# A Semantic Approach to Antecedent Selection in VP Ellipsis

- Master Thesis -
Master Human-Machine Communication

Dennis de Vries
1211986
dndevries@gmail.com

Internal advisor: Jennifer Spenader, University of Groningen
External advisor: Johan Bos, University of Rome "la Sapienza"

## Abstract

Consider this example of Verb Phrase Ellipsis (VPE):

> The man [$_{ANT1}$ stood up because the door bell [$_{ANT2}$ rang]], but his son [$_{VPE}$ didn't].

Of the two possible antecedents, ant1 is the correct one. In earlier studies by Hardt (1997) and Nielsen (2005), syntactical features of candidate antecedents were used to determine the most plausible antecedent. With their methods, they reached accuracies of 84% and 79% respectively.

Inspired by the ongoing theoretical debate on whether ellipsis is resolved syntactically or semantically, the research described in this thesis elaborates on these studies by adding a number of semantic features. To acquire semantic information from discourse, I use Boxer (Bos, 2005), a semantic parser that constructs Discourse Representation Structures (Kamp and Reyle, 1993) from syntactically parsed discourse. These semantic features are (1) semantic similarity of VPE and antecedent subjects, (2) parallelism of propositional phrases, (3) similarity in tense and (4) similarity in modality. To determine semantic similarity of nouns in (1) and (2), I use WordNets path distance measure.

Like in Hardt (1997), a scoring mechanism is used to determine which of the possible antecedents is the most plausible one. Each feature that an antecedent may or may not have contributes to the score of an antecedent with a particular positive or negative value. These values are optimized using a Genetic Algorithm with close to 400 manually annotated examples of VPE from the Wall Street Journal part of the Penn Treebank.

The added features have shown no improvement over baseline performance. A number of possible reasons for this low performance and suggestions for improvement are given in the discussion section.

# Contents

# 1 Introduction

Ellipsis, and Verb Phrase Ellipsis in particular, is an important subject in this thesis, and the aim of the thesis can't be properly understood without any prior knowledge of Ellipsis. Therefore, I'll start with a description of Ellipsis, followed by an overview of the research described in the thesis.

## 1.1 Verb Phrase Ellipsis

Ellipsis is a anaphoric phenomenon in which part of a clause is omitted, leaving a syntactically and semantically incomplete sentence. This omission can be interpreted using information that occurred earlier in the discourse. Some different types of ellipsis are shown below. Of every example, the second sentence is the resolved version of the first.

(1) Verb Phrase Ellipsis (VPE): omission of a verb phrase
   a. *John bought a new car, but Bill didn't.*
   b. *John bought a new car, but Bill didn't buy a new car.*

(2) Noun Ellipsis: omission of a noun
   a. *John bought Bill's car and Bill bought Mary's.*
   b. *John bought Bill's car and Bill bought Mary's car.*

(3) Sluicing: omission of an inflectional phrase
   a. *John bought a new car and Bill asked when.*
   b. *John bought a new car and Bill asked when John bought a new car.*

(4) Gapping: omission of all verbal elements from a verb phrase
   a. *John bought a new car and Bill a bike.*
   b. *John bought a new car and Bill bought a bike.*

(5) Pseudo-gapping: crossing between VPE and Gapping
   a. *John bought a new car and Bill did a bike.*
   b. *John bought a new car and Bill bought a bike.*

Because VPE is the subject of this thesis, it is also the type of ellipsis this section mostly deals with. Generally, humans easily interpret ellipsis by using world knowledge and an inherent knowledge of syntax. To automatically resolve ellipsis in an NLP application is a much harder job because this same knowledge that humans use will have to be collected and made explicit to interpret data that is basically missing. Over the years, a large amount of research has been dedicated to ellipsis, and most of it was concerned with theoretical questions about the processes at the basis of ellipsis resolution

and the level on which it is performed. In general, researchers on ellipsis can be split up into two different camps: one which believes that ellipsis is a syntactic phenomena that is resolved by copying syntactic material from the antecedent to the ellipsis site (Williams, 1977; Haik, 1987; Fiengo and May, 1994; Hestvik, 1995; Kennedy and Merchant, 2000), and one that thinks that ellipsis is resolved semantically, using information from a semantic representation of discourse (Darymple et al., 1991; Kehler, 1993; Shieber et al., 1996.; Hardt, 1999).

Each of these two approaches has shown its advantages and each is able to resolve certain types of VPE sentences that the other can't. An advantage of the syntactic approach is, as Nielsen (2005) puts it, that it can predict unacceptable interpretations using syntactic constraints. Take for instance this sentence containing VPE, of which the resolution is shown within brackets.

(6) John loves his wife, and Bob does too. [love his wife]

When resolving this sentence, there are two interpretations possible because of the pronoun "his". In the first, Bob loves his own wife (the *sloppy* interpretation), and in the second, Bob loves Johns wife (the *strict* interpretation. The meaning of this sentence is ambiguous, but that of the next sentence isn't.

(7) * John$_i$ defended himself$_i$, and Bob$_j$ did too. [defend himself$_i$]

This sentence is shown in a strict interpretation where only a sloppy one is possible. A syntactic constraint from Chomsky's Binding theory called Condition A (Chomsky, 1981) states that a reflexive pronoun must have an antecedent within its local clause. When this constraint is incorporated in a syntax based method of VPE resolution, (7) is shown to be unacceptable. More syntactic constraints like this one that can predict unacceptable interpretations of VPE exist and because these interpretations are seen as acceptable by semantic approaches, they can be seen as an argument in favour of the syntactic approach.

There are also examples of VPE sentences that require more interpretation, or semantics, to resolve. Take for instance these two examples:

(8) Mary wants to go to Spain and Fred wants to go to Peru, but because of limited resources, only one of them can. [go to Spain or go to Peru]

(9) A lot of this material can be presented in a fairly informal and accessible fashion, and often I do. [present a lot of this material in a fairly informal and accessible fashion]

The VPE in (8) has a split antecedent, and in (9) the antecedent is passive, while the VPE is active. Both antecedents of these examples don't have

a syntactic form suitable for VPE resolution with a syntactic approach. Simply copying syntactic material from the antecedent to the VPE isn't enough here, more semantic inference is necessary.

## 1.2 Overview

Almost all of the many papers on ellipsis approach the subject on a theoretical level, exploring how it is interpreted by humans and often only dealing with difficult ambiguous cases, where the correct antecedent is considered a given. Though this is important research that offers great insight in the complex process of ellipsis resolution, it doesn't always contribute to the goal of creating a full system of ellipsis resolution. To realize such a system, two more steps other than the actual resolution will have to be taken first. Cases of ellipsis will have to be identified automatically first, followed by selection of the correct antecedent corresponding to the ellipsis. Once this is done, the resolution of the ellipsis, interpreting its correct meaning, can be performed.

Any method for identifying ellipsis and selecting the antecedent will have to be trained and tested on real data. Only in this way can a robust system be created that best resolves ellipsis in cases that are most frequent in real data, instead of mostly dealing with interesting but rare cases. Still, very little empirical research has been done in this area. To my knowledge only three such empirical studies have been done on ellipsis resolution (Hardt, 1997; Nielsen, 2005; Bos, 2007), which are described in Chapter 3. For a large part the current research is an extension of these three studies, focusing on the problem of selecting antecedents in VPE. The reason for choosing to cover only cases of VPE and ignore other types of ellipsis are that earlier research did the same, that VPE is the most frequent type of ellipsis, providing the most examples, and that it keeps the method simpler. Many things that can be said about VPE also go for other types of ellipsis, so a good method for VPE resolution could be expanded to incorporate other ellipsis types as well.

As explained earlier, research on ellipsis and VPE is divided into syntactic and semantic approaches. Although this distinction generally applies to research on the final resolution step of ellipsis, it can also be applied to antecedent selection and maybe even to ellipsis detection. Hardt (1997) and Nielsen (2005) approach these problems with a rather syntactic method, using syntactic features to identify cases of VPE and their antecedents. On the other hand, Bos (2007) introduces the basis for a more semantic method, using semantic representations of discourse instead of syntax.

This thesis describes a method for identifying VPE antecedents based on the semantic approach taken from Bos (2007). I try to avoid getting into the

theoretical syntax-semantics debate, but instead try to determine whether the use of semantic information in an empirically based system of ellipsis resolution offers better results than the use of syntactic information alone. I use Discourse Representation Structure from DRT (Kamp and Reyle, 1993) which was also used in Bos (2007) as the basis for an empirically based, semantic approach to antecedent selection for VPE. Antecedent selection is performed by looking at (semantic) features of candidate antecedents. This selection method is optimized by using a Machine Learning algorithm called a Genetic Algorithm to train the system on manually annotated examples of VPE from corpus data. So, with a minimum of theory on ellipsis I try to optimally identify VPE antecedents just by training on real examples.

Chapter 2 gives a description of the semantic discourse representation used in this project. Chapter 3 gives an overview of earlier research on empirical approaches to VPE resolution. Chapter 4 on research methods describes the used corpus data, preprocessing of that data, the list of antecedent features, the antecedent scoring mechanism and the Genetic Algorithm. Chapters 5, 6 and 7 respectively contain the results, the discussion and the conclusion.

# 2 DRT

In order to introduce semantics into our empirical method of resolving VPE, a semantic representations have to be constructed of the empirical data on which the method is based. For this we use a semantic theory called *Discourse Representation Theory* (DRT) by Kamp and Reyle (1993), which describes how discourses can be translated into logical semantic representations called *Discourse Representation Structures* (DRS). First I explain the basics of DRT and then I describe a method for VPE resolution in DRT introduced by Bos (2007).

## 2.1 Basics of DRT

A single DRS consists of a number of so-called discourse referents, representing entities, and a number of logical predicates called DRS-conditions, representing semantic relations amongst these entities. A DRS-condition can either be predicate or another DRS. The visual representation of a DRS is typically in the form of a box, with the collection of discourse referents at the top, called the DRS's *universe*, and the collection of DRS-conditions beneath it. Consider the following discourse example taken from Kamp and Reyle (1993):

(10) John owns Ulysses. It fascinates him.

The DRS corresponding to these two sentences is shown in figure 11.

(11)

| $a\ b\ c\ d\ e\ f$ |
|---|
| $Jones(a)$ |
| $Ulysses(b)$ |
| $event(c)$ |
| $own(c)$ |
| $agent(c, a)$ |
| $patient(c, b)$ |
| |
| $a = d$ |
| $b = e$ |
| $event(f)$ |
| $fascinate(f)$ |
| $agent(f, e)$ |
| $patient(f, d)$ |

As you can see, this DRS contains a combined semantic representation of two sentences, and in fact, any number of sentences can in theory be represented by a single DRS. One of the advantages that this brings is that it creates possibilities for resolving anaphoric elements. For example, the second sentence in (10) contains two pronouns that refer to objects in the first sentence. When we look at the corresponding DRS, the top six DRS-relations represent the first sentence and the bottom four DRS-relations represent the second sentence. The remaining two relations show the way in which the pronouns in sentence 2 refer to objects in sentence 1 by stating which discourse referents in sentence 2 are equal to which discourse referents in sentence 1. This shows that a DRS of multiple sentences combined can contain more information than the separate DRSs of those sentences.

The general structure of DRSs might need some further clarification. As you can see in figure 11, sentences quantify over events, which are associated with a certain head verb (in this case "own" and "fascinate"). These events can in turn be modified by adverbs and adjuncts and optionally are related to an *agent*, a *patient* and/or a *theme*, which are standard thematic roles that can be filled by a discourse referent. An example of a DRT containing an event which has an agent and a theme role and is modified by both an adverb ("quick") and an adjunct ("yesterday") is shown in figure 12.

**John quickly drove home yesterday.**

(12)

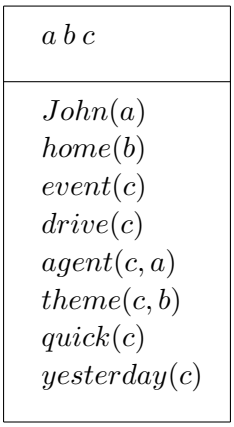
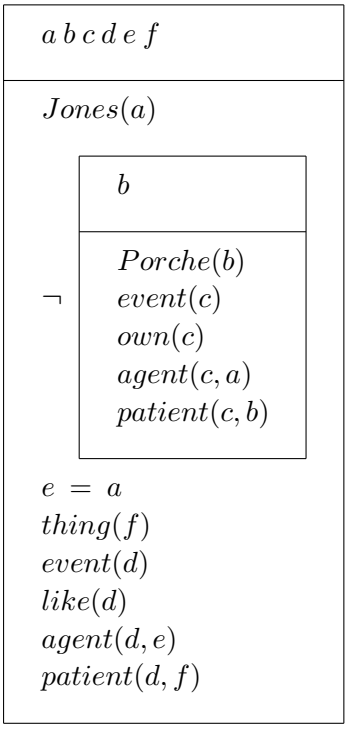



Figure 13 shows the DRS for an example sentence taken from Kamp and Reyle (1993). To cope with the negation in the sentence, the DRS requires an inner DRS that contains the event that is to be negated. Note that the discourse referent that represents "a Porche" is introduced within the inner DRS. In DRT there are accesibility rules that state that it is always possible to refer to referents that are introduced within the current DRS or an outer DRS, but never to refer to referents introduced in inner DRSs. Following this rule means that referent $f$ introduced by the word "it" cannot be equal

to referent $b$ introduced by "a Porche". This is the way that definites and indefinites are handled in DRT: indefinites are introduced in their local DRS and can't be refered to from the outside and definites are introduced in the global DRS and can be refered to from anywhere. Consequentially, if the indefinite "a Porche" were to be substituted for the definite "the Porche", it would be introduced in the outer DRS, making it possible for "it" to refer to it.

**Jones does not own a Porche. He likes it.**

(13)




## 2.2 VPE resolution in DRT

Similar to how the resolution of anaphores can be represented in DRT, the resolution of Ellipsis and, in our case, Verb Phrase Ellipsis can also be represented in DRT. Although the goal of this project is to correctly identify antecedents for VPE and not to actually resolve the Ellipsis, it is still good to show that DRT is a good framework for Ellipsis resolution and that it can be used at the basis of a full system of VPE resolution. Therefore I will now describe a method for VPE resolution in DRT that was introduced by Bos (2007). His paper describes a preliminary version of a full system for VPE detection and resolution and will be covered in more detail in section 3.3.

An event in a DRS can be seen as a semantic version of a verb phrase,

so in DRT terms, you could say that at a VPE site an event is elided. Resolution of this VPE is performed by finding the correct antecedent event and copying it along with (part of) the DRS-relations that are associated with it to the site of the Ellipsis. Figure 15 shows the DRT corresponding to VPE sentence 14 with the VPE unresolved.

(14) John runs fast and Bill does too.

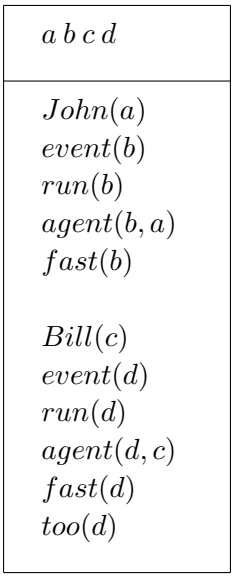
(15)

| $a\ b\ c\ d$ |
|---|
| $John(a)$<br>$event(b)$<br>$run(b)$<br>$agent(b, a)$<br>$fast(b)$<br><br>$Bill(c)$<br>$event(d)$<br>$do(d)$<br>$agent(d, c)$<br>$too(d)$ |

Even though the VP at the VPE site is omitted, it is still represented in the DRT by an event headed by its auxiliary verb "do" and with its subject "Bill" as the agent. This event is identified as VPE because normally the verb "do" always has a patient role[1], which it doesn't have here. During resolution this event will have to be replaced by an event that is constructed from a combination of information from the antecedent event (source event) and the VPE event (target event). Figure 16 shows how the VPE of figure 15 would be resolved.

[1] except for special cases like *do well/good*

(16)




What we see is that some DRS-conditions are copied from the source event to the target event, namely the verb "run" and the modifier "fast". What we also see is that the antecedent's agent was not copied. The reason for this is that as a rule, elements that are parallel between the source and the target event aren't copied during resolution, and both the VPE and the antecedent already have an agent. Take for instance the following example:

(17) John walks in the park. Bill does on the street.

Here, the prepositional phrase *in the park* that modifies the antecedent event is parallel to a similar prepositional phrase that modifies the VPE event. If this antecedent modifier were to be copied, it would result in this incorrect resolution:

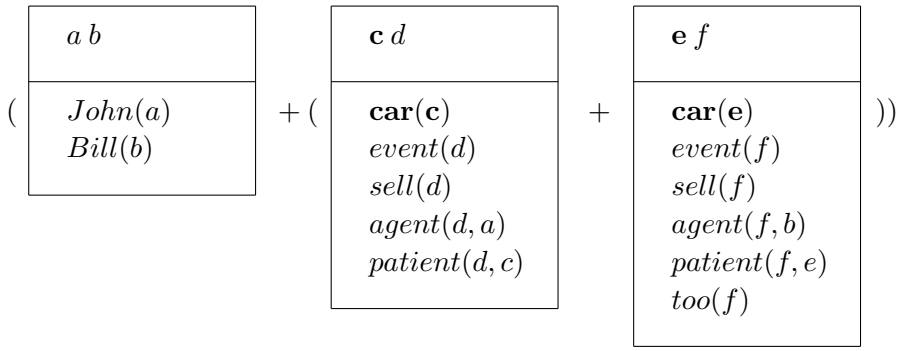
(18) * John walks in the park. Bill walks in the park on the street.

Basically the same goes for thematic roles. If the antecedent event has a type of thematic role that the VPE event has too, than it isn't copied during resolution. This is always the case for the agent role, because all antecedent and VPE events always have at least an agent role. A parallel patient or theme role could also occur, but it might be questionable whether such cases are grammatically correct, like (19) which contains parallel patient roles and would be resolved without copying the patient from the source to the target event.

(19) ? John reads a story. Bill does a poem.

In addition to the rule that parallel modifiers aren't copied to the VPE site, there are three other rules for copying non-parallel discourse referents and their associated conditions from the source event to the target event, which I will illustrate with examples from Bos (2007). To show more clearly how these rules operate, the example DRSs are split into one global DRS, containing all definite discourse referents, and separate DRSs for each sentence in the discourse. This way the distinction between global and local discourse referents becomes apparent. The first rule states that if a discourse referent represents an indefinite and is therefore locally declared within the domain of the source DRS (the DRS containing the source event), then it will be copied to the target DRS (the DRS containing the target event) along with its conditions. This is illustrated by the DRS in Figure 20.

**John sold a car. Bill did too.**

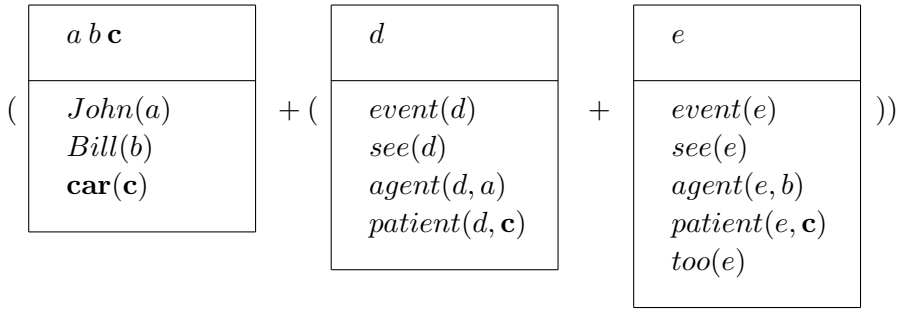



(20)

All definites are defined in the left-most DRS which is the global DRS. The indefinite "a car" which was introduced locally in the source DRS is copied to the target DRS, also introducing the new discourse referent *e*.

Vice versa, the second rule states that if a discourse referent associated with the antecedent DRS represents a definite and is therefore introduced in the global DRS, it won't be copied to the target DRS during resolution. A resolved example of this is shown in figure 21.

**John saw the car. Bill did too.**

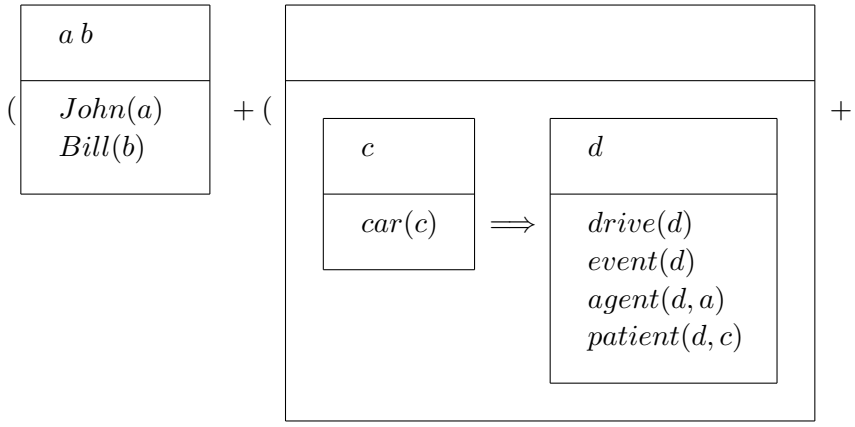


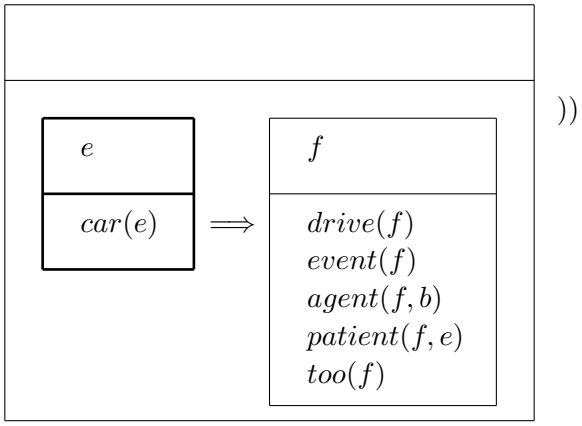

(21)

The discourse referent and DRS-condition for "the car" were introduced globally and therefore aren't copied to the target DRS.

The third and final rule deals with conditionals, which is a subject which I won't go too deep into. For a detailed description of conditionals in DRT, I'd like to refer the reader to Kamp and Reyle (1993). The rule states that if a discourse referent associated with an antecedent is part of a conditional DRS, the antecedent of that conditional is copied to the target DRS. This is illustrated by the resolved DRS in figure 22.

**John drove every car. Bill did too.**




(22)

If the antecedent of the conditional in the source DRS had not been copied, the patient role of the target event would have had to refer to the antecedent of the original conditional, which would result in a wrong resolution:

(23) John drove every car. Bill drove every car that John drove too.

# 3 Earlier research

The first large scale empirical work on VPE with the aim to automate identification and resoution was done by Hardt (1997). In his research he used syntactical features to identify cases of VPE and their correct antecedents in syntactically annotated corpus data. Nielsen (2005) elaborated on this by using various Machine Learning Algorithms to automatically create rules for identifying antecedents with the use of syntactical features. Bos (2007) on the other hand has taken a more semantic approach to the same process. In his research he used the semantic parser Boxer (Bos, 2005), which takes syntactically parsed data as input and produces Discourse Representation Structures (DRS) as output. These DRSs were then used to detect VPE occurrences, select their corresponding antecedents and resolve the Ellipsis. In the following I will present a more detailed description of these three papers.

## 3.1 Hardt (1997)

Hardt (1997) was the first to conduct a corpus based study of VPE resolution. The corpora used in the study were the Brown Corpus (Francis and Kucera, 1982) and the Wall Street Journal Corpus of the Penn Treebank (Marcus, Santorini and Marcinkiewicz, 1993). His method incorporates identification of VPE cases and selection of VPE antecedents.

### 3.1.1 VPE identification

To identify VPE occurrences in the corpora, he used search patterns that looked for missing verb phrases in the syntactical annotations of the corpora. For the Wall Street Journal Corpus the search pattern was:

(24) `(VP (-NONE- *?*))`

which represents a VP containing an empty expression, which could be VPE. Because the Brown Corpus doesn't contain the -NONE- category, a search was conducted for sentences with an auxiliary verb, but no verb phrase. A manual search for VPE occurrences was also performed on a small part of the corpus to test the performance of the automated search algorithm. A recall rate of 44% and a precision rate of 53% was achieved when comparing the automatically found occurrences to the manually found ones. After all corpus texts were automatically searched for VPE, the antecedents of the correctly found occurrences were annotated manually, resulting in a total of 644 VPE-antecedent pairs.

### 3.1.2 Antecedent selection

A system for VPE antecedent selection called VPE-RES was also constructed. To resolve a VPE sentence, first all VP's in a context of three sentences, including the VPE sentence and the two before it, are considered possible antecedents. A syntactic filter is then applied to the possible antecedents, filtering out VP's that on the basis of certain syntactic relations can't possibly be the correct antecedent. In effect this rules out all VP's that contain the VPE in a sentential complement. An example of such a sentence from Hardt (1997) is:

(25) She said she would.

The VP with "would" cannot have the VP headed by "said" as it's antecedent because the VPE is contained in a sentential complement to "said".

After these VP's are ruled out as possible antecedents, the remaining candidates are assigned a preference rating determined by a number of preference factors. These preference factors include recency, clausal relation, parallelism and quotation. Each candidate antecedent is initially given a weight of 1, which is then modified by the preference factors.

If no other preference factors apply, the *recency* preference factor determines that the most recent VP is the correct one. It modifies antecedent weights in such a way that the least recent antecedent's weight gets multiplied by the recency factor, 1.15, and that moving towards the VPE, antecedents get further multiplied by the recency factor. So for example, when there are three possible antecedents, their weights get multiplied by 1.15, 1.32 and 1.52 respectively. Also, antecedents following the VPE are treated the same way, giving a lower value to antecedents that are further away from the VPE.

The next preference factor is what Hardt calls *clausal relation*. When a VPE stands in a clausal relation to a certain VP, then that VP is almost certainly the correct antecedent and is given a very high weight. An example of such a clausal relation from Hardt (1997) is:

(26) All [$_{VP}$ felt freer to [$_{VP}$ discuss things]] than students [$_{VPE}$ had] previously.

In this sentence, the VP headed by "felt", which is the correct antecedent, is modified by the comparative phrase containing the VPE, which means that this VP has a clausal relation with the VPE and gets a high boost. If this preference factor would be deactivated, the more recent VP headed by discuss would be chosen incorrectly.

The third preference factor used by Hardt (1997) is *parallelism*, which can actually be divided into two separate features. Although more forms of parallelism might give clues to which VP is the correct antecedent, only auxiliary verb match and be-do conflict are used. With auxiliary verb match, VP's that contain the same auxiliary verb as the VPE are preferred to VP's that don't. This is shown in the following example from Hardt (1997):

(27) Someone with a master's degree in classical arts who works in a deli would [$_{VP}$ be ideal], litigation sciences [$_{VP}$ advises]. So [$_{VPE}$ would] someone recently divorced or widowed.

Here, the correct antecedent "be ideal" will be preferred over the more recent "advises" because its auxiliary verb "would" matches with that of the VPE. The second parallelism feature, the be-do conflict, gives a penalty to a possible antecedent if it contains an auxiliary of the be-form when the VPE has an auxiliary of the do-form. This constraint was already suggested in Hardt (1992). An example from Hardt (1997) illustrates this:

(28) You [$_{VP}$ know what the law of averages [$_{VP}$ is]], [$_{VPE}$ don't you]?

Neither of the antecedents has an auxiliary verb match with the VPE, but the second one with "is" gets an extra penalty from the be-do conflict factor, so the less recent, correct antecedent is chosen.

The fourth and final preference factor used by Hardt (1997) is *quotation*. This factor supports the rule that when the ellipsis site is within quotes, there is a preference for antecedents that are within quotes too. For example,

(29) "We [$_{VP}$ have good times]." This happy bulletin [$_{VP}$ convulsed Mr. Gorboduc]. "You [$_{VPE}$ do] ? ", he asked between wheezes of laughter.

In this example the correct antecedent is "have good times", which just like the VPE, is within quotes. The quotation preference factor causes the incorrect antecedent "convulsed Mr. Gorboduc" to be penalized because it isn't within quotes while the VPE is, leading to a preference for the less recent, correct antecedent.

After the antecedents are scored using these four preference factors, the highest scoring antecedent is selected as being the most probable one. A last post-filtering step is applied on the selected antecedent. If it contains the VPE clause, that clause is removed from the antecedent. For example, if this post-filtering wasn't performed, the next sentence would be resolved into (30a) instead of (30b).

(30) John talked louder than Bill did.

a. * John talked louder than Bill talked louder than Bill did.
b. John talked louder than Bill talked.

A comparison was made between antecedents chosen by VPE-RES and antecedents chosen by human subjects. Because it is sometimes ambiguous which antecedent is the correct one and sometimes more than one option can be correct, it is possible that the system and the human subject choose different antecedents which are both correct. For that reason three different comparison measures were used, giving three different types of test results:

- Exact Match: The antecedent chosen by the system is a word-by-word match with the one chosen by the human annotator.
- Head Match: The head verb of the antecedents chosen by the system and the human annotator match, but the rest may differ.
- Head Overlap: The antecedent chosen by the system contains the head verb of the antecedent chosen by the human annotator or vice versa. The head verbs don't have to match and the rest of the antecedents may differ.

Of the 644 examples of VPE, 96 were randomly selected to serve as a test set. With a test on this set using the Exact Match measure, the system reached a performance of 76% correct against 14.6% correct on a baseline using only the recency feature. Other tests were performed where along with the recency factor separate features were activated or deactivated and these tests revealed that all used features have a positive effect on the system's performance. The combination of the *syntactic filter*, the *post-filter* and *clausal relations* provided the highest improvement on performance, an increase of 42,4% compared to a recency only baseline.

### 3.2 Nielsen (2005)

This paper describes work that is for a large part an elaboration on Hardt (1997). It's aim is to describe a complete system for resolving VPE, which can be broken down into these three parts:

- Detecting occurrences of VPE
- Finding correct VPE antecedents
- Resolving the VPE

The most interesting step for our project is the second one, selecting correct antecedents, but the other two steps will be explained briefly as well. For

all three steps in his system, Nielsen (2005) used data from the British National Corpus (Leech, 1992) and the Penn Treebank (Marcus, Santorini and Marcinkiewicz, 1993). The former is fully tagged with Part of Speech tags and the latter is annotated with syntactic parse trees. For all experiments he used the available corpus annotations, except for a small side study in which he parsed the corpora automatically and used the resulting syntactic data instead. He manually searched the corpus data and found 637 examples of VPE, which he then annotated for their antecedents.

For the first and the second step in his system, Nielsen uses Machine Learning algorithms that train on VPE examples from the corpus data to learn how to identify VPE and how to select the correct antecedent. A selection of the most commonly used ML algorithms is implemented side by side, not to compare their individual performances, but to test the usefulness of the data. For example, if performance of all algorithms goes up by adding a certain new feature extracted from the data, then that provides more proof for the importance of that feature than if the performance of only one algorithm would go up.

### 3.2.1 VPE identification

To automatically detect cases of VPE in corpus data, Nielsen trained Machine Learning algorithms on the POS information of manually found cases of VPE. The performances on this task by a number of algorithms were tested: Transformation-based Learning (Brill, 1993b), two implementations of Maximum Entropy Modelling (Jaynes, 1957), a Decision Tree learner called C4.5 (Quinlan, 1993) and Memory Based Learning (Stanfill and Waltz, 1986). Of these, Transformation-based Learning and Decision Tree Learning were abandoned at this stage due to technical difficulties and problems with the sparseness of the data respectively. With the Penn Treebank other syntactic methods were also used to detect VPE, along with an extra ML algorithm called SLIPPER (Cohen and Singer, 1999).

On the BNC, using only POS information, an F1 score of 76% was reached. On the Penn Treebank data, an F1 score of 82% was reached, also using extra features derived from the syntactic information of the treebank. In the study where automatically parsed data was used, results dropped to an F1 of 71% due to errors introduced by the parser. Overall, Maximum Entropy Modelling gave the best results

### 3.2.2 Antecedent selection

Because some alterations were made to the annotations of the Penn Treebank since Hardt created his VPE-RES system for identifying VPE an-

Table 1: Features used in ML experiments for antecedent selection. Nielsen (2005) introduced those marked by an asterisk

| Feature | value |
|---|---|
| Recency | integer |
| ∗Sentential distance | integer |
| ∗Word distance | integer |
| Clausal relation | binary |
| Comparative relation | binary |
| Auxiliary match | binary |
| Be-do mismatch | binary |
| Quotes mismatch | binary |
| ∗As-apositive | binaty |
| ∗Polarity | binary |
| ∗Parallel adjuncts | binary |
| ∗VPE-RES | integer |

tecedents in 1997, Nielsen first made a suitable reimplementation of that method so that he would be able to make a fair comparison with his own method. With this reimplementation an Exact Match score of 62.67% was reached, which is considerably lower than Hardt's results due to the changes in the data and algorithm. The Head Overlap measure on the other hand gave a performance of 86.67%, which is more in line with the results from Hardt (1997). This, and an error analysis, lead Nielsen to believe that the Head Overlap measure is a fairer measure and it is used as the standard success criteria in the rest of his research. To improve on the baseline, some new features[2] were added and additionally a number of Machine Learning algorithms were used to optimize antecedent selection based on these features. These algorithms are Decision Trees, Memory Based Learning, two forms of Maximum Entropy Modelling and SLIPPER. The scoring mechanism from Hardt (1997) used continuous numbers, which the above, and most other ML algorithms don't work with. Therefore Nielsen had to use an alternative method. The antecedent features were converted to a feature vector with a limited range of values. The continuous recency value is grouped into a limited number of ranges and the other features are binary anyway. Table 1 shows the features used in Nielsen's ML experiments along with their value types.

When training the ML algorithms on the training data, rules are created that classify VP's as correct or incorrect antecedents, based on the features used to train with. For example, the decision tree classifier produced rules like these:

[2]Note that in Hardt (1997) these features were called *preference factors*.

(31) "**If** a VP is located more than three words after the VPE **and** the VPE auxiliary verb is *do*, **then** that VP is not the antecedent."

(32) "**If** a VP is not the most recent one **and** it is located 4 words or less before the VPE **and** there is no quotes mismatch, **then** that VP is the antecedent."

Nielsen added a number of features that Hardt (1997) hadn't used in his method. Increase in performance from these features was measured by comparing to a baseline implementation in which only recency was used. In the following the added features that improved system performance are discussed. These features are marked with an asterisk in Table 1

First, some experiments with different recency value groupings were performed from which was concluded that a grouping in which the most recent antecedent forms a group, the second most antecedent forms a group and all the other antecedents together form a group produces the best results. Two other recency features were tested, namely sentential distance of the antecedent to the VPE, which offered no improvement, and pure word based distance of the antecedent, which offered only a small improvement.

If a VPE occurs in an *as*-appositive construction, there could be a preference for antecedents with a certain set of features. Adding this as a feature improved Nielsen's results slightly compared to the baseline.

Tests on incorporating polarity as a feature showed that using a boolean representing the disjunction of the polarities of the VPE and the antecedent provided some improvement on the results.

"Parallel adjuncts" describes parallelism between the VPE and antecedent in the form of adjuncts. It can take four values: (1) the antecedent has an adjunct, (2) the VPE has an adjunct, (3) both have identical adjuncts and (4) both have unidentical adjuncts.

(33) a. John worked harder yesterday than Bill did yesterday.
b. John happily ate his sandwich when Bill did angrily.

Sentence (33a) contains a VPE and its antecedent VP which have an an adjunct in common, "yesterday". In sentence (33b), the VPE and its antecedent also both have an adjunct, "happily" and "angrily", but these are non-identical. This feature also gives some improvement to the baseline.

The comparison method of adjuncts that is used is just a simple string comparison, so the words have to be completely identical. Nielsen proposes to use more advanced semantic comparison based on a resource like WordNet (Miller, 1995) to improve the effectiveness of the feature. Such a method of semantic comparison is actually used in the current project (see section 4.3).

The last feature that Nielsen added is an antecedent ranking based on the VPE-RES algorithm by Hardt (1997). As explained earlier, VPE-RES scores antecedents based on their features and a ranking of antecedents can be made based on these scores. The antecedent that gets the highest score based on VPE-RES gets a feature value of 1, the second 2, and so on. This also improved the performance of the baseline.

Nielsen proposed another feature which he could not implement due to technical limitations. In some sentences containing VPE, the correct antecedent can be found by determining whether the subjects of the VPE and the antecedent VP match. The difficulty is that these subjects are formed by pronouns, and in order to find out if they match, pronoun resolution has to be performed. (34) is an example of such a sentence from Nielsen (2007) in which "you" and "he" refer to the same person.

(34) "Do you want to call Eugene?" He didn't [want to call Eugene], but it was not really a question ...

Because no adequate pronoun resolution system was available at the time, this feature couldn't be used.

Tests by Nielsen on the number of antecedents that are considered during antecedent selection showed that a range of 10 candidates before and after the VPE location gives a local optimum in the results, so this range is used in following experiments.

To test the performance of the Machine Learning approach and the new antecedent features, Nielsen performed a cross-validation on all available data, including training and test data. Testing in this manner is a good way to deal with sparse data because as much data as possible is tested on without introducing bias in the results. A detailed description of this testing method is given in section 4.6. It is interesting to note that the original VPE-RES algorithm by Hardt (1997), using the scoring mechanism, outperforms Nielsens Machine Learning approach using the same original features by 2.19% using the Head Overlap score. Although, when Nielsen added the additional features he suggested, perfomance surpassed that of Hardt. The top performing configuration using the original features, the extra features and limiting the amount of antecedents to 10, reaches a Head Overlap score of 85.87% and an Exact Match score of 54.79%. In comparison, Hardt (1997) reached a much higher score of 76% using the Exact Match score. But to make a fair comparison, when reimplementing VPE-RES on the new version of the Penn Treebank, an Exact Match score of only 53.06% was achieved, which is lower than the 54.79% that was reached using Nielsens ML approach.

In the experiments using re-parsed data, results were, understandably, lower. The re-implementation of Hardt's algorithm achieves a Head Overlap

score of 75.71%, which is 4.2% lower than the experiments on annotated data. Using the ML approach with extra features, a Head Overlap score of 77.68% is obtained, 8.19% lower than the experiments on annotated data.

### 3.2.3 VPE resolution

In this part of his system, Nielsen's goal is to construct readable, grammatically correct sentences in which the VPE is resolved. The strategy that he uses here is one of syntactic reconstruction, because he considers this simpler than a semantic approach. A semantic system of VPE resolution, he says, would theoretically be capable of handling all types of cases encountered in empirical data, but would also be very complex and consequently difficult to construct. So instead of looking at the deeper discourse structure, he tries to classify cases of VPE according to their syntactic or surface form. For some of the simple types of those classes he defines rules that will automatically and correctly construct a grammatical sentence in which the VPE is resolved, given the correct antecedent VP. Nielsen distinguishes between 17 different types of VPE which are each resolved in a distinct way. He further categorizes these classes into trivial cases, intermediate cases and difficult cases.

An example of a class that belongs to the trivial cases is one where simply copying the antecedent VP to the ellipsis site is sufficient, except for a small adjustment that will have to be made in case of negation. Sentence (35) illustrates such a case.

(35) Jewelry makers rarely pay commissions and aren't expected to [pay commissions] anytime soon.

Intermediate cases involve more rewriting of the VPE site, as is the case in VPE's that have *which* as their object. Sentence (36) is resolved into sentence (37) by copying the antecedent and substituting *which* for the word *but*.

(36) If he didn't talk sense, which he does.

(37) If he didn't talk sense, *but* he does *talk sense*.

Some of the difficult cases are those that contain pronominal ambiguity, that don't contain an explicit antecedent VP or that don't contain an antecedent at all and therefore require inference. Take for example sentences (38) in which the antecedent is unspoken, but can still be inferred through interpretation.

(38) a. Cigarette ?

b. No, I didn't think you would. [smoke/want a cigarette]
c. You don't mind if I do ? [smoke a cigarette]

Nielsen implemented a system for resolution that first classifies cases of VPE into the 17 classes he defined. Only for the trivial cases he then applies a number of transormation rules, one for each class, which should produce a grammatical sentence in which the VPE is resolved. With this system, he achieves a successful resolution rate of 80.97% on trivial cases.

## 3.3 Bos (2007)

In this unpublished paper, preliminary research on a new approach to VPE resolution is described. The main difference of this approach compared with the two previously described is that it is based on semantics instead of syntax. The data that is used in this research isn't syntactically annotated, but is parsed using a combination of a syntactic and a semantic parser. The data is first parsed by the syntactic parser, the C&C parser (Clark and Curran, 2004) which is based on Combinatory Categorial Grammer (CCG) (Steedman, 2001). The syntactically parsed data is then parsed again by a semantic parser called Boxer (Bos, 2005), which outputs Discourse Representation Structures.

Like in Nielsen (2005), Bos split his method of VPE resolution into three parts: VPE detection, antecedent selection and VPE resolution. The data consists of 139 manually found cases of VPE from the Wall Street Journal section of the Penn Treebank, split into a development and a testing section. For simplicity, only occurrences of VPE with the auxiliary verb *do* are covered, making the assumption that cases with other auxiliary verbs can probably be resolved in the same manner.

### 3.3.1 VPE Detection

The method Bos uses for identifying cases of VPE is the same as the one described in section 4.2. See for example figure 14 which shows the DRS of the sentence "John runs fast and Bill does too." of which Boxer is able to identify the VPE. The version of Boxer used by Bos (2007) marks an event symbolized by the verb *do* as VPE locations when if the only argument associated with it is an agent, like in this example. The only addition to this simple criteria is a filter that rules out occurrences of *to do* that are part of a fixed expression like *having to do something* or *doing good* and also occurrences that are part of a wh-question. On the VPE examples in the test section of the WSJ corpus, this method of VPE detection achieves a precision and recall score of 0.74 and 0.87 respectively, resulting in an F-score of 0.80. These results are better than those reported by Hardt (1997)

and slightly lower than the results from Nielsen (2005) on VPE detection. However, when comparing to the results of the side experiment from Nielsen (2005), using re-parsed data, there is an improvement in F-score from 71% to 80%, though it must be noted that the comparison isn't completely fair due to differences in data size and covered VPE types.

### 3.3.2 Antecedent Selection

Whereas Hardt (1997) and Nielsen (2005) used a variety of antecedent features for selecting the correct one, Bos (2007) utilizes a method which is similar to the baseline method in Hardt (1997), for the only feature that is used is recency. Consequently, the most recent VP before the VPE location is always chosen. In practice this means that in the DRS containing the VPE, the most recent event containing at least an agent role is selected, where recency is measured as the token distance between the VPE auxiliary verb and the head verb of the antecedent. The succes criteria is Head Match, so the head verb of the annotated antecedent must match with the head verb of the automatically selected antecedent. This method results in an accuracy of 72% on the test data, higher than the 62% baseline performance reported by Hardt (1997).

### 3.3.3 VPE Resolution

The method for resolving VPE in DRT that Bos (2007) employs has already been discussed in section 2 and it differs greatly from the method of resolution used by Nielsen. The most important difference is that Nielsen's aim is to form a grammatically correct sentence in which the VPE is resolved, whereas Bos performs the resolution on a more abstract, semantic level represented by DRT.

The result of this difference is that Nielsen has to deal with the many different ways in which VPE occurs on the surface form, leading to the introduction of just as many resolution rules. Due to the abstract nature of DRT, Bos only need to implement four rules: one stating that parallel elements aren't copied to the target DRS and three more concerned with how DRS-relations and discourse referents of non-parallel elements are copied to the target DRS. The difficulties of all the variations in syntactic representation are handled by the parsers, which results in a much simpler and more elegant method of VPE resolution than if a purely syntactic method were employed.

However, Bos (2007) didn't do any tests on this last resolution step, so nothing can be said about the actual performance of his resolution method.

# 4 Methods

In this thesis I present a new method for finding and resolving cases of VPE which is largely a combination of features from Hardt (1997), Nielsen (2005) and Bos (2007). As in Bos (2007) sentences are first parsed by the semantic parser Boxer and like Hardt (1997) and Nielsen (2005) I use features of possible VPE antecedents to determine which antecedent is the most likely candidate. Furthermore I implemented a combination of the antecedent scoring mechanism from Hardt and the machine learning approach from Nielsen.

This section contains descriptions of the data used in this research, the parsing steps applied to the data, the scoring mechanism and features used in antecedent selection and finally the genetic algorithm which optimizes the antecedent selection process.

## 4.1 Data

The data that was used both for training and testing the algorithm was taken from the Wall Street Journal (WSJ) part of the Penn Treebank. This corpus consists of about 1 million words in about 50,000 lines. For training and testing purposes of this research only discourse fragments containing VPE occurrences are necessary, so these have to be extracted first. While Hardt and Nielsen used automatic methods for locating occurrences of VPE in the corpus, this research focuses on the resolution of VPE, so therefore we wanted to use a set of VPE sentences which is as complete and correct as possible. For that reason we manually searched the whole WSJ corpus for cases of VPE, annotating their location and their correct antecedent. This was done by using a tool that searches the corpus for all auxiliary verbs and shows each of them, including their context, to the annotator who then decides wheter it is a VPE location or not. The annotation scheme that was used consisted of the word location of VPE site, which is the location auxiliary verb preceding the ellipsis, and the start and end word indexes of the correct antecedent. This means that antecedents have to be continuous and can't be split up in the middle. This was no problem however, because no cases of VPE with discontinuous antecedents were encountered. The corpus was split in three parts and each part was annotated by one annotator. One section of the corpus was annotated by all three annotators to determine if there were differences in annotation between annotators. Using this annotation method, we found 399 cases of VPE, which means that VPE occurs about once every 125 sentences in the WSJ corpus.

## 4.2 Preprocessing

As already mentioned several times, to obtain semantic representations of our data, it was first syntactically parsed by the C&C parser into CCG and then semantically parsed by Boxer into DRT. The C&C parser by Clark and Curran (2004) is a statistical parser trained on the CCGBank (Hockenmaier, 2003), which is identical to the Penn Treebank, but with its annotations converted to CCG. Apart from a CCG derivation, the parser also outputs POS-tags, lemmas and types of named-entities. Boxer converts all this information into DRSs. Its standard output format is Prolog, but in this research the XML output format was used. For readability, Boxer can also present its output in the graphical box format in which all DRT examples in this thesis and other literature are presented.

Boxer is also able to indentify cases of VPE and make a list of their possible antecedents. The version of Boxer used in Bos (2007) was only able to identify VPE with the auxiliary verb "do", but the one used here has been expanded by Bos to include the following types of auxiliary verbs:

(39) do, be, have, can, could, may, must, might, will, would, shall, should

In addition to this, VPE sentences with the word "to" like example (40) are also detected by this new version of Boxer.

(40) John is going to work, but Bill doesn't want to.

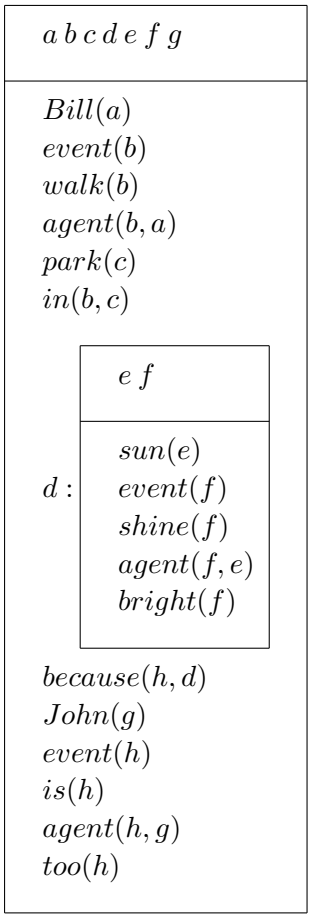
An inspection of the data showed that antecedents for VPE never occur more than two sentences before the sentence containing the VPE. Therefore, for each VPE example we parse the sentence containing the VPE and the two preceding sentences, resulting in one DRS representing all three sentences. From this output the data needed for our experiments are extracted: the VPE event along with its adjuncts and thematic roles (agent/patient/theme), and possible antecedent events that are automatically found by Boxer along with their adjuncts and thematic roles. Also, for each of these candidate antecedents feature values are determined and stored. Section 4.3 lists all antecedent features used in this research and shows how each feature is extracted from the data.

The following constructed example will illustrate this process in a comprehensive manner. Take the following discourse containing VPE in the last sentence and 2 events which are considered possible antecedents:

(41) a. Bill is walking in the park.
b. Because the sun is shining brightly, John is too.

Figure 42 shows the corresponding DRS:

(42)




In this example, the VPE event is denoted by referent $h$, which has an adjunct "too", and and agent "John". There are two possible antecedent events, $b$ and $f$, of which the correct antecedent is event $b$ headed by the verb "walk". Event $b$ has an adjunct, *in(b,c)*, representing "in the park", and an agent "Bill". Event $f$ also has an adjunct, "bright", and an agent "sun". This information is stored in a format illustrated in Table 2.

Because the XML format of DRSs produced by Boxer contains information on word position, the surface form of each antecedent can be reconstructed by determining the minimum and maximum word positions of all its associated DRS conditions, excluding parallel elements like the agent relation. This way the surface forms of the two possible antecedents, "walking in the park" and "shining brightly", are extracted. In addition to parallel elements, elements that are related to the antecedent event and at the same time contain the VPE event at some deeper level are excluded as well. The effect of this last step is similar to that of the post-filter in Hardt (1997) (see example (30)).

The information in Table 2 is then extended with features based on the

Table 2: Data representation of the VPE and its possible antecedents in (41)

| VPE event | DRS relations | antecedent events | DRS relations |
|---|---|---|---|
| h | adjunct: "too"<br>agent: "John" | b | verb: "walk"<br>adjunct: in(park)<br>agent: "Bill" |
| | | f | verb: "shine"<br>adjunct: "bright"<br>agent: "sun" |

DRS, which will later be used to determine the correct antecedent automatically. The next section describes these features and also shows how they are extracted from the data.

## 4.3 Features

The features that can be used for antecedent selection, are bound by the information that the data provides. In previous research that data consisted of the syntactically annotated corpus data, from which features based on syntax and surface form can be extracted. In this project the data also consists of corpus data, but parsed into DRSs, which contain less information about syntax and surface form, but more about semantics. Therefore some features used in previous research could not be used here and some new features have been added.

This section describes the features that were used in this research one by one and also explains why some features used in other research weren't implemented here.

### 4.3.1 Recency

Earlier research has shown that recency is the most important feature for identifying the correct antecedent. In Nielsen (2005) it accounts for by far the largest improvement of results of all used features and in Bos (2007) it was the only feature used, basically always choosing the most recent antecedent, while still returning good results.

There are a couple of ways in which recency can be implemented. One is ranked recency, in which you number each antecedent according to how many antecedents stand in between it and the ellipsis site. Antecedents which have the same head verb get the same recency rank. Then there

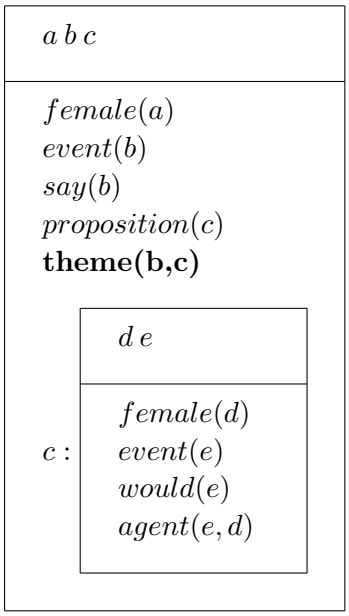
is word distance, which is determined by the word distance between the auxiliary verb of the VPE and the head verb of the antecedent. The third recency measure is sentence distance, which tells how many sentences the antecedent is removed from the VPE. These three recency measures have all been implemented as separate features in varying combinations.

### 4.3.2 Sentential Complement

The filter that both Hardt and Nielsen used to filter out antecedents that contain the VPE in a sentential complement has been implemented here as well. In DRT, these complements are identified by the fact that they stand in a theme relation to the VP in question, as in Figure 43. Note that this feature is not used for training the Genetic Algorithm. Instead, antecedents that have this feature are simply removed from the list of candidate antecedents.

(43)

**She said she would.**



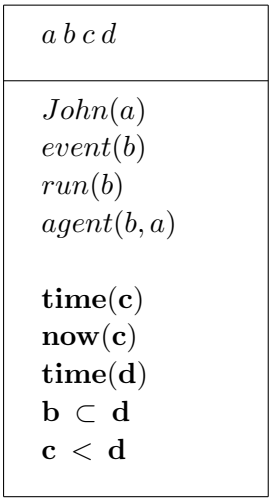

### 4.3.3 Tense Match

No previous empirical studies have tested whether there is a preference for antecedents with matching (or mis-matching) tense compared to the VPE, so since Boxer is able to incorporate tense information in its output, this is a good opportunity to do so. Tense match is a boolean feature that is 1 when the tense of the verb at the ellipsis site is equal to the tense of the head verb of the antecedent and 0 when it is not. Figure 44 shows how temporal

information is represented in DRT.

(44)

**John will run**



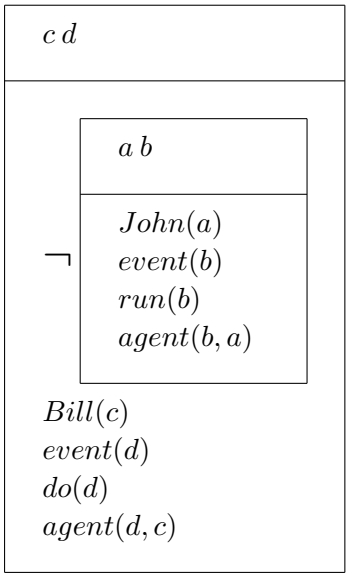

For each event there are two time predicates, one representing the time frame of the event and one representing the present. Another DRS-relation (the bottom one in figure 44) shows how these time frames relate to each other. In this case the present is shown to take place before the time frame of the event, which indicates that the event is in future tense. Three tenses are possible: past, present and future. For improved readability, this type of tense information is left out in all other DRS examples throughout this thesis.

### 4.3.4 Polarity Match

In line with tense match, there could be a preference for matching polarity between ellipsis and antecedent. Nielsen (2005) already incorporated this feature in his work and found that it offered a small improvement. Like tense match, the polarity match feature has a boolean value that is 1 when there is a match and 0 when there isn't. Figure 45 shows an example of a DRS for a VPE sentence in which there is a polarity mismatch between the VPE and its antecedent. It has been taken into account that an event that is nested deeper in sub-DRSs might be negated twice, canceling its negation.

**John doesn't run, but Bill does.**

(45)




### 4.3.5 Modal Match

Boxer is able to determine the logical modality of a DRS. There are two forms of logical modality: possibility and necessity. We don't have any evidence that tells us if there is a preference for antecedents with a matching modality, but using Machine Learning, this can be established empirically. In fact, it might also turn out that there is a preference for mismatching modalities. Like the previous two features, this one is also represented by a boolean value.

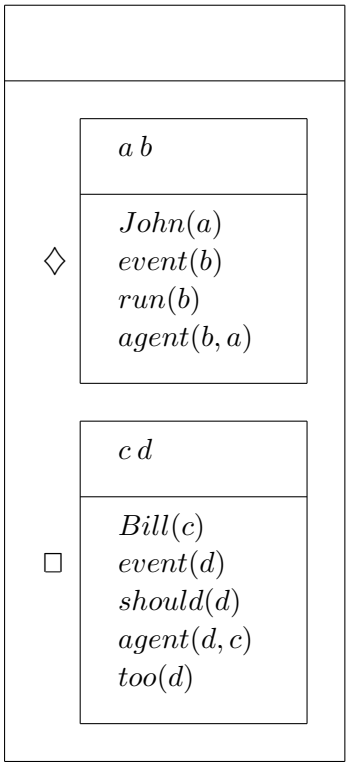
Figure 46 illustrates this feature. A *possibility* modality of a DRS is depicted by a diamond symbol and a *necessity* modality is depicted by a square symbol. In this example the VPE and its antecedent don't have matching modalities. It often occurs that a DRS doesn't have a modality, and it can therefore also occur that the source event's DRS doesn't have modality while the target event's DRS does, or vice versa. These cases are also considered as a modality mismatch. As with tense information, modality information isn't included in any other DRS examples in this thesis to ensure readability.

The modality of a clause is expressed through a modal verb. The modal verbs that correspond to each modality are:

- Necessity: must, will, would, shall and should
- Possibility: can, could, may and might

**John might run and Bill should too.**

(46)




### 4.3.6 Clausal Relation

This feature is implemented in the same way as in Hardt (1997), as described in section 3.1.2. To recapitulate, a case of VPE stands in a clausal relation to a possible antecedent VP if the clause which contains the VPE modifies that VP. This is the case in the following example.

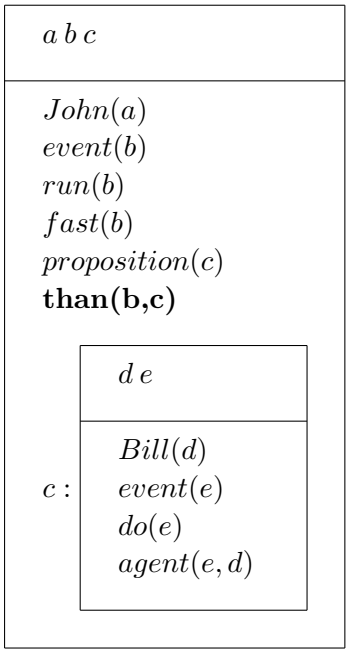
(47) John runs faster than Bill does.

There is a very strong preference for antecedents that contain the VPE in such a relation, so it is probable that this feature will give a good improvement on performance. The value representing this feature is a boolean.

A clausal relation between antecedent and VPE is found in a DRS when the VPE is located in a DRS that is subordinate to the DRS containing the antecedent event and when this sub-DRS has some relation to the antecedent event which isn't a theme relation (for then the sentential complement filter would kick in). See for example figure 48 which shows a DRS for a sentence in which the VPE is located within a comparative related to the antecedent.

**John runs faster than Bill does.**

(48)




### 4.3.7 Subject Similarity

This is a feature that hasn't been used by Hardt (1997) and Nielsen (2005) and it brings a bit more semantics into the method. The idea is that there could be a general preference for antecedents which have a subject that shows parallelism to the subject of the VPE, in the sense that they are semantically similar to each other. To illustrate:

(49) The bus [$_{ant1}$ stopped because the light [$_{ant2}$ went red]] and the car [$_{vpe}$ did] too.

The words "car" an "bus" are more similar semantically than "car" and "light", so with the subject similarity feature the correct antecedent *ant1* will be chosen over the more recent *ant2*. A simple method involving WordNet (Miller, 1995) is used for determining semantic similarity. WordNet has a function called path similarity which returns a semantic similarity measure for two terms by determining how long the shortest path between the two terms through the thesaurus network is.

In WordNet, nouns are represented in their base, single forms, but in natural text, nouns occur in plural and other forms as well. To counter this problem, WordNet has a function that reduces inflexions of nouns to their base form. Another problem is that WordNet does not contain named entities, but we would still like to be able to compare them. As mentioned in

section 4.2, the C&C parser is able to identify named entities, categorizing them to one of these types: location, organisation, e-mail, url, person, title or just name when unknown. To be able to evaluate their semantic similarity with other nouns, named entities are replaced by the word describing their category. More will be said on this feature in the discussion section.

### 4.3.8 Parallel Prepositional Phrases

Other elements that can show parallelism between the ellipsis and its antecedent are prepositional phrases (PP).

(50) Bill [$_{ant}$ jogs] on the street, but he [$_{vpe}$ doesn't] in the park.

(51) Bill [$_{ant}$ jogs on the street], but he [$_{vpe}$ doesn't] on Wednesday.

These two examples show that although VPE and antecedent both have a PP, that doesn't automatically mean that they are parallel. The PP's "on the street" and "in the park" in (50) are parallel, but in (51) the PP's "on the street" and "on Wednesday" aren't. This also has an impact on what is included in the antecedent. In (50) the parallel PP "on the street" isn't part of the antecedent whereas the same PP is included in the antecedent of the VPE in (51).

This principle of parallel PP's actually generates extra possible antecedents. Here is an example of a sentence in which the antecedent and the VPE both have two PP's:

(52) Bill [$_{ant}$ jogs with his dog] on the street, but he [$_{vpe}$ doesn't] after dark in the park.

Because it isn't possible to determine directly which PP's are parallel and which aren't without any semantic information, all combinations of PP's will have to be considered, leading to as many antecedents as there are possible combinations. Here are a few of the combinations possible in sentence 52 of which the first is the correct one (Equal indexes correspond to parallelism between PP's):

(53) Bill jogs with his dog [$_{1}$ on the street], but he doesn't after dark [$_{1}$ in the park].

(54) Bill jogs [$_{1}$ with his dog] [$_{2}$ on the street], but he doesn't [$_{1}$ after dark] [$_{2}$ in the park].

(55) Bill jogs [$_{1}$ with his dog] on the street, but he doesn't after dark [$_{1}$ in the park].

And this is how the VPE in sentences 53 - 55 would be resolved:

(56) Bill jogs with his dog on the street, but he doesn't jog with his dog after dark in the park.

(57) Bill jogs with his dog on the street, but he doesn't jog after dark in the park.

(58) Bill jogs with his dog on the street, but he doesn't jog on the street after dark in the park.

It is likely to assume that when two PP's are parallel, they will also show similarities to each other. So to determine whether two PP's are parallel, it might be a good idea to measure how similar they are. There are two elements which play a role in similarity between two PP's, the prepositions and the objects, and both are used as a feature within the parallel PP's feature. Each parallel PP within a possible antecedent will get a score based upon a general parallel PP feature value, match of the prepositions and similarity of the PP objects based on the previously described WordNet measure.

Looking at (53), we see that although the prepositions of the parallel PPs, "on" and "in", don't match, their objects, "the street" and "the park" do have semantic similarity because they are both locations. This is an indication for their parallelism.

### 4.3.9 Features not used

The DRSs from Boxers output don't contain any auxiliary verb information. This means that any feature used by Hardt (1997) or Nielsen (2005) which is based on auxiliary verbs can't be used in our system. This includes the auxiliary match, which determines if the antecedent and VPE have the same auxiliary verb, the *be-do* conflict, which gives a penalty to antecedents containing a form of *be* if the VPE contains a form of *do*, and finally the auxiliary form feature, which simply includes the plain auxiliary verb types of VPE and antecedent as separate features. It should be pointed out however that DRS modality, which is used as a feature, is based on auxiliary verbs, so indirectly some auxiliary verb information is used in antecedent selection.

Another feature that isn't used in the current research is quotation. As with the auxiliary verb features, quotation wasn't used because the DRSs produced by Boxer don't contain any information about quotation. Still, it could be possible to implement quotation as a feature, but the information would have to be extracted from the surface form of the sentence, not from the DRS itself.

Nielsen (2005) introduced a simplistic feature based on parallelism of adjuncts between VPE and antecedent, with which he achieved some improve-

ment. The feature simply compares strings, so the adjuncts have to match exactly. This feature isn't used in our system, but we do use a more advanced feature that describes parallelism of a single type of adjunct, namely prepositional phrases. Our feature doesn't simply compare strings, but checks for semantic similarity. Investigating parallelism of other types of adjuncts should be interesting, but is left for future research.

### 4.4 Scoring mechanism

As said earlier, our method for selecting the most plausible antecedent for a VPE occurrence is similar to the one in Hardt (1997). Each possible antecedent is given an initial score of 0, after which values are added or subtracted according to which features it has. After assigning final scores to all possible antecedents, the one with the highest score is chosen as the most likely candidate. The values which are added to or subtracted from the antecedent scores are fixed for each type of feature, but there are differences in how these values are calculated. For example, if the fixed value for the boolean feature *tense match* is 3 and the VPE has a tense match with an antecedent, then that antecedent will get 3 added to it's score. On the other hand, if the numerical feature *recency* has a fixed value of -0.5 and the head verb of the antecedent is 10 words away from the VPE, the antecedent gets 5 subtracted from it's score.

The reason for choosing this scoring system and not the selection rules implemented by Nielsen (2005) is that some features have numerical values and are therefore less easy to fit into these rules. When using numerical values in rules, they will have to be put into value ranges, like in these two rules from Nielsen (2005) in which the two possible ranges for recency are smaller or equal to 1, and larger than 1:

(59)
```
Rule 37:
   Antecedent auxiliary = to
   Recency <= 1
   In-quotes = not_clashing
   VPE-RES rank <= 2
   -> class TRUE [95.6%]
```

(60)
```
Rule 45:
   Recency > 1
   Word distance <= 4
   In-quotes = not_clashing
   -> class TRUE [95.0%]
```

With such a classification it is impossible to distinguish between two an-

tecedents that for example have recencies of 3 and 5. More value ranges can be added to accommodate this problem, but running a ML algorithm on too many values can degrade its performance.

In Nielsen (2005) recency is the only non-boolean feature, which might make the necessity for using a numerical method smaller, but in the current project two new numerical features, subject similarity and parallel PPs, are introduced. Using the scoring mechanism, the information that these values contain can be fully used, while grouping them into a limited amount of ranges to make them suitable for ML algorithms like Memory Based Learning would degrade their information value.

## 4.5 Genetic Algorithm

To optimize the scores given to the various preference factors a Genetic Algorithm was used. In this section I will first explain the workings of a general Genetic Algorithm and then elaborate on how it has been implemented in the current research.

Genetic Algorithms are more or less based on evolution in biological systems. For example, in nature evolution can cause a population of animals to adapt its behaviour or physical characteristics through survival of the fittest. The individuals that are best equipped for generating offspring in their particular living environment will determine what the next generation of individuals will look like because they produce the most offspring. Exchange of genetic information during mating and slight random genetic mutations will cause offspring to differ from their parents. These changes will also affect their chances of surviving and producing offspring, either for the better or for the worse. But since fitter individuals are more likely to produce offspring, those who have improved will likely populate the next generation.

This process closely resembles how simulated evolution in the form of a Genetic Algorithm works. The individuals that are optimized are hypotheses instead of animals, but for the rest a genetic algorithm is fairly similar to the biological description above. Hypotheses are often represented by a bit string, which is like a genetic code. Like in biology, this code describes the individual's actual form and/or behaviour. The hypotheses which are optimized in the algorithm can be many things, from simple model parameters to entire computer programs. Initially a set of hypotheses, called a population, is randomly created, after which in each step of the algorithm the best individuals of the current population are selected to produce the following one. To determine how 'good' an individuals is and consequently whether it will 'mate' and generate offspring or not, a fitness value is used. How this fitness value is calculated depends on the application, e.g. when

the algorithm is used to optimize model parameters, the fitness value can be determined by how well the model output matches empirical data using the individual's parameter set. Individuals with a high fitness value are likely to be chosen to produce offspring.

New individuals are created by crossover and mutation. In crossover two parents are split in one or more places, after which part of the information from one parent is combined with part of the information from the other to form a new individual. This is illustrated in the following example where individuals are represented by bit strings with crossover points represented by periods:

(61) parent 1: 100.*101*.001
parent 2: *000*.011.*011*

child: 000.101.011

After crossover, mutation is applied to the child individual, randomly flipping some bits in its string:

(62) 000.101.110 $\Longrightarrow$ 00*1*.101.*0*10

A mutation rate parameter determines how high the probability is that a bit will be flipped. A high mutation rate will cause large variability in individuals, leading to a broad search of the hypothesis space without converging to an optimal hypothesis. A low mutation rate will cause the search to be narrow and to converge to an optimal hypothesis more quickly. But when a search is too narrow, there is a high chance that it will converge to a sub optimum. A way to tackle these two problems is to introduce a variable mutation rate. If the mutation rate starts out high and gradually diminishes, it will first give a broad search of the hypothesis space and then converge on the optimal solution.

For the current optimization task in VPE antecedent selection I have chosen to use a genetic algorithm because it is especially useful in optimizing sets of values. Nielsen (2004) didn't use the scoring system introduced by Hardt (1997), because he used various Machine Learning Algorithms that can't handle these scores, so he produced rules like (59) and (60) which only take a limited amount of values as input. As Nielsen himself states, the scoring mechanism by Hardt would provide more flexibility. In addition, because Nielsen's system performs worse than VPE-RES when using the same features defined by Hardt, he argues that the method used in VPE-RES might be superior to his and that implementing his own newly defined fea-

Table 3: Example of an individual from the population of the Genetic Algorithm

| Feature | Value |
|---|---|
| Recency | –0.4 |
| Tense Match | 0.3 |
| Polarity Match | –0.2 |
| Modal Match | 0.2 |
| Clausal Relation | 0.9 |
| Subject Similarity | 0.6 |
| Parallel PP | 0.2 |

tures to VPE-RES might improve performance. Instead of manually finding the optimal score parameters, he proposes to automatically optimize them using Genetic Learning techniques.

So in short, the choice of the combination of VPE-RES's scoring mechanism and a GA was based on two reasons:

1. A Genetic Algorithm is better at handling coninuous values than other Machine Learning algorithms.

2. Nielsen (2005) suspects that selecting antecedents with a scoring method like VPE-RES gives better results than selecting antecedents with classification rules produced by a ML algorithm.

The implementation of the GA for the current research went as follows. In contrast to the description of GA's above, the individuals which form the population don't consist of bit strings, but of the set of score parameters that determine the weights attributed to the various antecedent features. These score parameters are comparable to the values that Hardt used to lower or highten the scores of antecedents based on the preference factors that applied to them. If, for instance, an individual has a recency score parameter of –2, antecedents will get a value equal to their to their recency value multiplied by –2 added to their score. For each antecedent feature there is such a parameter, and each individual consists of a list of these parameters instantiated to a random value. This parameter list can include all features described earlier, or a subset if other combinations have to be tested. An example individual is represented by Table 3, showing its list of features and example values corresponding to those features. This individual will give antecedents a penalty acording to their receny, a boost if their tense matches, a small penalty if their polarity matches and so on. This might not be the best configuration, but that is tested by calculating the individual's fitness value.

The fitness of an individual is determined by testing its parameter set on annotated VPE training sentences. For each training sentence, its annotated antecedent is compared to the top scoring antecedent based on the individual's parameter set. Hardt (1997) and Nielsen (2005) used their three success criteria for this: Exact Match, Head Match and Head Overlap. Because in our annotation scheme the head of the antecedent VP isn't marked, Head Match and Head Overlap cannot be used here. The only remaining usable measure, Exact Match, was already found to be too strict by Nielsen (2005), because antecedent choice can sometimes be ambiguous. There are cases in which the annotator and the system select different antecedents which are both correct but in which Exact Match will classify the found antecedent as wrong. To introduce a milder criteria that overcomes this problem and that is not based on the head of the VP, an f-score based on precision and recall of antecedent words is used:

(63) $$f = \frac{2 \times precision \times recall}{precision + recall}$$

Consider the following example from Hardt (1997):

(64) "In July, Par and a 60% owned unit agreed to plead guilty in that inquiry, as did another former Par official."

System output: plead guilty in that inquiry
Annotator selection: agreed to plead guilty in that inquiry

The system and the annotator came up with different antecedents, so there is no Exact Match, but there is overlap between them. The precision and recall scores of the antecedent selected by the system, based on the words, are 1,00 and 0,71 respectively, producing an f-score of 0,83. The f-scores of all antecedents that are found using the individual's parameter set on the testing examples are calculated. The individual's fitness values is then determined by the mean of all these f-scores.

Creating new individuals from the fittest ones of the parent population is done using crossover and mutation. Crossover is executed in the same manner as in the general description of a GA. Some randomly chosen parameters from one parent are combined with some of another to form a new individual:

parent 1: **var1a, var1b, var1c**
parent 2: var2a, var2b, var2c

child: var2a, **var1b**, var2c

But for mutation there is a bit of a difference. Because an individual doesn't consist of a bit string, but of a list of values, you can't just mutate it by

randomly flipping some bits. Instead, for each parameter value a small random value is added or subtracted. This value is taken from a normal distribution with a standard deviation that is equal to the mutation rate, so in this genetic algorithm the mutation rate's function doesn't define the chance of a mutation taking place for a certain value, but it determines the mean size of mutations on parameter values.

If the genetic algorithm is kept running for a while, the changes in the population's fitness scores should slowly go down, leading to an optimal solution.

## 4.6 *K*-fold Cross-validation

Because the data is very sparse, a testing method is required that make optimal use of it. $K$-fold cross-validation is such a testing method, because it uses all available data both for training and for testing. Normally, training and testing on the same data would introduce bias to the results, but with $K$-fold cross-validation that is not the case. The data is split into $K$ equal parts, after which $K$ experiments are run. In each experiment, a different section (fold) of the data is used as test data while the remaining folds serve as training data. The mean of these $K$ experiments is taken as the final result.

The WSJ corpus consists of 25 sections which were used for splitting the data for cross-validation. Each fold consisted of the VPE examples from 5 sections, leading to a total of 5 folds.

# 5 Results

To test the contributions of different antecedent features to antecedent selection performance, all features were compared to the performance of a baseline. Both Hardt (1997) and Nielsen (2005) used recency as a baseline because this is the most important and simple feature in VPE antecedent selection. They tested other features in combination with recency to test how much these individually improved performance over the baseline.

Interestingly, when tested along with recency, none of the features described earlier improve our system's performance compared to a baseline which uses recency alone. Even when using all other features at the same time no improvement is gained over recency. Experiments by Hardt (1997) and Nielsen (2005) already showed the high influence of recency on system performance, but apparently in our case it completely overshadows the influence of the other features.

To overcome this adversity and to determine if other features than recency contain any information value with respect to VPE antecedent selection, another, simpler baseline than recency is used. With this baseline an antecedent is randomly selected from the list of possible antecedents. Because of this change in baseline a small adjustment has to be made to the antecedent selection mechanism as well. Previously when more than one antecedent had the highest score, the most recent one was chosen, but now that the baseline is random instead of based on recency, recency will have to be taken out of antecedent scoring completely. For that reason an antecedent is randomly selected from the tied winning antecedents in those cases. The results of experiments done using the new baseline are shown in Table 4. For reference the table shows the Exact Match percentages as well, but in the discussion of the results only the f-scores will be considered.

Table 4: Mean f-scores and Exact Match percentages from tests on individual features

| Feature | Mean f-score | Exact Match |
|---|---:|---:|
| Baseline (random) | 0.16 | 5.9% |
| Recency | 0.57 | 28.3% |
| Tense match | 0.18 | 8.2% |
| Polarity match | 0.23 | 10.2% |
| Modality match | 0.15 | 6.3% |
| Clausal relations | 0.28 | 12.2% |
| Agent similarity | 0.25 | 14.1% |
| Parallel PP's | 0.16 | 8.9% |
| All features | 0.57 | 30.0% |

Before we come to the description of the results, some other facts have to be mentioned first. In our testing method we make use of Boxer's automatic indentification of VPE and their possible antecedents. Unfortunately Boxer doesn't always correctly identify VPE and doesn't always include the correct antecedent in it's list of possible antecedents. To be precise, of the total of 399 VPE examples found in the WSJ corpus, there were 95 which Boxer either didn't identify as such, or didn't find any possible antecedent events for. Furthermore, there were 47 examples for which Boxer did find possible antecedent events, but all antecedents corresponding to those events had an f-score of 0, meaning that they don't overlap at all with the annotated antecedent. The first group of problematic examples was excluded from the experiment because they don't contain any information value. The second group of 47 was kept, leaving the total number of used examples at 304.

If for each of these examples of VPE we would take the antecedent with the highest f-score, then the total mean f-score would be 0.626. If Boxer were able to find all the correct antecedent events and our method of extracting the full antecedents from that would work perfectly, the mean f-score of all highest scoring antecedents would of course be 1 instead. This lower score has three causes. The first one is a problem that has to do with how Boxer handles comparatives, which will be explained in detail in the discussion section. Also, sometimes the correct antecedent event isn't found, but a full antecedent derived from another event does overlap with the annotated antecedent, resulting in an f-score between 0 and 1. The final cause is that due to parsing errors by the C&C parser, DRS relations that should have been connected to the antecedent event are connected to another event or vice versa. In short this means that the best performance that we can reach with this method and this data set is a mean f-score of 0.626.

Going back to the results in Table 4, the first observation that we can make is that, as we expected, recency is by far the highest scoring feature. In section 4.3.1, three different kinds of recency were defined, but testing proved that there was no difference in performance between ranked recency and word distance, and that sentence distance didn't give any improvement. For this reason, only word distance was used as a recency feature.

The features *Modality* and *Parallel PP's* show no improvement over the baseline and the increase that *Tense match* provides is only marginal.

An interesting fact is that the scoring parameter for *Polarity match* that is produced after training, is negative. This means that there is a preference for antecedents that have a polarity that mismatches with the VPE polarity, like in:

(65) John runs, but Bill doesn't

As mentioned earlier, none of the features combined with recency produced

a higher f-score than recency alone. However, because Hardt (1997) and Nielsen (2005) both used the exact match score, it is still interesting to look see how my system performed using that measure. It appears that, using the exact match score, all features have a higher performance than the baseline, and implementing all features together outperforms recency alone by 1.7%.

# 6 Discussion

The results are somewhat disappointing and there are a number of things that might be responsible for this.

## 6.1 Corpus size

Compared to previous research, the data set used in this study is very small. Hardt (1997) worked with 644 VPE examples and Nielsen (2005) trained his system on 637 examples. In the current research, only 304 examples were used for training, of which 47 didn't have any information value because Boxer didn't find any antecedents with an f-score higher than 0 for them. This results in a total of 257 VPE examples on which the training results were really based. This is too little data to properly train on, especially if some antecedent features are quite rare, like Parallel PP's.

## 6.2 C&C parser errors

Errors produced by the syntactic C&C parser are also a reasons for the low results. No full investigation of the performance of the parser on the WSJ corpus has been made, but an inspection of VPE examples which got low f-scores on their automatically found antecedents showed that these are often caused by parser error. A number of different types of parser errors encountered in the data are shown in (66) to (68).

(66) The rationale for [$_{ANT}$ responding to your customers] needs faster than the competition [$_{VPE}$ can] is clear ...

In this sentence, the parser interprets the noun "needs" as a verb, changing the whole syntactic structure of the sentence. The antecedent that the system extracts from this is "responding".

(67) Just as the 1980s bull market [$_{ANT}$ transformed the U.S. securities business], so too [$_{VPE}$ will the more difficult environment of the 1990s, says Christopher T. Mahoney, a Moody's vice president.

In this example, the parser labels "the U.S. securities business" as the subject of the verb "will", and "the more difficult environment of the 1990s" as the object of "transformed". The VP headed by "transformed" is successfully chosen as the correct antecedent, but according to the parser the NP headed by "environment" also belongs to that VP. Because an antecedent needs to be continuous in our annotation scheme, the reconstructed antecedent becomes "transformed the U.S. securities business, so too will the more difficult environment of the 1990s".

(68) In addition, further packaging of mortgage-backed securities, such as Blackstone's fund, have reduced the effects of prepayment risk and [$_{ANT}$ automatically reinvest monthly payments] so institutions don't have [$_{VPE}$ to].

In this last example, the parser sees "prepayment risk and automatically reinvest monthly payments" as a conjunction of two NP's, labelling the verb "reinvest" as an adjective. The antecedent that the system finds is "reduced the effects of prepayment risk and automatically reinvest monthly payments".

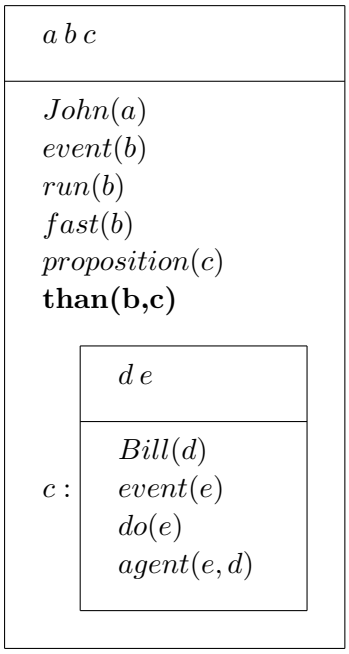
Errors produced by the syntactic parser make it difficult to compare the results to those from Hardt (1997) and Nielsen (2005), because they used annotated data instead. Only the side study by Nielsen on VPE antecedent selection using parsed data is comparable. On the exact match score, he achieved a 40% accuracy compared to 30% accuracy in our study.

## 6.3 Comparatives

Another problem is one that was already mentioned briefly in the results section and which has to do with the way Boxer treats comparatives. Take for instance example 69 which shows the DRS of a sentence in which the VPE stands in a comparative relation to its antecedent.

**John runs faster than Bill does.**

(69)



Resolving the VPE in this sentence should result in (70a), but when extracting the antecedent from the DRT, (70b) is obtained instead.

(70) a. John runs faster than Bill runs.
b. John runs faster than Bill runs fast.

The reason for this is that according to the DRT, the clause headed by "than" is related to the event with the verb "run". In reality, it modifies the comparative adverb "fast", but in DRT that is not possible since adverbs themselves can't be modified, only discourse referents. This means that there is no way to know that the adverb is part of the comparative containing the VPE and that it shouldn't be copied to the VPE location.

This type of error can lower the f-score considerably and cases of VPE that stand in such a comparative relation to their antecedent are quite frequent, so it probably has quite some impact on the overall results. A study on annotating VPE in the WSJ corpus by Bos and Spenader (2009) showed that comparatives are the most frequent form of VPE. The study also showed that in about 15% of all VPE occurrences, the VPE clause modifies its antecedent in a comparative relation as in example 69, causing the problem described above.

## 6.4 Noun similarity measure

Two features, agent similarity and parallel PPs, use WordNet to determine semantic similarity. This measure for semantic similarity, path similarity, is a very crude one and much better methods also based on WordNet are available (Pedersen et al., 2004). These methods make use of the fact that the lower a concept is located in WordNet's hierarchy, the more specific it is, and the higher, the more general it is. From this follows that concepts that are close together at the top of the hierarchy are less related than concepts that are close together further down. For instance, "dalmatian" and "poodle" is a noun pair that is quite low in the hierarchy and that is only two steps apart. They are both types of dogs, which makes them very semantically related. Another noun pair which is only two steps apart, but which is located near the top of the hierarchy, is "destruction" and "ornamentation", which are both a type of "state". These two terms aren't very semantically related, but still they are just as close together as "dalmatian" and "poodle".

Some methods determine semantic similarity between concepts using the information content[3] of the concepts and of their least common subsumer[4]

[3]Specificity of a concept; concepts low in WordNet's hierarchy are more specific than those higher up.

[4]The lowest concept in the hierarchy that subsumes both compared concepts, e.g. the

(Resnik, 1995; Lin, 1998; Jiang and Conrath, 1997). Other methods use various path distance and hierarchy depth measures to do the same (Leacock and Chodorow, 1998; Wu and Palmer, 1994).

## 6.5 Parallel Prepositional Phrases

The performance of agent similarity feature might improve considerably if the simple path similarity measure would be substituted by any of these alternative measures. The parallel PP feature suffers from another difficulty however. When looking at examples of VPE sentences from the data in which both the VPE and one of its possible antecedents contain at least one PP, it turns out that these PPs are almost never parallel to each other. I estimate that in the data set there are less than ten examples in which actual parallel PPs occur. So, combinations of PPs that are found in the data are far more often non-parallel than parallel. As a consequence, when training a GA on the data, antecedents which contain a non-parallel PP are favoured over antecedents in which the PP is considered parallel. The amount of examples that do contain parallel PPs is too small to automatically learn to distinguish them, and because 10-fold cross-validation is used for testing, the test data also contains too few examples. Therefore this feature doesn't have any effect on the results.

To make it work, parallel PPs will have to be made identifiable. To realize this, more positive examples will have to be added to the data and a better semantic similarity measure will have to be used to be able to distinguish them from the non-parallel examples.

## 6.6 Parallel adjuncts

Although this feature didn't improve our results, the fact remains that parallel PPs (and parallel adjuncts in general) will have to be dealt with during VPE resolution. For a VPE sentence to be resolved correctly, parallel elements from the antecedent musn't be copied to the VPE site, and to accomplish this, these parallel elements will have to be identified first. Furthermore, many studies confirm the important role parallelism plays in ellipsis and ellipsis resolution (Dalrymple, Shieber, and Pereira, 1991; Prüst, Scha, and van den Berg, 1991; Asher, 1993; Hobbs and Kehler, 1997; Asher, Hardt and Busquets, 2001). Therefore, considering parallel adjuncts isn't only necessary during the resolution step, but they might also give important clues for selecting the correct antecedent. Take for instance this sentence:

---

least common subsumer for "poodle" and "dalmatian" is "dog"

(71) a. John [$_{ANT1}$ admiringly [$_{ANT2}$ listened while the band [$_{ANT3}$ played]]].
b. Bill [$_{VPE}$ did disgustedly].

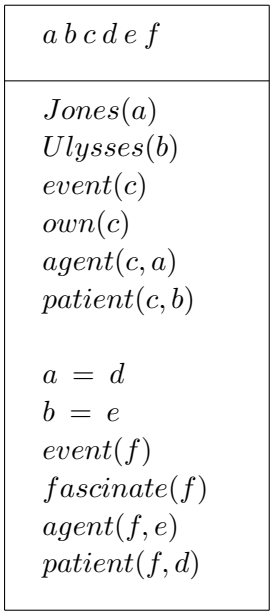
If parallel adjuncts would only be handled at the resolution step, the choice in possible antecedents would be between *ANT1* and *ANT3* and the correct antecedent VP headed by "listened" might not have preference over the one headed by "played". With our method of creating separate antecedents for each possible interpretation with respect to parallelism during antecedent selection, the fact that *ANT2* has a parallel element could be uncovered earlier, resulting in selection of the correct antecedent.

## 6.7 Boxer, DRT and semantics

The aim of this research was to use semantic information for selecting antecedents in VPE, and for this reason the semantic parser Boxer was used to extract this information. But although Boxer parses discourse into a semantic representation, it doesn't really add any semantic information to the data. The new data that Boxer's output does provide, modality, polarity and tense match, could also be extracted from syntax quite easily and, except for modality, isn't very semantic. The only real semantic information used for antecedent selection didn't come from Boxer, but from WordNet. A welcome addition to the capabilities of Boxer would be pronoun resolution, as demonstrated in example 72, because first of all, information about whether agents and/or patients in the VPE and the antecedent match (see example 34) might be very important for antecedent selection and second, DRT is very suitable for representing this.

**John owns Ulysses. It fascinates him.**

(72)




The question arises whether maybe Boxer should just be left out of the loop and VPE resolution should be performed on just syntactic information, because that's all Boxer's output is really based upon. And of course semantic information from an external source like WordNet could then be added to such a syntax based method, in much the same way it was added in the current research. In the case of antecedent selection the answer should probably be yes. But when looking at the total process of VPE resolution, I believe Boxer and DRT do have advantages over syntactic methods like Nielsen (2005). As argued in section 3.3.3, the abstracted nature of DRT makes the VPE resolution step much easier, because the syntactic variability that otherwise would have to be handled has been eliminated and therefore only a small amount of simple resolution rules are necessary. For instance, sentences (73a) and (73b) contain the same semantic information, but one is in passive form and the other in active form. A syntactic method will need two different rules for resolving the VPE in both sentences, while in DRT, both sentences will have the same representation and both will be resolved in the same way.

(73) a. That contract had to be signed by everyone, so Bill did too.
b. Everyone had to sign that contract, so Bill did too.

I also believe that when looking at possible NLP applications in which VPE resolution would be implemented, like for example a dialogue system, producing syntactically correct, resolved sentences, like Nielsen's system does, won't be very useful. The system would have to know what the meaning of

the discourse provided by the user is in order to provide a suitable response. A well-formed, resolved version of the user's input would have no use here. In DRT, all sorts of resolution and inference steps can be performed on discourse while keeping a clear, logical representation and not having to worry about syntax, which will only come in play at the initial creation of the DRS.

# 7 Conclusion

In this thesis, a new method of antecedent selection for VP ellipsis, using semantic antecedent features, is described. Results were somewhat disappointing because none of the features that were implemented could improve the results of the recency feature alone. However, this does not mean that these features are useless for antecedent selection. In the discussion chapter a number of other reasons for the low results are described. If those problems were solved, the implemented features would probably show better performance.

For future research, I recommend a number of changes to the current study. First, to get good results from a Genetic Algorithm or another ML algorithm, a large enough data set is required. The amount of VPE examples used in this study was far too small, so more corpus data should be used. This should especially increase performance of infrequent features like the Parallel PP's feature.

I further recommend using syntactically annotated data. Although a real world NLP application would have to incorporate a syntactic parser, that bit of realism just introduced errors into our results, hiding the real performance of the antecedent features, which is what we actually want to test. Right now it is hard to determine which part of the results is due to the parsing errors and which is due to the performance of the features. Boxer could still be used, parsing annotated data into DRT instead of syntactically parsed data.

To really make the *agent similarity* and *parallel PPs* features work, a better noun similarity measure is required. Compared to other features implemented in this study, *agent similarity* performs reasonably well, and with a similarity measure that is very simple. It is very probable that this feature will work very well if the similarity measure is improved. A number of more advanced measures based on WordNet has been mentioned, but of course other methods could be used as well.

In addition to parallel PPs, parallelism of other adjuncts should be studied too. This is fairly simple to implement in a similar way to parallel PPs. If both the VPE and a candidate antecedent contain an adjunct of the same type, e.g. an adverb, then these adjuncts should be compared semantically. If they are very similar, there is a high chance that they are also parallel, increasing the preference for the corresponding antecedent.

The problem that has to do with the way Boxer represents comparatives in DRT is a bit more difficult. Im not sure if this representation of comparatives is inherent to DRT, or if it just how Boxer treats them. Either way, if comparative cases if VPE are to be resolved properly by Boxer, this problem

should be dealt with.

## References


[1] Asher, N. (1993) *Reference to Abstract Objects in English.* Dordrecht.

[2] Asher, N., Hardt, D. and Busquets, J. (2001) *Discoure parallelism, ellipsis, and ambiguity.* Journal of Semantics, 18(1).

[3] Bos, J. (2005) *Towards wide-coverage semantic interpretation.* In Proceedings of IWCS-6, pp. 42-53, Tilburg, The Netherlands.

[4] Bos, J. (2007) *Robust VP Ellipsis Resolution in DR Theory.* ms.

[5] Bos, J., Spenader, J. (2009) *An annotated corpus for VP ellipsis.* ms.

[6] Brill, E. (1993) *Automatic grammar induction and parsing free text: A transformation-based approach.* In Proceedings of ACL

[7] Clark, S. and Curran, J. R. (2004) *Parsing the WSJ using CCG and Log-Linear Models.* In: Proceedings of the ACL.

[8] Cohen, W. W. and Singer, Y. (1999) *A simple, fast and effective rule learner.* In: Proceedings of the 16th National Conference on AI.

[9] Dalrymple, M., Shieber, S. and Pereira, F. (1991) *Ellipsis and higher-order unification.* Linguistics and Philosophy, 14(4).

[10] Francis, W. N. and Kucera, H. (1982) *Frequency analysis of English usage: Lexicon and grammar.* Houghton Mifflin, Boston.

[11] Hardt, D. (1992) *An Algorithm for VP Ellipsis.* In: Proceedings of the 30th Annual Meeting of the Association for Computational Linguisitics.

[12] Hardt, D. (1997) *An empirical approach to vp ellipsis.* Computational Linguistics, 23(4).

[13] Hobbs, J. and Kehler, A. (1997) *A Theory of Parallelism and the Case of VP ellipsis.* In: Proceedings of the 35th Annual Meeting of the Association for Computational Linguisitics.

[14] Hockenmaier, J. (2003) *Data and models for statistical parsing with Combinatory Categorial Grammar.* Ph.D. thesis, University of Edinburgh.

[15] Jaynes, E. T. (1957) *Information theory and statistical mechanics.* Physical Review, 106, pp. 620-630.

[16] Jiang, J. and Conrath, D. (1997) *Semantic similarity based on corpus statistics and lexical taxonomy.* In: Proceedings on International Conference on Research in Computational Linguistics, pp. 1933.

[17] Kamp, H. and Reyle, U. (1993) *From Discourse to Logic: An Introduction to Modeltheoretic Semantics of Natureal Language, Formal Logic and DRT.* Kluwer, Dordrecht.

[18] C. Leacock and M. Chodorow. 1998. Combining local context and WordNet similarity for word sense identification. In: C. Fellbaum (ed.), *WordNet: An electronic lexical database*, pp. 265283, MIT Press.

[19] Leech, G. (1992) *100 million words of english : The British National Corpus.* Language Research, 28(1), pp. 1-13.

[20] Lin, D. (1998) *An information-theoretic definition of similarity.* In: Proceedings of the International Conference on Machine Learning.

[21] Marcus, M. P., Santorini, B. and Marcinkiewitcz, M. A. (1993) *Building a large annotated corpus of English: The Penn Treebank.* Computational Linguistics, 19(2).

[22] Miller, G. A. (1995) *WordNet: a lexical database for English.* Communications of the ACM, 38(11), pp. 39-41.

[23] Nielsen, L. A. (2005) *A corpus-based study of Verb Phrase Ellipsis Identification and Resolution.* Ph.D. diss., King's College London.

[24] Pedersen, T., Patwardhan, S. and Michelizzi, J. (2004) *Wordnet::similarity -measuring the relatedness of concepts.* Proceedings of the Nineteenth National Conference on Artificial Intelligence.

[25] Prüst, H., Scha, R. and Van den Berg, M. (1991) *A discourse perspective on verb phrase anaphora.* Linguistics and Philosophy, 17(3), pp. 261-327.

[26] Quinlan, R. (1993) *C4.5: Programs for Machine Learning.* San Mateo, CA: Morgan Kaufmann.

[27] Resnik, P. (1995) *Using information content to evaluate semantic similarity in a taxonomy.* In: Proceedings of the 14th International Joint Conference on Artificial Intelligence, pp. 448453.

[28] Stanfill, C. and Waltz, D. (1986) *Toward memory-based reasoning.* Communications of the ACM, pp. 1213-1228.

[29] Steedman, M. (2001) *The syntactic process.* The MIT Press.

[30] Wu, Z. and Palmer, M. (1994) *Verb semantics and lexical selection.* In: 32nd Annual Meeting of the Association for Computational Linguistics, pp. 133138.