and the first argument, but adverbial connectives can find their arguments anywhere in prior discourse. Enforcing syntactic regularity on adverbial connective argument selection would considerably reduce its performance. This is a result of the tension created between potentially informative features that may only be applicable to certain subsets of the problem. Specialized models that capture the distribution for a given connective or type of connective do not have to deal with such possibly conflicting or misleading information from the events for other connectives. These models do indeed improve performance, as seen here, but have the weakness of a reduction of applicable training data. The general model stays strong in this regard, so its interpolation with these models also

improves performance by taking advantage of the strengths of both strategies.

In all, there is a need to train more specific models for the task of argument head identification. Using such an approach improves performance, specifically for those groups of connectives which whose behaviors are not adequately captured in a general model. A general learner does, however, contribute to a global view of discourse argument head identification, and therefore must also be brought into consideration. Using simple interpolation is an appropriate method of doing so, as will be shown in this thesis.

New features also provide some improvement in performance over those used in W&P. These additional features encode (1) morphological properties of connectives and their arguments, (2) additional syntactic configurations, and (3) the wider context of preceding and following connectives. The latter help create more coherent assignments by indirectly adding greater sensitivity to other decisions.

Chapter 2

Background and Data

In this chapter, I give background on data structures for representing discourse and describe the resource used for the present task, the Penn Discourse Treebank [22]. This corpus captures structures subsumed by explicit connectives and their arguments without entering into the more complicated issue of how to group these connectives. This makes it a useful companion to any particular flavor of compositional discourse representation, amenable to multiple target representations of full discourse.

2.1 Discourse and Data Structures

The study of how discourse connectives relate units of discourse in an explicit manner has had a long-standing tradition in empirical investigation. Many corpus studies have been conducted to learn how to represent as much natural data as possible. These studies offer valuable insights into how an empirical machine learning approach should be conducted in the field of discourse. Much of the work in developing theories of discourse centers on the target representation of discourse structure. The computational ramifications of a specific theory of discourse can be directly drawn from these data struc-

Coordinating	Subordinating	Other (e.g., adverbial, prepositional)
<i>and</i>	<i>because</i>	<i>afterwards</i>
<i>or</i>	<i>when</i>	<i>previously</i>
<i>but</i>	<i>since</i>	<i>nonetheless</i>
<i>yet</i>	<i>even though</i>	<i>actually</i>
<i>then</i>	<i>except when</i>	<i>again</i>

Table 2.1: Examples of connectives, grouped by syntactic function (from [14]).

tures.

Knott (1996) provides an extensive examination of discourse connectives and their properties [14]. An important dimension on which they vary is their syntactic type: *subordinating conjunctions*, *coordinating conjunctions*, *discourse adverbials*, *prepositional phrases*, and *phrases taking sentence complements* [24]. Examples of connectives grouped by syntactic types are given in Table 2.1. From this research, it became apparent that discourse connectives maintained different argument selection tactics largely based on their syntactic function.

Webber et al [30] discuss semantic coherence and how it constrains discourse with respect to the data structures it subsumes. Here, directed acyclic graphs (DAGs) are shown to be useful for relating elementary discourse units (EDUs) for structural connectives (such as Knott’s coordinating or subordinating connectives). Incoherence can be directly related to crossing dependencies within this context; for instance, a structural connective cannot find an argument to the left of the argument of a previous connective. However, this is not

the case for discourse adverbials. These are shown to be anaphoric in nature; their first arguments can be found anywhere in the prior discourse—regardless of crossing dependencies. The DAGs described by Webber et al are amenable to Discourse Lexicalized Tree-Adjoining Grammar, which they use to formally treat the problem. Their analysis maximizes informativity from both structural and non-structural connectives by allowing the more anaphoric discourse adverbials not to adhere to those restrictions enforced for subordinating and coordinating connectives by describing them in terms of DAGs. However, by explaining one group of connectives in terms of anaphoricity and the rest in terms of a data structure, generalizability over all connectives and how they relate in the discourse is lost.

Work by Marcu [20] uses data from a corpus study to create a discourse parser based in Rhetorical Structure Theory (RST) using shallow textual cues to detect discourse connectives. However, the theoretical device of RST differs significantly from a DAG in that it is a hierarchical tree. This structure cannot account for anaphoricity of adverbial connectives and assumes that all discourse can be found somewhere within a hierarchical structure. Furthermore, tree branches are labeled with discourse functions which are often not emergent properties of the specific discourse connective.

Soricut & Marcu [27] improve upon this approach by moving from a decision tree to a probabilistic parser, improving accuracy. The implementation still performs considerably better on producing unlabelled structures than labeled structures. This suggests that the state of the art using tree-based

data structures performs best only at learning the human-annotated *structure* of discourse, but does not perform well in coherently extracting discourse level meaning between sentences. Because it is important for the structural information to be related to some explicit meaning, such an implementation would most likely have limited value for meaning extraction, which requires an understanding of *how* units of discourse are related, and not simply the structural manner in which they relate. Furthermore, this implementation only addresses sentence-internal relations, which is a heavy restriction for discourse adverbials.

Wolf & Gibson [32] conduct a corpus study that groups discourse connectives with their role in relating elementary discourse units within a text. They also use an alternate form of representing discourse. These are chain-based graphs which allow crossing dependencies. They argue against tree-based representation of discourse such as in SDRT as incapable of capturing a vast majority of discourse relations. They also argue against Webber et al's conceptualization of structural versus non-structural connectives and the data structures they require; by allowing crossing dependencies, this approach allows selection of structural and non-structural adverbials but seems to over-generalize and produce redundant links between discourse elements.

It can already be seen here that without a predictive and established theoretical data structure, using the abstracted discourse structures for machine learning tasks may result in good performance within an annotation environment, but may not be an informative model for other tasks. Marcu [18]

has argued that the usefulness of an approach in this domain can be shown in its performance increase in related tasks, such as document summarization. While information from parsing offers improvement in this field, there has been no effort to contrast and compare performance using parsers from different theories of discourse to determine the effectiveness of a particular theory over another, or whether simply introducing another layer of structural information about the document offers improvement in performance for the task.

Discourse connectives have also been treated in the area of Segmented Discourse Representation Theory (SDRT). Baldrige & Lascarides (B&L) [1] show that representing full discourse structure encounters significant and varied difficulties in strict application. Using SDRT by itself can result in some information redundancy as well as crossing dependencies for relations other than adverbials. They try to constrain this behavior by approximating SDRT graphs into trees which are closer to the DAG representation discussed above. This is applied to discourse annotation of the Verbmobil section of the Redwoods Treebank, but encounters fairly low inter-annotator agreement. However, this data source serves as an interesting proving grounds for later attempts at discourse parsing for SDRT. B&L [3] later outline a probabilistic head-driven parsing method which is modeled after the Collins parser [7]. This achieves similar labelled and unlabelled performance to Marcu's RST-style parser while also being more amenable to anaphoricity and dialogue-based discourse, which are strengths of SDRT.

SDRT is further explored by Baldridge et al (2007) [2], which describes shortcomings found in the Baldridge and Lascarides (B&L) parser. The main weakness of this implementation is its inability to use a wide variety of features, as it relies solely on distributional information, like the Collins parser. This has a dramatic impact when trained on small data sets, such as those used in these studies. This is a non-trivial problem in the area of discourse for all theories. Annotating full discourse structure is very time-intensive, as noted by Marcu (2000), even with a large amount of automation in the process. For SDRT, Baldridge et al annotate corpora with named entity information from the Message Understanding Conference and the Automated Content Extraction program. These are somewhat small corpora given that a full discourse parse instance requires an entire text. This becomes problematic with respect to statistical learning techniques; while they are excellent at handling sparsity, there is a degree to which the training data must adequately represent what needs to be modeled.

2.2 The Penn Discourse Treebank

A variety of corpora annotated for discourse structure has been created and used for the construction of discourse parsing. Each corpus generally follows its own theoretical assumptions regarding the structure of text and how it relates to the abstractions which may construct discourse. Such corpora have included the Rhetorical Structure Theory Discourse Treebank [5], the Discourse Graphbank [33], and Segmented Discourse Representation Theory

overlays to the Wall Street Journal portion of Penn Treebank as well as the Verbmobil Corpus [2].

These resources assume an underlying theory of abstract discourse relations, such as *Elaboration*, *Explanation*, and *Narration*. They also assume that discourse structure is encompassed by some specific sort of directed graph (e.g. trees [5] versus acyclic graphs more generally [2] versus fully general graphs [33]). The goal of discourse parsers for these resources is to identify how different utterances, or elementary discourse units, are connected to one another via these relations.

Despite these efforts, there is still no general agreement as to what constitutes an adequate representation of discourse structure, especially for annotating texts. The issues include determining a precise set of discourse relations, questions about whether trees are adequate, and whether it is necessary to allow for underspecified representations of discourse structure. Furthermore, as full discourse parsing requires large amounts of data, the degree of representational variation between the multiple corpora is frustrating; as many of the corpora treat the same portion of the Penn Treebank, there is much reinventing of the wheel and little mutual informativity.

The Penn Discourse Treebank (PDTB) is a valuable resource for computing aspects of discourse without the need to consider these many issues. The PDTB is specifically designed with direct correlation between surface text and discourse information in mind. By only considering direct links between discourse and text, there are no theoretical structures beyond those overtly rep-

resented by explicit discourse connectives and the spans of text which comprise their first and second arguments. The argument structure of each connective is strictly binary and non-recursive. Representations of discourse available in SDRT, RST, and various graph-based theories all subsume this more simplistic structure. This produces a shallow representation that can be annotated reliably, in large part because it remains tightly connected to the text itself and does not posit higher level structure to which we do not have such direct access. It also makes it straightforward to evaluate performance on discourse connective argument identification.

The PDTB is an excellent resource for examining the argument selection criteria of specific connectives. Using the data from Knott, the exact behavior of specific connective types can be seen. To see how different connectives behave differently, it is useful to consider some actual examples and their annotations from the PDTB. These examples follow the labeling convention used in W&P[31], which adds head identification (based on modifications to the Collins head-finding algorithm originally implemented in [16]) into the conventions used in [29]: the connective is in a box, ARG1 in *italics*, ARG2 in **bold**, and head words of each argument are underlined. The following examples show coordinating (1), subordinating (2), and discourse adverbial (3) connectives and their arguments as annotated in the PDTB:

- (1) *Choose 203 business executives, including, perhaps, someone from your own staff,* and **put them out on the streets**, to be deprived for one month of their homes, families, and income.

- (2) *Drug makers shouldn't be able to duck liability* because **people couldn't identify precisely which identical drug was used.**
- (3) *But* while International Business Machines Corp. and Compaq Computer Corp. say the bugs will delay products, *most big computer makers said the flaws don't affect them.* “Bugs like this are just a normal part of product development,” said Richard Archuleta, director of Hewlett-Packard Co.'s advanced systems development. Hewlett announced last week that it planned to ship a computer based on the 486 chip early next year. “These bugs don't affect our schedule at all,” he said.” Likewise, AST Research Inc. and Sun Microsystems Inc. said the bugs won't delay their development of 486-based machines. “We haven't modified our schedules in any way,” said a Sun spokesman. To switch to another vendor's chips, “would definitely not be an option,” he said. Nonetheless, **concern about the chip may have been responsible for a decline of 87.5 cents in Intel's stock to \$32 a share yesterday in over-the-counter trading, on volume of 3,609,800 shares, and partly responsible for a drop in Compaq's stock in New York Stock Exchange composite trading on Wednesday.**

Subordinating and coordinating connectives are typically connected to both of their arguments syntactically in the same sentence or their argument is in the immediately preceding sentence: they are structural [30]. Adverbials, on the other hand, can be anaphorically linked to their ARG1, as is clear in (3). It

is exactly this kind of difference that suggests using different models for these very different types of connectives should be a helpful strategy.

Discourse connectives and their arguments were selected by annotators on the raw Wall Street Journal portion of the Penn Treebank. At a later stage, the argument text spans were overlaid with gold-standard trees in the merged section of the corpus, and Gorn addresses were provided specifying the forest of nodes which comprised the full span. The spans have no necessary overlap with even basic syntactic constituency because the annotators worked from the text, not from gold-standard syntactic trees. These node sets often include one terminal member of a phrase without its sisters, or they group together elements which do not share an immediate governing node. Out of the 15,440 discourse connectives in sections 02-22 of the PDTB, 14,477 (93.8%) of ARG1 spans are not syntactically aligned. 12,709 (82.3%) of ARG2 spans are also not syntactically aligned. This means that most discourse connective argument spans do not share a governing syntactic node. These statistics are consistent with observations on syntactic alignment with discourse structure that motivated the choice to work closely from the text: the corpus designers did not want to bias choices about discourse units based on earlier choices about syntactic structure [11]. An example of a sentence that portrays this characteristic of the PDTB is the following:

- (4) Under two new features, participants will be able to transfer money from the new funds to other investment funds *or*, if **their jobs are terminated**, *receive cash from the funds*.

The notable characteristic about the connective argument structure in (4) is the position of *or* and its constituency. In most treatments of syntax, let alone discourse, creating a coherent unit from all words in ARG1 of (4) would create crossing dependencies, which is generally problematic.

Annotators dealt with each connective independently. As a result, two connectives can have interleaved, overlapping arguments. For example, the connectives *because* and *then* occur next to one another in the following passage; the ARG2 of *because* subsumes that of *then* while the ARG1 of *then* is before that of *because*:

- (5) John loves Barolo. He ordered three cases of the '97. But *he had to cancel the order* because **then he discovered he was broke.**
- (6) John loves Barolo. *He ordered three cases of the '97.* But he had to cancel the order because then **he discovered he was broke.**

Overlap therefore becomes an issue which may be valuable to be considered when building a model. Including features that state what the previous and following connectives and whether or not there is overlap in the candidate being considered for those and the current connective should help in appropriately selecting the correct candidate. A discriminative model could utilize this feature to penalize candidate heads which are more local to or syntactically dominated by overlapping discourse connectives, to which they would more likely serve as ARG2 heads.

It is important to select a word with some syntactic motivation to represent an argument span, but due to the lack of consistent alignment between syntax and discourse, it is necessary to enforce a syntactic relationship between the various nodes in the span forest. Following W&P, this is done by finding the least common ancestor (LCA) node which governs all terminal nodes within the argument span. Performance is evaluated on whether the model successfully selects the terminal head of the LCA node. Selecting representative lexical heads rather than full spans avoids extra processing steps in further applications where providing a span would, in many cases, require extraction of lexical head information on the outset. Furthermore, given the alignment discrepancy between syntax and discourse, identifying full spans is a non-trivial task with unclear and inconsistent syntactic counterparts.

Chapter 3

Models

There are two stages in identifying the arguments of a discourse connective: the heads of candidate arguments must be identified and then the best candidate must be chosen. For the first stage, W&P select candidates only within a distance of ten *steps* of the connective. A step is defined as the traversal of a sentence boundary or dependency link. The heuristic may traverse sentence boundaries for ARG1s, but stays within the same sentence as the connective for ARG2s, which are generally syntactically dependent upon the connective. This candidate selection process is in many ways similar to the pruning heuristic used in semantic role labeling [34]. This limits the number of candidates which must be considered during classification in a principled manner.

Selecting the proper base algorithm is an important step in any natural language learning task. Because of the general issues associated with all machine learning in NLP, considering a statistical model is an obvious first step. Standard classification is one solution to consider, but for this specific task, the number of elements to be considered makes this prohibitive. Because argument heads correspond to discourse connectives, the boolean decision of

whether a head is an argument or not is dependent upon the connective in question. Naive classification could result in multiple heads for the same argument or no heads selected for certain connectives. It is therefore important to consider other statistical approaches which may retain the advantages of traditional classification models for natural language learning.

To identify the best candidates, W&P use a maximum entropy ranker with a large feature set which considers syntax, dependency, and lexical semantics. Maximum entropy models are widely used for classification tasks in natural language processing—they are accurate, can incorporate non-independent, overlapping features, and are reasonably fast to train. The advantage to using rankers (as opposed to classifiers) for this particular task is that the model assumes a single correct answer among a set of possible candidates at a given instance. By correlating each instance with a connective, all candidates for that connective are considered at once, and the best according to the model is selected. This has been shown to improve accuracy for tasks with a similar structure, such as question answering [25] and pronoun resolution [10]. In the composition of the PDTB, there cannot be more than one first argument and second argument for a single connective. Ranking is thus a good fit for this task. It further concretizes the parallels between coreference and argument selection. Both tasks find ranking to be a useful way to articulate a problem dependent on more than one variable, as it determines the likelihood of a candidate dependent upon another given but variable item—in this case, the connective, but in the case of coreference, a specific pronoun.

Using a maximum entropy ranker as well, models are trained for ARG1 and ARG2 selection separately. The model for ranking with respect to the identity of a candidate head α_i as an argument head $\hat{\alpha}$ given a connective π and a document x is as follows:

$$P_{\hat{\alpha}}(\alpha_i|\pi, x) = \frac{\exp(\sum_k \lambda_k f_k(\alpha_i, \pi, x))}{\sum_{\alpha_j \in C_{\hat{\alpha}}(\pi, x)} \exp(\sum_k \lambda_k(\alpha_j, \pi, x))} \quad (3.1)$$

where the f_k are feature functions, the λ_k are their respective weights, and $C_{\text{ARG}}(\pi, x)$ is the set of candidate arguments per connective π within a document x . This approach will be identified as the general connective model, or GC. The number of training instances coincides with the number of connectives in the training set.

The implementation for maximum entropy ranking used here is that found in the Toolkit for Advanced Discriminative Modeling¹ [17] to determine weights for this model and the specialized models described in the next two sections. For all models, an empirically determined gaussian prior of 100 is used based on performance on the development set.

3.1 Specialized Models

Building a generalized model for selecting the heads of all connectives can be a particularly difficult problem. Because of the variety of functions a connective can serve in providing a coherent discourse, learning a model over

¹<http://tadm.sf.net>

all connectives may lose valuable information specific to the set of inferences particular to a connective, which may serve to constrain selection of a head based on its lexical semantics. Furthermore, connective head selection behavior is syntactically predictable for two varieties of connectives. By creating a general predictive model over all syntactic types, this useful pattern becomes washed over; it weakens anaphoric head selection for adverbial connectives and confuses selection for more structural connectives such as coordinators and subordinators.

The theoretical motivation for building specialized models comes from two separate backgrounds. In the area of discourse, Marcu [20] points out that specific connectives have particular argument selection behaviors which may be learnable. Because of this, it may be valuable to investigate models specific to the connective. Conversely, Webber et al [30] show that certain functional types of connectives have radically different argument selection patterns. As anaphoric connectives and structural connectives may compete in a general model of argument selection, suggesting that building models specific to the functional types of connectives may offer some error reduction.

By building specialized models which only consider a specific subgroup of connectives, it is hoped that the features which truly discriminate the correct head from all other candidates will gain stronger weighting, as the amount of conflicting information is removed from the general problem. This is similar to approaches outlined for coreference by Denis [8], which used specialized ranking models for different kinds of coreference learning. These models were

trained and tested using the same set of features and ranking model for antecedent selection, but associated a ranking instance with one of five models which were divided by the form of the referent. Despite a reduction in training instances for each model, groupings which had more reliable patterns for coreference prediction performed particularly well, whereas in a single model, they suffered from less clear-cut learning of these patterns among other types of referents.

3.1.1 Connective Specific Models

While discourse connectives have similar functions—to join two units of discourse in some meaningful way to show how the two utterances relate—the set of inferences associated with each is starkly different. For instance, *however* could not be used interchangeably with *furthermore* without the loss of the semantic presuppositions on how ARG1 and ARG2 relate in these specific situations.

Discourse connectives, how they relate to arguments, and how those arguments relate to another are subject to a large, conflated set of different sources of information, such as lexical semantics, pragmatics, temporal semantics, and world knowledge. By assuming that some of these various sources of information are constrained by the discourse connective in consideration (or that the connective selection is constrained by certain given facts among these sources of information), it may be possible to better predict proper argument heads.

Enforcing this assumption is done by training models specific to the connective in question. This should offer a valuable alternative to building a generalized model. It has the propensity to effectively predict the individual argument selection properties of a given connective without assuming its behavior can serve to predict that of other connectives. For example, *then* typically relates two temporally related arguments whereas *but* usually contrasts two propositions regardless of temporal considerations. A general model would most likely be unable to capture this distinction between the two.

For each individual connective, a specific model is trained for it using all instances in the training data. During testing, prediction for a connective is determined based only on the predictions of this specific model. In a few cases, a connective unseen in the training material occurs in the test material; for such cases, the GC model is used as a backoff model. This connective specific approach will be referred to as SC.

3.1.2 Type Specific Models

Connectives link arguments in three primary different ways. Coordinating connectives (i.e. *and*) assume argument spans are generally syntactically similar. Subordinating connectives (i.e. *because*) are dominated or adverbially linked to the ARG1 and dominate the ARG2. Adverbial connectives (i.e. *nonetheless*) can appear in several places in the sentence and have no necessary syntactic relationship to their ARG1. Because of this tendency, this subset of connectives more closely mirrors the task of coreference resolution.

While the ranking model used has been proven to be appropriate for the task [10], modeling all functional types of connectives together may weaken its predictive strength. Because of this, it is important to model each set independently to more closely reflect the differing behaviors of each connective group. The type of each connective is determined from the list of connectives and their properties given by Appendix A in Knott (1996) [14]. As a default, any connective that is not mentioned as subordinating or coordinating there is assumed to be adverbial. Using this approach arrives at a slightly different frequency of the three types than outlined in W&P, but stays fairly close to the distribution amongst all types. This set of (three) models is referred to as the type connective model, or TC.

3.2 Interpolated Models

As will be apparent in the results, the individual models described in the previous sections trade off precise modeling of the distribution for specific connectives or types of connectives with the extra evidence (and larger training set) available with the general GC model. Combining these single model using standard linear interpolation is a simple way of realizing the benefits of both.

Interpolation is done in two steps. The first model considered is TC interpolated with GC, which is then itself interpolated with SC. Simple interpolation as seen between TC and GC has the following form:

$$P_{TG}(a|c_i) = \lambda_{t_i}P_{t_i}(a) + (1 - \lambda_{t_i})P_g(a) \quad (3.2)$$

where a is a candidate argument, c_i is the specific connective under consideration, t_i is the type of that connective, P_{t_i} is the model for that type, P_g is the general connective model, and λ_{t_i} is a interpolation weight for connective type t_i controlling the influence of the general model.

The interpolation of SC with TC and GC takes a similar form that builds on P_{TG} :

$$P_{SGT}(a|c_i) = \lambda_{c_i}P_{c_i}(a) + (1 - \lambda_{c_i})P_{TG}(a|c_i) \quad (3.3)$$

where P_{c_i} is the specialized model for connective c_i .

The connective specific weights λ_{c_i} and connective type weights λ_{t_i} are determined in a very simple manner that ensures that connectives or types which have been seen fewer times leave more mass for P_{TG} or P_{SGT} , respectively. For example:

$$\lambda_{c_i} = \frac{freq(c_i)}{freq(c_i) + C} \quad (3.4)$$

where C is some constant chosen based on performance on held-out development data. λ_{t_i} is set similarly. For λ_{c_i} , we set C to be 99, the number of different connectives available in the corpus. For λ_{t_i} , C is set to be 3, the number of different types of connectives.

While more complex model combinations could be considered to further improve accuracy, this simple approach shows encouraging results when used

with these base models. Furthermore, it is robust to many different values of C for both connectives and connective types.

Chapter 4

Features

Discourse tasks such as connective argument identification are influenced by many aspects of the text and its underlying content. W&P use a wide-range of features, listed in Table 4.1. Examples are given for the first candidate and connective pair listed in the training data—the first document with a PDTB layer on Section 02. The candidate head is ‘institutions’; the connective is ‘Also’. The relevant text excerpt is shown below using the conventions described in Chapter 2, with the candidate head in SMALLCAPS:

- (7) For its employees to sign up for the options, a college also must approve the plan. Some 4,300 INSTITUTIONS are part of the pension fund.

The new options carry out part of an agreement that the pension fund, under pressure to relax its strict participation rules and to provide more investment options, reached with the SEC in December.

The new "social choice" fund will shun securities of companies linked to South Africa, nuclear power and in some cases, Northern Ireland.

Also excluded will be investments in companies with "significant" business stemming from weapons manufacture, alcoholic beverages or tobacco.

Features A-G encode aspects of the surface text, such as the connective string itself. Features A-L and M-Q provide a deeper level of analysis by capturing phrase structure and dependency path information, respectively. Features R-Y include some lexical semantics with respect to the behavior of the candidate and connective.

There are of course other factors that can be extracted from a document to use as features for the task. Here, an additional set is considered which attempts to account better for the prediction of one connective based on other neighboring connectives and that would be especially useful for models for specific connectives or connective types. This also includes morphological stemming and additional syntactic features not considered by W&P. In part, these are intended to provide greater robustness to the disconnect between syntactic constituents and connective arguments in the PDTB [11, 22]. This implementation also removes or modifies some of W&P’s features based on performance observed in the development material.

The W&P feature set is changed in the following way: directional information for constituent paths is omitted, making the features simply serial part of speech strings. Feature J is removed. This is replaced by the lexical path from the candidate to the connective, left blank if there is a single transition. As the dependencies given are realized as a graph rather than a hierarchical tree, dependency path features do not contain directional information. This helped particularly in improving ARG1 scores on the development data.

	Description	Example
A	Connective position in sentence (start, mid, end)	beginning
B	Whether candidate is in same S as connective	False
C	Connective phrase string	Also
D	Downcase connective phrase string	also
E	Candidate string	institutions
F	Candidate before or after connective	before
G	A & B	beginning_False
H	Constituent path from candidate to connective	NNS_NP_S_RB_NP_S
I	Constituent path length (# of nodes traveled)	5
J	Collapsed constituent path without part of speech	institutions_are_part_of_the_pension_fund...also
K	Abbreviated path, node repetitions collapsed	NNS_NP_RB_S
L	C & H	Also_NNS_NP_S_RB_NP_S
M	Dependency path from candidate to connective	nsubj_ROOT_ROOT_ROOT_advmod
N	Dep. path, head word of first link from connective	part_nsubj_ROOT_ROOT_ROOT_advmod
O	Collapsed dep. path removing coordinating links	nsubj_ROOT_ROOT_ROOT_advmod
P	Collapsed dep. path removing repetitions of links	nsubj_ROOT_advmod
Q	C & M	Also_nsubj_ROOT_ROOT_ROOT_advmod
R	Connective function (coord., sub., or adv.)	Adv
S	A & R	beginning_Adv
T	M & R	nsubj_ROOT_ROOT_ROOT_advmod_Adv
U	Candidate is an attributing verb	False
V	Candidate has a clausal complement	False
W	U & V	False_False
X	Candidate is a verbal clausal complement	False
Y	X & governing verb is an attributing verb	False_False

Table 4.1: Wellner & Pustejovsky's features.

	Description	Example
α	Previous connective string	NONE
β	Following connective string	also
γ	Connective in quotes	False
δ	Candidate in quotes	False
ϵ	γ & δ	False_False
ζ	γ & δ AND same quote	False_False_False
η	Word to the left of the candidate	4,300
θ	Word to the right of the candidate	are
ι	Word to the left of the connective	.
κ	Word to the right of the connective	excluded
λ	The unique set of node labels in the constituent path	NNS_NP_S_NP_S
μ	Lemmatized candidate	institution
ν	Inflection features of candidate (-ing/-ed, etc.)	s
ξ	Candidate a constituent of another connective	False
o	Candidate argument position in dependency graph	0
π	Connective argument position in dependency graph	0
ρ	Candidate in a copular sentence (boolean)	True

Table 4.2: Additional features.

The additional features are summarized in Table 4.2. These add more context within each decision. Because a ranking instance occurs for each connective with all candidate heads considered at once, there is more room to consider dependencies between features. This is particularly important for cases of time-sensitive connectives, where a connective such as *seven months after* would most likely not have an ARG1 in the present tense or an ARG2 in the past tense. By regularly noting that candidates with certain inflection features, (such as *-ed* when considering the aforementioned connective) are more regularly selected as heads than their non-inflected or otherwise-inflected adversaries, the model’s hypothesis regarding head selection is more finely honed. This is particular to ranking as opposed to classification, as it is dependent upon the given connective. Because of this, it can be assumed that features such as these which are specifically informed by a particular connective would serve to significantly improve prediction within more specialized models.

Features α through κ are intended to introduce a higher level of context-sensitivity using simple, surface-level cues. Depending on the surroundings of the connectives, there may be preferences for different types of arguments which a model dealing with a specific connective or type of connective might more easily represent. This is also the reasoning behind feature ρ , which is introduced under the assumption that these connectives have varied distributional properties regarding selection of copular heads. Similarly for feature λ , certain phrasal nodes may block well-formed candidate heads within the discourse—for example, a subordinating or coordinating adverbial may give

heavy weighting to any groupings which include traversal of a sentence, as they select arguments within the sentence with predictable constituency relations. Features ρ and λ were shown to improve performance on development data.

Some features were introduced to account for error analysis in previous work. W&P note that their model’s greatest weakness was in attribution; adding features to address this aims to block the clearest form of attribution. The model should learn to heavily weight against quoted text for connectives outside of quotes in most cases, for instance. Features γ through ζ deal with within-quotes discourse as opposed to document-level discourse. In some ways, these features are elaborations on feature U, as they may help to block narrative document-level discourse from being related to discourse stemming from a specific speaker whose dialogue is being directly attributed. In other ways, it prevents errors from more orthogonal patterning. Quoted text in a previous sentence most likely would not contain an argument head of a connective not in quotes. This was not previously considered, and is possible source of error for adverbial connectives.

The output of the **morpha** lemmatizer [23] is used for several features to allow generalization over lemmata as well as introduce weak tense-based features as relations between the first and second argument, as in feature ν . This is an important aspect to include as a weak indicator of temporal factors, which should help for connectives such as *fully eight months before*, *four days after*, and *ever since*.

Work on temporal semantics shows that modeling sentence-internal temporal relations and discourse structure are codependent tasks [15]. Just as discourse information has been suggested as valuable for identifying temporal relations, features which address the temporal relation between arguments may, in a shallow manner, improve performance here. An improvement in performance given these features would suggest avenues for future work which incorporates more elaborate temporal information, or perhaps determines it jointly.

As the candidate selection process is a fairly naive algorithm, feature ξ is introduced to attempt to discourage selection of head words which are immediate constituents of other discourse connectives. These are much more likely to be ARG2s of other connectives than first arguments of the connective in question.

The additions and modifications made to the benchmark feature set ultimately seek to draw in a greater degree of discourse-level context. These should also serve to improve the predictive power of connective specific models. Introducing features which consider tense and adjacent connectives should also improve sensitivity to the greater discourse, which is important in constraining a somewhat shallow selection process. Intuitively, as a connective’s argument selection behavior is a consequence of the full discourse, this should improve the performance of the model.

Chapter 5

Experiments

As is standard with experiments using the Wall Street Journal portion of the Penn Treebank, the model is trained on sections 02-22, developed on sections 00 and 01, and tested on sections 23 and 24. This allows direct comparison between this model and W&P in terms of performance. Similar to W&P, ARG1 and ARG2 identification is separately evaluated along with a metric for selecting both arguments correctly for a given connective (referred to as CONN accuracy). Performance is also broken down in terms of connective types (These types were determined from Knott (1996) [14]. See Chapter 3).

In order to evaluate the performance of the model on non-gold-standard syntactic parses, outputs are also evaluated using automatically parsed Penn Treebank text. While W&P use auto-parses from the Johnson-Charniak parser [6], the model used here is the Bikel Statistical Parser¹ [4]. The parsed training data here is derived from a 5-fold model where training and development receives the benefit of the full gold-standard training set.

Results for ARG1, ARG2, and CONN accuracy for the various models are given in 5.2. Scores are given for W&P's best simple ranking model (W&P-

¹<http://www.cis.upenn.edu/~dbikel/software.html>

BASE) and their reranker (W&P-RERANK). The models given here are also simple ranking models, like W&P-BASE,² and would all likely improve with reranking. As seen in W&P, it should be expected that ARG2 selection is considerably easier than ARG1 selection, as its behavior is less varied with respect to constituency. This is therefore the focus of these experiments; ARG1 selection in W&P was consistently the weaker of the two. As distance grows between the candidate and connective, dependency and constituency path features between the connective and the candidate became sparser and less useful. Because of these factors, it was decided early on to focus feature addition and model selection almost solely on improving ARG1 accuracy.

5.1 Base model results: new features

Both of the W&P scores use all of their features. Changing some of the features and removing others, as described in Chapter 4, arrives at a model, GC-W&P-REVISED, that has better ARG1 accuracy (78.1 vs. 75.0), worse ARG2 accuracy (91.8 vs. 94.2), but is overall more accurate at getting both connectives right (73.0 vs. 71.7). When additional features are added (Table 4.2), performance improves on all measures (indicated by the row for GC-ALL), and in particular the CONN accuracy nearly rivals that of W&P-RERANK (73.9 vs. 74.2). These results show the utility of the additional features for ARG1 accuracy, but also demonstrate that W&P’s original set is better for ARG2 identification. This suggests that the two tasks should be

²This model was replicated, obtaining the scores reported by W&P.

tackled with feature sets tuned to each, rather than a single feature set as both we and W&P have done here.

It is interesting to note the improvement on performance compared to the auto-parse sub-task pursued by W&P, as seen on Table 5.2. The models implemented here do not suffer as severely using autoparse data. The introduction of extra features which reduce reliance on syntactic information more than likely contributes to this. Bringing in more local features such as word adjacency and more discourse-level features such as the preceding and following connective should play a role in mitigating the expectable reduction in accuracy compared to the gold standard.

The differences between this implementation and W&P with respect to performance on auto-parse data can be attributed to a number of conflated issues. First, the implementation of a separate statistical parser may contribute to the reduction in performance for the W&P models—here, Bikel parses are used, but W&P use parses from the Charniak-Johnson parser. The original features may not have been well-suited for the outputs of the Charniak-Johnson parser; the Bikel implementation could be performing particularly well in providing syntactic paths which are especially felicitous with connective argument selections. That the implementation in GC-W&P-REVISED does not suffer nearly as much as those found in the Charniak parses also suggests this, but could also be attributed to the modifications of some features in this model.

5.2 Base model results: specialized models

Next, the results for the connective-type model, TC-ALL, shows that further gains are made by using a model which treats each type of connective separately. In particular, this greatly improves ARG1 accuracy (by 3%, from 78.7 to 81.7). This is as expected, since it is with respect to ARG1 that the different connective types behave most differently. As discussed in Chapter 2, subordinating and coordinating connectives usually find their ARG1’s structurally, whereas adverbial connectives find them anaphorically (and usually at a greater distance).

The connective specific model, SC-ALL, does not do as well as TC-ALL on ARG1, but it is still better than the general GC-ALL model and it is our best simple model on ARG2. This suggests that TC-ALL is still too course-grained: ARG2’s are found in the same sentence, and both GC-ALL and TC-ALL must have some connectives which are overpowering the preferences of others. On the other hand, because ARG1’s are further away for adverbial connectives, TC-ALL may be better able to better use the evidence from all adverbial connectives while not suffering from the great reduction in training events that SC-ALL necessarily incurs (consider, for example, that many connectives—especially the adverbial ones—are found only once in the training material). In particular, this means that the general problem of sparser features for ARG1’s is greatly aggravated in the SC-ALL model.

The most notable detail of the performance of specialized models on the auto-parse data is the decrease in performance for SC-ALL. Referring

Model	ARG1	ARG2	CONN
W&P-BASE	75.0	94.2	71.7
W&P-RERANK	76.4	95.4	74.2
GC-W&P-REVISED	78.1	91.8	73.0
GC-ALL	78.7	92.1	73.9
TC-ALL	81.7	92.6	76.1
SC-ALL	80.3	93.1	75.8
TC-GC-INTERP	81.7	93.2	77.2
SC-TC-GC-INTERP	82.0	93.7	77.8

Table 5.1: Accuracy scores of various models using gold parse data.

Model	ARG1	ARG2	CONN
W&P-BASE	67.9	90.6	62.7
W&P-RERANK	69.8	90.8	64.6
GC-W&P-REVISED	76.9	89.0	70.2
GC-ALL	77.1	89.1	70.2
TC-ALL	78.9	89.7	72.4
SC-ALL	78.7	85.4	68.4
TC-GC-INTERP	79.8	89.9	73.1
SC-TC-GC-INTERP	80.0	90.2	73.6

Table 5.2: Accuracy scores of various models on Bikel auto-parsed data.

to Table 5.4, it seems that the weakest performance for this model is among subordinating connectives. This is most likely a result of a less reliable syntactic representation exploiting this model’s main weakness—its lack of training data. Without a consistent syntactic pattern, it seems that subordinating connectives become even more dependent on larger amounts of training data to converge upon a strong hypothesis.

5.3 Interpolated model results

Clearly, there is value in using more specific models, but they must be balanced by more general models to protect against sparsity. The simple linear interpolation described in Chapter 3.2 is a straightforward way to combine these models. As indicated by the row for TC-GC-INTERP, Interpolating TC-ALL with GC-ALL does help. In particular, ARG2 accuracy improves from 92.6 for TC-ALL on its own to 93.2. Because ARG2’s are more similar across the different types of connectives, the interpolated model is able to incorporate additional—and more importantly, relevant, appropriate, and more numerous—evidence from GC-ALL.

Finally, interpolating the three levels together, SC-TC-GC-INTERP, provides the best results. This combined model can use SC-ALL when it has a specific connective that had many training instances and thus can be modeled well by the single specialized model while relying more on TC-GC-INTERP for connectives which were observed just a few times in the training material. (These contributions are governed by the connective specific λ_{c_i}

Model	Subord.	Coord.	Adverb.
GC-W&P-REVISED	80.8	74.2	58.7
GC-ALL	80.9	75.5	59.8
TC-ALL	82.8	77.5	67.5
SC-ALL	83.9	78.0	59.0
TC-GC-INTERP	82.8	77.5	67.8
SC-TC-GC-INTERP	83.9	78.1	67.5

Table 5.3: Breakdown of CONN scores by connective type using gold parse data.

Model	Subord.	Coord.	Adverb.
GC-W&P-REVISED	76.2	73.3	55.4
GC-ALL	76.9	73.5	54.0
TC-ALL	78.2	75.4	58.2
SC-ALL	67.9	77.4	53.2
TC-GC-INTERP	77.3	76.5	60.7
SC-TC-GC-INTERP	78.0	77.1	60.7

Table 5.4: Breakdown of CONN scores by connective type on auto-parsed data.

weights, as defined in Chapter 3.2). This model achieves the best performance in all three metrics and provides a 3.6% relative improvement over W&P’s reranking model. One would expect to get even better results by reranking the output of SC-TC-GC-INTERP.

The interpolated models seemed to suffer considerably in terms of ARG2 selection. This is, in some way, to be expected; ARG2’s are always subordinating, and if the parser does not predict the correct constituency relationship between a certain connective and its argument, selecting its head is rendered more difficult. As errors cascade through a parse, the likelihood of

recovering a proper candidate head through Collins-style selection rules also decreases. These still result in the best performance using auto-parse syntactic data rather than the gold standard.

5.4 Results by connective type

It is instructive to consider the results for each of the models on CONN accuracy for each different type of connective. This is given in Table 5.3, with auto-parsed results in 5.4. The most salient number is 67.5 for adverbial accuracy for TC-ALL. Clearly, this single model represents the most important split in specialization since it allows the structural dependencies of subordinating and coordinating connectives to be modeled differently from the anaphoric dependencies of adverbials. It thus captures the longer distance adverbial ARG1's more accurately than GC-ALL. Although SC-ALL also specializes (even further than TC-ALL), it is hurt by subsequent sparsity since many of the models that constitute it are trained on just a few examples. TC-ALL thus provides a good balance between both extremes of GC-ALL and SC-ALL.

SC-ALL, on the other hand, does best of all single models for subordinating and coordinating connectives. As mentioned above, sparsity is less of an issue for these connectives since their arguments are usually found much closer and in stricter syntactic relationships to them.

Of course, the interpolated models straightforwardly incorporate the benefits of all these strategies and perform better across the three connective types than any of the single models.

Auto-parse data becomes especially problematic for certain connective types. As previously mentioned, SC-ALL, which performed best of all on subordinating connectives, suffered the most in performance reduction when the gold-standard parse was not available. Obviously, variability in what was once a very predictable relationship conflated with a reduction in training data is the major source of this issue. TC-ALL has the best performance on subordinating connectives for this sub-task. This model has the benefit of learning a similar pattern over a larger amount of training instances, which mitigates error attributable to noise created by any weaknesses in the auto-parse.

TC-ALL still has the best performance on coordinating connectives using auto-parse data. I speculate that this is due to the difficulty of accounting for coordination in tree-based syntax in general. Even if there is a consistent pattern in the automatic parse, the connective would not dominate the ARG1. The model should be able to learn that such connectives must be in the same sentence as the connective, and must precede it. In this way, coordinating connectives become in some sense a more anaphoric problem with a scope that is much more restricted compared to adverbial connectives. Due to its more limited scope, SC-ALL is also poised to take advantage of lexical affinity in a way which TC-ALL may not. This could be its key to success here.

Chapter 6

Conclusion

This work has shown that accuracy in identifying the arguments of discourse connectives can be improved by building models for specific connectives and/or different connective types. These models allow the specific distributions for a connective or connective type to be modeled more closely, but suffer from not having as much training material as a general model that uses all connectives. It additionally shows that the strengths of both the specialized models and the general model can be realized by combining them with simple interpolation.

It is also shown here that additional features provide some improvement in performance. These features use morphological analysis, further syntactic patterns, and information about the distribution of other connectives in relation to the connective under consideration. Using these new features and interpolating a connective specific model, connective type model, and general model, achieves 77.8% accuracy for identifying *both* arguments of connective, a 3.6% absolute accuracy improvement over the state-of-the-art result of W&P [31].

An immediate way to improve these results is to use separately tuned

feature sets for ARG1 and ARG2 identification, as can be seen in the difference between W&P’s basic model and the revised version of their feature set (Table 5.2): the former does better on ARG2, whereas the latter is better on ARG1. Our subsequent models—using specialization—improve ARG1 accuracy using the same features as the latter, and they nearly rival that of W&P’s basic model for ARG2. Thus, building on W&P’s features for ARG2 identification would likely improve the results of the specialized models outlined here. And of course, the best models would be expected to additionally benefit from W&P’s reranking approach.

Some of the additional features here allow the wider discourse context to be considered indirectly. However, each decision is still made independently of the others, so decisions about one connective do not influence those for others—even though they are clearly relevant. One way of modeling these decisions more globally would be to use integer linear programming [26]. In the related task of determining coreference and anaphoricity, Denis and Baldridge (2007) constrain the outputs of specific models to combine their evidence globally and ensure that individual assignments are coherent with respect to one another [9]. For example, for this task, one could use a constraint that says that two connectives should not choose exactly the same argument span.

It would also be interesting to consider the output of models built for discourse connective argument identification as features or constraints in a full discourse parser, such as those described by Marcu, [19], Baldridge & Lascarides (2005) [3], and Baldridge & Lascarides (2007) [2]. Due to the

noncommittal nature of the Penn Discourse Treebank with respect to different theories of discourse relations and discourse structure, the output of our implementation could be applied to any of these approaches. Using this information to improve discourse parsing could be useful in improving the performance of other tasks, such as document summarization, as shown by Marcu (2000). To fully understand how each parser performs and best improves accuracy, an informative experimental setup would include parsers based in multiple theories of discourse representation. Results would be compared for both unconstrained parses as well as parses constrained by the implementation described here.

A further application would be to use our output to rebuild a full discourse argument span along with a discourse chunker as features for candidate spans. As some EDU chunkers take span delineation into account [28], this may also be an interesting avenue to utilize joint modeling.

Higgins et al [12] use a support vector machine to bootstrap an online essay evaluation service. They use a RST-style discourse trees in order to compare sentences within the essay to their rhetorical function within the text to evaluate coherence. Constraining the RST parses (or a parser which is more amenable to crossing dependencies) with information from this implementation may be useful in improving performance. It could also lend an important aspect of its own to essay evaluation. The implementation here is able to model the specific argument selection behavior of a connective; by comparing given argument heads of a specific connective to the connective itself, one may be able to evaluate whether the use of specific connectives results in coherent

argumentation within the text.

Hovy (1993) [13] argues for the use of RST-style discourse structures to constrain language generation output in AI. However, such an implementation may be incapable of understanding anaphoric relations to the prior discourse and would require the development of a large discourse taxonomy to learn relations. The model provided here could easily feed in large amounts of unannotated text to the language generation implementation, both capturing structural and anaphoric explicit discourse relations. At the same time, it may in some way offer an understanding of how relations work via statistical inference. This may be a useful parallel to or replacement for a more structured discourse representation.

Bibliography

- [1] J. Baldridge and A. Lascarides. Annotating discourse structures for robust semantic interpretation. In *Proceedings of the 6th International Workshop on Computational Semantics*, Tilburg, The Netherlands, 2005.
- [2] Jason Baldridge, Nicholas Asher, and Julie Hunter. Annotation for and robust parsing of discourse structure on unrestricted texts. *Zeitschrift fur Sprachwissenschaft*, 26(213–239), 2007.
- [3] Jason Baldridge and Alex Lascarides. Probabilistic head-driven parsing for discourse structure. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 96–103, Ann Arbor, MI, 2005.
- [4] Daniel M. Bikel. Design of a multi-lingual, parallel-processing statistical parsing engine. In *Proceedings of the second international conference on Human Language Technology Research*, pages 178–182, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.
- [5] Lynn Carlson, Daniel Marcu, and Mary-Ellen Okurowski. Building a discourse tagged corpus in the framework of rhetorical structure theory. 2001.

- [6] Eugene Charniak and Mark Johnson. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 173–180, Morristown, NJ, USA, 2005. Association for Computational Linguistics.
- [7] Michael Collins. Three generative, lexicalised models for statistical parsing. In *Proc. of the 35th Annual Meeting of the ACL*, pages 16–23, Madrid, Spain, 1997.
- [8] Pascal Denis. *New Learning Models for Robust Reference Resolution*. PhD thesis, University of Texas at Austin, Austin, Texas, USA, 2007.
- [9] Pascal Denis and Jason Baldridge. Joint determination of anaphoricity and coreference resolution using integer programming. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 236–243, Rochester, New York, April 2007. Association for Computational Linguistics.
- [10] Pascal Denis and Jason Baldridge. A ranking approach to pronoun resolution. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1588–1593, Hyderabad, India, 2007.
- [11] Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Rashmi Prasad, Aravind Joshi, and Bonnie Webber. Attribution and the (non-)alignment of syntactic

- and discourse arguments of connectives. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 29–36, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.
- [12] Derrick Higgins, Jill Burstein, Daniel Marcu, and Claudia Gentile. Evaluating multiple aspects of coherence in student essays. In *Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference*, Boston, MA, 2004.
- [13] Edward Hovy. Automated discourse generation using discourse structure relations. *Artificial Intelligence*, 63(1-2):341–386, 1993.
- [14] A. Knott. *A Data-Driven Methodology for Motivating a Set of Coherence Relations*. PhD thesis, University of Edinburgh, 1996.
- [15] M. Lapata and A. Lascarides. Learning sentence-internal temporal relations. *Journal of Artificial Intelligence Research*, 27:85–117, 2006.
- [16] C. Manning M. Marneffe, B. Maccartney. Generating typed dependency parses from phrase structure parses. In *Proceedings of the 5th edition of the conference on language resources and evaluation*, 2006.
- [17] R. Malouf. A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of the Sixth Conference on Natural Language Learning (CoNLL-2002)*, pages 49–55, 2002.

- [18] Daniel Marcu. The rhetorical parsing of natural language texts. In *Meeting of the Association for Computational Linguistics*, pages 96–103, 1997.
- [19] Daniel Marcu. The rhetorical parsing of unrestricted natural language texts. In *Proceedings of ACL/EACL*, pages 96–103, July 1997.
- [20] Daniel Marcu. The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics*, 26(3):395–448, 2000.
- [21] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of English: the Penn Treebank. *Computational linguistics*, 19:313–330, 1993.
- [22] E. Miltsakaki, R. Prasad, A. Joshi, and B. Webber. The penn discourse treebank.
- [23] Guido Minnen, John Carroll, and Darren Pearce. Applied morphological processing of english. *Nat. Lang. Eng.*, 7(3):207–223, 2001.
- [24] R. Quirk, S. Greenbaum, G. Leech, and J. Svartvik. *A Grammar of Contemporary English*. Longman, London, 1972.
- [25] Deepak Ravichandran, Eduard Hovy, and Franz Josef Och. Statistical QA - classifier vs re-ranker: What’s the difference? In *Proceedings of the ACL Workshop on Multilingual Summarization and Question Answering—Machine Learning and Beyond*. Association for Computational Linguistics, 2003.

- [26] Dan Roth and Wen-tau Yih. A linear programming formulation for global inference in natural language tasks. In *Proceedings of the 8th Conference on Natural Language Learning*, 2004.
- [27] Radu Soricut and Daniel Marcu. Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of Human Language Technology and North American Association for Computational Linguistics*, Edmonton, Canada, 2003.
- [28] Caroline Sporleder and Mirella Lapata. Discourse chunking and its application to sentence compression. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 257–264, Morristown, NJ, USA, 2005. Association for Computational Linguistics.
- [29] Penn Discourse Treebank. Annotation guidelines for the penn discourse treebank. Manual, University of Pennsylvania, Institute for Research in Cognitive Science, 2003. Downloads as postscript. Also available as a webpage <http://www.cis.upenn.edu/dltag/annotation-manual/annotation-manual.html>.
- [30] B. L. Webber, A. Knott, M. Stone, and A. Joshi. Anaphora and discourse structure. *Computational Linguistics*, 29(4):589–637, 2003.
- [31] Ben Wellner and James Pustejovsky. Automatically identifying the arguments of discourse connectives. In *Proceedings of the 2007 Joint Con-*

ference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pages 92–101, 2007.

- [32] Florian Wolf and Edward Gibson. Representing discourse coherence: A corpus-based study. *Computational Linguistics*, 31(2):249–287, 2005.
- [33] Florian Wolf, Edward Gibson, Amy Fisher, and Meredith Knight. The discourse graphbank: A database of texts annotated with coherence relations. Linguistic Data Consortium, 2005.
- [34] Nianwen Xue and Martha Palmer. Calibrating features for semantic role labeling. In *Proceedings of EMNLP*, 2004.

Vita

Robert Blaine Elwell was born in Schenectady, New York on 5 June 1984. He received the Bachelor of Arts degree in Linguistics from the State University of New York at Albany in 2005, and entered the graduate program in Linguistics at the University of Texas at Austin. He has followed a research program of computational linguistics and data-intensive linguistic theory. During this time, he has found gainful employment using his linguistic knowledge as a transcriptionist as well as research assistant and teaching assistant positions in the area of computational linguistics.

Permanent address: 2501 Wickersham Ln Apt 1132
Austin, Texas 78741

This thesis was typeset with L^AT_EX[†] by the author.

[†]L^AT_EX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's T_EX Program.