IDENTIFYING REDUCED PASSIVE VOICE CONSTRUCTIONS IN SHALLOW PARSING ENVIRONMENTS

by

Sean Paul Igo

A thesis submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Master of Science

in

Computer Science

School of Computing

The University of Utah

December 2007

THE UNIVERSITY OF UTAH GRADUATE SCHOOL

SUPERVISORY COMMITTEE APPROVAL

of a thesis submitted by

Sean Paul Igo

This thesis has been read by each member of the following supervisory committee and by majority vote has been found to be satisfactory.

Chair: Ellen Riloff

Robert Kessler

Ed Rubin

THE UNIVERSITY OF UTAH GRADUATE SCHOOL

FINAL READING APPROVAL

To the Graduate Council of the University of Utah:

I have read the thesis of Sean Paul Igo in its final form and have found that (1) its format, citations, and bibliographic style are consistent and acceptable; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the Supervisory Committee and is ready for submission to The Graduate School.

Date	Ellen Riloff Chair, Supervisory Committee

Approved for the Major Department

Martin Berzins
Chair/Dean

Approved for the Graduate Council

David S. Chapman
Dean of The Graduate School

ABSTRACT

This research is motivated by the observation that passive voice verbs are often mislabeled by NLP systems as being in the active voice when the “to be” auxiliary is missing (e.g., *“The man arrested had a...”*). These errors can directly impact thematic role recognition and applications that depend on it. This thesis describes a learned classifier that can accurately recognize these “reduced passive” voice constructions using features that only depend on a shallow parser. Using a variety of lexical, syntactic, semantic, transitivity, and thematic role features, decision tree and SVM classifiers achieve good recall with relatively high precision on three different corpora. Ablation tests show that the lexical, part-of-speech, and transitivity features had the greatest impact.

To Julia James

CONTENTS

CHAPTERS

APPENDICES

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I owe a great debt to my advisor, Ellen Riloff, for her patience and expertise. Likewise to the Natural Language Processing research group for many useful suggestions. I would also like to thank Ed Rubin from the University of Utah Linguistics Department and my wife, Julia James, for guidance from the theoretical linguist's point of view. Thanks as well to Robert Kessler, for having served on my committee during two separate attempts at graduate school.

I would also like to express appreciation to my father, Dr. Robert Igo, and to Dottie Alt, for encouraging me to complete my degree.

Last but not least, I would like to thank my momma and Elvis.

CHAPTER 1

OVERVIEW

An important part of natural language processing (NLP) is the association of noun phrases with verbs. Noun phrases can play different *thematic roles* with respect to the verb. Two thematic roles of special interest are *agent*, performer of the action described by the verb, and *theme*, that which is acted upon. Correct assignment of noun phrases to their thematic roles is crucial to understanding a sentence properly.

Thematic roles exist in all human languages, and recognition of them depends on syntactic and morphological properties of the language. The scope of this research is limited to English; in particular to text and not speech.

In English, agents frequently appear as subjects of sentences and themes occupy the direct object position. This is common enough that the term *subject* is sometimes used incorrectly as a synonym for *agent*. However, verbs in passive voice can reverse the positions of agent and theme with respect to the verb. For instance, these two sentences have the same meaning:

```
1.  Sam washed the dog.
2.  The dog was washed by Sam.
```

In sentence 1 *washed* is in active voice while in sentence 2 it is in passive voice. *Sam* moves from subject position to a prepositional phrase (PP) headed by *by*, but is the agent in both cases.[1] Similarly, *the dog* moves from direct object position to subject position, but remains the theme. In this way passive voice affects the relationship of syntactic positions to thematic roles. It also allows agents to be omitted entirely. For example:

```
3.  The dog was washed.
```

[1] Of course, not every NP in subject position with respect to a passive-voice verb is a theme, nor is every NP occurring after one in a PP headed by *by* an agent. For instance, *apple trees* does not behave as an agent in the sentence “The castle was surrounded by apple trees.” However, these syntactic relationships are a useful generalization.

Because of these effects, it is important that NLP systems concerned with thematic role recognition – and meaning more generally – be able to determine that a given verb is in passive voice. For convenience I will call verbs in passive voice *passive verbs*, and those in active voice *active verbs*. It is possible for non-verb words to have passive voice properties, but my research is limited to verbs.

Passive verbs can cause distinctive patterns in phrase structure; I call these patterns *passive constructions*. Sentences 2 and 3 show the most common passive construction, in which the verb is a past participle and has an auxiliary, a form of the verb *to be*. Since it is so common, I call this construction an *ordinary passive*.

Section 1.1 below will describe different passive constructions that are difficult to recognize and form the focus of my research. Section 1.2.1 will explain these constructions' impact on NLP applications.

1.1 Reduced passive voice constructions

The following sentence shows an ordinary passive voice construction in a relative clause (underlined):

4. `The dog that was washed yesterday lives next door.`

Though the structure of this sentence is more complex than that of sentence 2, it is still easy to identify the passive verb because *washed* is a past participle and is preceded by the auxiliary *was*. Furthermore, *the dog* is clearly the theme, as it was in sentence 2. However, it is fairly common for writers to shorten a sentence like this by using a reduced relative clause. In this case, the result is this:

5. `The dog washed yesterday lives next door.`

The reduced relative clause hides not only the relative pronoun (*that*) but the passive auxiliary *was*. Its hidden elements are readily understood by human English readers but present a challenge to many NLP systems, which may incorrectly parse this as an active voice construction and consider *the dog* to be the agent rather than the theme.

Passives without auxiliaries can occur in other forms as well, such as the verb *stung* in sentence 6:

6. `Stung by 1000 bees, Maynard cursed furiously.`

The syntactic position of the theme of *stung* (*Maynard*) is different from that of the theme in sentence 2, but the agent is once again in a following *by*-PP and there is no passive auxiliary.

I will call passive verbs with no passive auxiliary *reduced passives*.

1.2 Motivation

Given sentence 5, an NLP system that relies solely on the *be*-auxiliary to identify passive verbs will consider *washed* to be an active verb. This could lead it to consider *the dog* to be the agent and *yesterday* the theme, which is incorrect. A more sophisticated syntactic analysis of the sentence might identify the reduced relative clause and allow for recognition of the passive verb without the auxiliary, but a significant number of NLP systems use a simple approach to syntax, called *shallow parsing*, which can overlook complexities like this. I will discuss the strengths and weaknesses of shallow parsing, and alternatives to it, in Chapter 2.

The next section presents an NLP application that is affected by passive voice recognition.

1.2.1 Example of reduced passives' effects

Many NLP systems are sensitive to syntactic positions of constituents in carrying out their tasks. Information extraction (IE) systems that use lexico-syntactic patterns (e.g. [32, 37]) are one example. IE systems build descriptions of certain types of events; each single event's description consists of a *template* with event roles, or *slots*, to be filled by attributes of the event. These slots resemble semantic or thematic roles, and depend on identification of thematic roles, but participate in an event description that may involve several verbs and their arguments. A template describing a kidnapping, for example, might have slots for the perpetrator, the victim, the location where the victim was abducted, and so on. IE systems often use a shallow parser to find noun phrases (NPs), verb phrases (VPs), and other syntactic features.

AutoSlog [31] is an IE pattern learner of this type. AutoSlog uses the Sundance shallow parser to label verb phrases as active or passive voice and determine syntactic roles, such as subject and direct object, for the noun phrases. AutoSlog then creates simple lexico-syntactic extraction patterns, called *caseframes*, to identify and retrieve information relevant to an event. For instance,

```
Event Type:  VEHICLE CRASH
Activation:  active verb CRASHED
Slot:  Vehicle (AIRCRAFT, AUTO):  subject
```

This caseframe recognizes vehicle crash incidents described in text by finding a verb phrase headed by the verb *crashed* in active voice. Given that, if the subject also fulfills the semantic constraint of being an aircraft or automobile, the system knows that the subject of the verb is the vehicle that crashed and fills the Vehicle slot with it. The semantic constraint depends on a dictionary, which must associate a given word with the AIRCRAFT or AUTO semantic type for it to qualify for this Vehicle extraction. The semantic constraint mechanism helps to limit extractions to sensible answers. For example, in the sentence below:

`The police crashed into the apartment.`

the police will not be extracted as a vehicle. A sentence like this one:

`The truck crashed into the tree.`

where *truck* is listed as an AUTO in the dictionary, would satisfy the semantic constraints and the Vehicle slot would be filled by *the truck*.

The following caseframes form a complementary pair for recognizing the victim and perpetrator of a kidnapping event, depending on whether the verb is in active or passive voice:

```
Event Type:  KIDNAPPING
Activation:  active verb KIDNAPPED
Slot:  Victim (HUMAN):  direct object
Slot:  Perpetrator (HUMAN):  subject

Event Type:  KIDNAPPING
Activation:  passive verb KIDNAPPED
Slot:  Victim (HUMAN):  subject
Slot:  Perpetrator (HUMAN):  Prepositional Phrase (BY)
```

These caseframes fill the slots from different syntactic roles depending on the verb's voice. If the verb *kidnapped* is in active voice and the direct object NP is classified as HUMAN, then the Victim slot is filled from the direct object. If the verb *kidnapped* is in passive voice, and the NP in the subject position is semantically classified as HUMAN, then the Victim slot is filled with the subject NP. Similarly, the Perpetrator will be found in the subject position in the active voice case, and in a PP headed by *by* in the passive voice case. As an illustration, this sentence:

`The terrorists kidnapped the workers.`

would fire the active voice caseframe and fill the perpetrator slot with *the terrorists* and the victim slot with *the workers.* The passive voice sentence:

`The workers were kidnapped by the terrorists.`

would fire the passive voice caseframe and arrive at the same slot assignments.

Another way of looking at this is that the Victim slot will be filled by the theme of *kidnapped* and the Perpetrator slot will be filled by the agent. Since the verb *kidnapped* has the property that both its agent and theme can be in the same semantic category (i.e., HUMAN), caseframes based on it are vulnerable to slot-assignment errors if the voice is not correctly recognized. For example, if the reduced passive verb *kidnapped* in this sentence:

`The workers kidnapped yesterday were released today.`

were to be misclassified as active voice, the active-voice caseframe would fill the perpetrator slot with *the workers*, when in fact they were the victims.

Caseframes that extract agents and themes with differing semantic roles can also be affected when a reduced passive verb is mistaken for an active voice verb. In such cases, the incorrectly-recognized agent and theme would fail the semantic constraints for the active voice caseframe and no extraction would be performed. Thus, useful data would be overlooked.

Thematic role recognition is a component of many other NLP tasks, including question answering (e.g., [33, 35]), opinion analysis (e.g., [3, 8, 15]), and summarization (e.g., [19]). Semantic role labeling (SRL), which is essentially thematic role recognition, has received a great deal of attention in recent years as a challenging task in its own right. Many SRL systems use an active/passive voice feature to assist in the semantic role labeling task (e.g., [10, 26, 12, 39]). Reduced passives are a potential problem for all these tasks, and they are common. Among the three corpora that I used for my experiments, reduced passives occurred in roughly 1/12, 1/9, and 1/8 of sentences, for an average of about one reduced passive per 10 sentences. The corpora and experiments will be described in detail in Chapters 2 and 3.

1.3 Thesis outline

Having established that reduced passives are common and a potentially serious problem for NLP systems that need to identify thematic roles, I decided to focus my thesis research on reduced-passive recognition. My hypothesis is that it is possible to recognize reduced passive verbs using a learned classifier trained with features derived from syntactic parsing and some supplementary semantic and verb usage data. The remaining chapters describe related work and then my own research.

Chapter 2 discusses reduced-passive recognition in terms of syntactic parsing. A parser whose grammar specifies phrase structures for reduced passives could, in theory, be used to identify them. This chapter describes several existing parsers and the results of evaluating them for reduced-passive recognition over three disparate text corpora.

Chapter 3 presents my research hypothesis, that reduced-passive recognition may be approached as a classification problem using a supervised learning approach. I discuss features that describe verbs in terms of syntactic, semantic, lexical, and other properties extracted from a shallow parse. I also present some transitivity and thematic role features, designed to capture properties of passive-voice verbs unavailable in a shallow parse, which require separate training.

Chapter 4 details my experimental design and shows the results for reduced passive classifiers using different learning algorithms and feature vectors. Experiments demonstrate that the classifier can be competitive with existing parsers.

Finally, Chapter 5 presents my conclusions about the classification approach and discusses possibilities for future improvements.

CHAPTER 2

PRIOR WORK

The purpose of this chapter is to discuss existing NLP programs' effectiveness in recognizing passive voice constructions. I have not encountered any research dealing specifically with reduced passive recognition, though some publications mention the issue in the context of broader topics such as syntactic annotation [20, 4] and information extraction [13]. Since no system exists purely for reduced passive recognition, my prior-work experiments consisted of testing programs that can *almost* recognize reduced passives: parsers that produce specific phrase structures that correspond to reduced passives.

The existing NLP programs for this task are all syntactic parsers. These take plain text as input and produce a representation of each input sentence's phrase structure according to a grammar that is either created by human experts or learned by analysis of training texts. Some parsers explicitly label verb voice, though none that do so recognize reduced passives. It is possible to recognize reduced passives through phrase structure alone, if the structure is based on a grammar that has constructions specifically for reduced passives. Some widely-available parsers use grammars that include reduced-passive constructions. I found three such parsers and created postprocessing software for them. I then used the programs and postprocessors to identify reduced passive verbs in three text corpora and evaluated their accuracy by comparing their judgments to hand-annotated versions of the same corpora. Overall, they were moderately to very successful.

However, not all parsers support reduced-passive recognition this way. Many *shallow* parsers use simple grammars that do not assign specific structures to reduced passive constructions, but these parsers have advantages over the more sophisticated parsers which make them worthy of consideration. In particular, they are faster and therefore more suitable for large-volume text processing tasks. They are also more robust when given fragmentary or ungrammatical input. Because of these properties, they are often used as components of larger systems, such as information extraction systems. Most IE systems use shallow parsers, and Chapter 1 established the need for accurate verb voice determination in IE systems because some of them rely on thematic role identification. Accordingly, my research concerns recognizing reduced passives using shallow parsers. In this chapter, I describe four such parsers, none of which supports reduced passive recognition.

Section 2.1 discusses the two categories of parsers mentioned above in further detail. Section 2.2 describes some properties of text corpora and the impact that they can have on parsers' performance as either training texts or new text. Section 2.3 describes how reduced passive constructions can be detected in the Penn Treebank corpus [18], showing how reduced passive recognition can be effected as a postprocessing of syntactically annotated text or Treebank-style parser output.

Section 2.4 provides details of the text corpora I used to evaluate parsers' reduced passive recognition, and Section 2.5 describes the experiments I performed. Finally, Sections 2.6 and 2.7 present the results of my experiments with existing parsers in each of the two main classes.

2.1 Parsing

The goal of parsing is to represent the structure of input text according to a syntactic theory. If the theory assigns an unambiguous structure to reduced passive constructions, a parser based on it will enable reliable identification of them. The structure found by parsers consists of *constituents*, or subgroupings of the tokens in a sentence, and some representation of the relationships between them. These representations vary widely.

For the purposes of this research, I classify parsers into two broad groups: *full* and *shallow* parsers. Full parsers attempt to discover a single structure which incorporates every word in the sentence. In most cases, this structure is hierarchical and corresponds to phrase structure; in these instances the representation is called a *parse tree*. Other parsers, such as *dependency* parsers, are concerned with constituent dependency relationships but otherwise perform the same kind of analysis. Because of their detailed grammars, full parsers' output is syntactically rich.

By contrast, shallow parsers do not necessarily assign every word in the sentence to a higher-level constituent. They identify simple phrases but do not require that the sentence be represented by a single structure. Often their phrase structure representation is a "flat," or sequential, set of major constituents such as noun phrases, verb phrases, and prepositional phrases, though there may be some limited hierarchy among them.

Full and shallow parsers are suitable for different kinds of NLP tasks. There are three main disadvantages to full parsers. First, their sophistication requires significant processing power and execution time, possibly more than is comfortably available for large-volume NLP tasks. Second, faced with a fragmentary or ungrammatical sentence, a full parser may generate no answer at all. Third, learned full parsers may require heavily annotated training data that is expensive to produce. In contrast, shallow parsers are typically faster than full parsers and more robust when given fragmentary or ungrammatical input. Training data for learned shallow parsers are simpler to generate, and some effective hand-crafted shallow parsers exist that do not require training at all. However, shallow parsers' elementary view of syntax may overlook important structural properties of a sentence.

Parsers of both types tend to perform better or worse depending on the degree to which input text conforms to the grammar guiding them. Lexicalized parsers of either type also depend on particular word usages as grammatical clues. Grammar and word usage are sometimes specialized; in the next section I examine the impact this has on parsing and discuss one widely-used corpus.

2.2 Corpora and domains

Many text corpora consist of documents concerning a narrow range of topics (*domain*) or which have a prescribed writing style (*genre*). The domain and genre can have a strong effect on grammar, word senses, and other linguistic properties. Learning-based NLP systems that are trained on a single domain have biases related to that domain. This can be useful if the system is intended to process documents only in that same domain, but it is a disadvantage otherwise. Passive voice usage is one property that varies with domain - certainly in frequency, as Section 2.4 shows, and possibly in other ways.

The Penn Treebank [18] is a widely-used corpus which consists of 2,313 *Wall Street Journal* articles that have been manually annotated with the parts of speech for individual words and a parse tree for each sentence. Thus, the Treebank contains the *correct* parse information for these 2,313 articles, and is therefore used as a gold standard for evaluating parsers. The syntactic theory it embodies is similar to Government and Binding Theory [11], with modifications and simplification of structure to accommodate the practical difficulties of creating a large corpus with limited human resources.

The Treebank is used as a training corpus for many parsers, which means that they may be biased toward the particular grammar used by *Wall Street Journal* reporters. Further, some of these parsers depend on specific words as syntactic cues, so those parsers might also be affected by specialized word usages in the domain.

Treebank annotation does not explicitly label verb voice, but the Treebank guidelines [18, 4] identify syntactic structures that generally correspond to both ordinary and reduced passives. These are clear enough that I was able to write a simple program, which I called "Entmoot," to identify both types of passives in the Treebank data with a high degree of accuracy. To confirm that this program can identify passives reliably, I manually annotated some Treebank documents with ordinary and reduced passives. Then I compared Entmoot's output against the manually annotated labels. In the next section, I describe the Entmoot reduced-passive recognition process in more detail.

2.3 Entmoot - finding voice in Treebank trees

The Entmoot program reads Treebank-formatted parse trees and searches for past-participle verbs. When it finds one it applies six rules to see if the verb's context implies that the verb is in passive voice. The full set of rules is given in Appendix A. Here, I will describe two of the rules in detail, for illustration purposes.

The "Obvious Ordinary Passive" rule examines the verb's context to see if there is a passive auxiliary verb in a structurally appropriate position. That is, the auxiliary must precede the verb and be a sibling to a verb phrase (VP) containing the verb. In Treebank phrase structure terms, that is:

- verb's parent is a VP (the Treebank guidelines allow verbs to head other phrase types.)
- Starting with the parent and climbing nested VP ancestors, the closest verb sibling before any VP ancestor is a passive auxiliary (*be*, *get*, etc.).

For instance, in the phrase below, *permitted* is passive:

```
       (NP (NN smoking) )
       (VP
           (MD will)
gp:        (VP
               (VB be)
par:           (VP
                    (VBN permitted) <================
                        (PP-LOC (IN in)
                            (NP (NNS bars) ))
```

Its parent (par) is a VP, and its grandparent (gp) is also. The child of gp preceding par is a *be*-verb, so the rule applies.

The "Obvious Reduced Passive" rule looks for the structure that the Treebank annotation guide specifically reserves for reduced passives. That is, one type of reduced passive is a past-participle verb that is the head verb of a verb phrase (VP), which is, in turn, contained by a noun phrase (NP). In Treebank phrase structure terms, it looks like this:

- Node's parent is a VP; optionally, grandparent, great-grandparent, etc. are VPs, with no non-VP phrases intervening
- Parent of oldest VP ancestor is NP
- None of VP ancestors' preceding siblings is a verb – there are similar constructions with intervening verbs which aren't reduced passives.

Here, *appointed* is a reduced passive:

```
gp:   (NP
         (NP (DT a) (JJ special) (NN master) )
par:     (VP
             (VBN appointed) <==================
                 (PP (IN by)
                     (NP-LGS (DT the) (NNP Supreme) (NNP Court) ))))
```

For words that match one of the six rules that I identified, Entmoot assigns the appropriate ordinary or reduced passive label.

I scored Entmoot's voice labeling performance by comparing its verb voice assignments to those I made in a hand-annotated *Wall Street Journal* gold standard (described in the next section). For ordinary passives, Entmoot's recall is 98.37% and its precision is 98.48%, producing a 98.43% F-measure. For reduced passives, its recall is 92.99% and its precision is 92.78%, yielding an F-measure of 92.88%. From a manual inspection of the mistakes, I found that some of the errors are due to shortcomings of the recognition rules, some to annotation error, but most of the errors seem to be due to mistakes such as incorrect part of speech tags in the Treebank corpus. The next section describes my gold standard in detail.

2.4 Gold standard and evaluation

For parser evaluation purposes, I manually annotated three sets of documents with passive voice labels. My gold standards consist of three small corpora: 50 randomly chosen documents from the Penn Treebank, 200 from the MUC-4 terrorism corpus [23], and 100 from the ProMED mailing list [25]. I used different numbers of documents from each domain because the documents varied in size and reduced-passive density; in the end all three corpora contained roughly equal numbers of reduced passives: 442 in the WSJ gold standard, 416 in the MUC-4, and 464 in the ProMED.

These corpora represent three distinct domains: Treebank documents are *Wall Street Journal* articles. The MUC-4 corpus is a set of press releases, transcripts, and news articles related to terrorist activity in Central and South America, and the ProMED corpus consists of archived semiformal email postings about disease outbreaks. In each of these corpora, about 4-6% of all verbs are reduced passives, but their impact is potentially broader than their scarcity implies. One unrecognized reduced passive may affect thematic role assignments for an entire sentence, so it is useful to see how many sentences contain them. Table 2.1 shows that around 10% of sentences in the gold standards have reduced passives. The proportion is higher than reduced passives' raw verb percentage because they almost always occur in sentences that have multiple verb phrases.

This shows that passive verb usage differs by domain but that it is common enough – with over 1 in 10 sentences affected – to be a concern.

Table 2.1: Frequency of passive verbs.

Corpus	**Total sentences**	**Sentences containing ordinary passive**	**Sentences containing reduced passive**
Treebank	4965	818 (16.48%, about 1 in 6)	415 (8.36%, about 1 in 12)
MUC-4	3304	953 (28.84%, about 1 in 4)	383 (11.59%, about 1 in 9)
ProMED	3030	955 (31.52%, about 1 in 3)	402 (13.27%, about 1 in 8)
Combined	11299	2726 (24.13%, about 1 in 4)	1200 (10.62%, about 1 in 10)

2.5 Assessing the state of the art

Having established that reduced passive constructions are common, I conducted the experiments described in this section in order to evaluate how well verb voice, including reduced passives, can be recognized by state-of-the-art parsers. A few parsers label verb voice explicitly, but most do not, so a postprocessing step is necessary to recover them. Figure 2.1 shows how my evaluation works: I created a postprocessing program for each parser which scans the parser's output and determines verb voice from whatever clues are available. The level of postprocessing needed should be minimal; that is, it should not require any data other than the parser's output nor any serious algorithm development effort. Entmoot serves as this postprocessor for parsers that output Treebank-formatted data, and I used other simpler programs for two other parsers.

The parser and postprocessor together effectively yield a passive-voice recognition system: a pipeline whose output is the original input text with passive verbs automatically labeled. Figure 2.1 shows this pipeline. For each parser, I performed these steps and compared the passive-annotated text to the hand-annotated gold standard to generate a score for the parser's effectiveness in verb voice identification.

Each parser's score is expressed in terms of recall and precision for ordinary and reduced passives.

2.6 Full parsers

To assess the state of the art in full parsing, I evaluated three well-known and widely-used parsers that are freely available on the web: MINIPAR, Collins, and Charniak. I will describe each below.

2.6.1 MINIPAR: Full parser with handwritten grammar

Dekang Lin's MINIPAR [16, 17] is a full parser based on minimalist syntactic theory. It uses a handwritten grammar and a large lexicon which is mechanically derived from WordNet [22] and augmented with proper names. During the course of parsing a sentence,

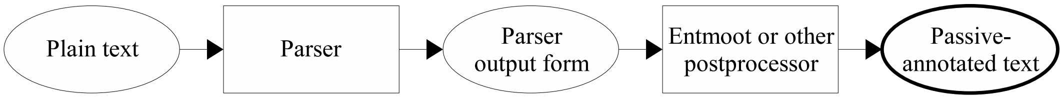

Figure 2.1: Parser evaluation process.

MINIPAR generates all the phrase structures possible for the sentence according to the grammar. It converts these to a set of dependency relations, which describe the head-modifier relationships among all the words in the sentence. Its output is the dependency tree corresponding to the most likely parse. Parse likelihood is evaluated using a statistical model derived from parsing a large unannotated corpus with MINIPAR and collecting probabilities of dependency relationships among particular words.

One of the head-modifier relationships output by MINIPAR is "vrel," or "passive verb modifier of nouns" which corresponds to a reduced-relative-clause type reduced passive.[1] Given a sentence with a reduced passive, such as this:

```
The dog washed yesterday was a border collie.
```

MINIPAR reports a "vrel" relationship between *washed* and *dog*, meaning that *washed* is a passive verb modifier of the noun *dog*. I wrote a program to convert MINIPAR's output to annotated text, appending a reduced-passive label to any word marked "vrel"; no attempt was made to label ordinary passives or other cases of reduced passives because there did not seem to be a similarly straightforward way to do so.

Table 2.2 shows MINIPAR's reduced-passive recognition performance, scored against the three gold standards described in Section 2.4: 50 *Wall Street Journal* articles (WSJ), 200 MUC-4 documents (MUC-4), and 100 ProMED mail documents (PRO). Scores given are recall (R), precision (P), and F-measure (F). MINIPAR recognizes 44%–51% of reduced passives; these low recall scores suggest that either MINIPAR is overly conservative in identifying "vrel" relationships, or that other relationships in addition to "vrel" are needed to recognize reduced passives more thoroughly. However, I did not find any other relationships that appeared to denote reduced passives, so a deeper analysis of the constituents' dependencies would be necessary. MINIPAR's precision scores, ranging from 57%–72%, show that it is moderately accurate with the "vrel" predictions it does make.

[1] MINIPAR's documentation does not explicitly state that "vrel" never has a passive auxiliary. From inspection of sentences which included the relation, I never observed any with an auxiliary nor any which seemed to be acting as anything but reduced passives or error cases in which the word labeled "vrel" was not even a verb.

Table 2.2: MINIPAR's performance for recognizing reduced passives.

Parser	Type of	WSJ			MUC-4			PRO		
	Passives	R	P	F	R	P	F	R	P	F
MINIPAR	Reduced	.48	.57	.52	.51	.72	.60	.44	.68	.53

2.6.2 Treebank-trained full parsers

The Collins [9] and Charniak [6] parsers are full parsers that learn their grammar rules from Treebank data.

Collins' parser represents a parse tree as a series of decisions hinging on phrases' lexical heads. These decisions are grammatical relationships between pairs of head words such as head projection, subcategorization, complement/adjunct placement, dependency, distance, and *wh*-movement. The parser as distributed can use any of three statistical models: Model 1, the simplest, does not account for complement/adjunct distinctions, verb subcategorization, and wh-movement. Model 2 adds complement/adjunct distinctions and verb subcategorization, and Model 3 adds these and wh-movement.

The Collins parser is not guaranteed to find a parse tree for a given sentence. I used Model 1 exclusively for my experiments because it failed on fewer sentences than Models 2 or 3. The distribution I used was version 1.0, released December 15, 2002.

Charniak's parser first generates a set of possible parses using a bottom-up chart parser based on a probabilistic context-free grammar learned from the Penn Treebank [7]. It then evaluates these parses using a top-down generative model which assigns a probability by considering each constituent and hypothesizing its phrase type, its lexical head, and its expansion into further constituents.

Both of these parsers produce Treebank-style parse trees, in which both ordinary and reduced passives can be extracted. For scoring, I used Entmoot (see Section 2.3), to convert their Treebank-style output to passive-annotated text. Overall, these parsers do well at passive voice recognition, both ordinary and reduced. Table 2.3 shows the recall (R), precision (P), and F-measure (F) scores for both types of passive for each of the full parsers over the three corpora.

For ordinary passives, the Collins and Charniak parsers score very high on Treebank data and only slightly lower on the MUC-4 and ProMED corpora. This suggests that ordinary passive constructions are easily recognized across domains.

For reduced passives, both parsers perform well on the WSJ corpus, but the scores

Table 2.3: Performance of Treebank parsers on different types of passive voice constructions.

Parser	Type of Passives	WSJ			MUC-4			PRO		
		R	P	F	R	P	F	R	P	F
Charniak	Ordinary	.95	.96	.95	.91	.96	.93	.93	.96	.94
Collins	Ordinary	.95	.95	.95	.91	.94	.92	.93	.94	.93
Charniak	Reduced	.90	.88	.89	.77	.71	.74	.77	.78	.77
Collins	Reduced	.85	.89	.87	.66	.67	.66	.64	.78	.70

drop noticeably on the other domains. One reason for this is part-of-speech tagging errors. Entmoot only considers past-participle verbs when searching for passives, so if a past-participle verb is mistagged as a past-tense verb it is disqualified regardless of its containing structure. Charniak's parser performs its own POS tagging and, in the ProMED corpus, mistagged nearly one in six of the reduced passives. The Collins parser requires input text to be preprocessed by a separate POS tagger. I used the recommended Ratnaparkhi POS tagger [27] and it mistagged about a quarter of the reduced passives in the ProMED gold standard in this way. The good performance by the Ratnaparkhi tagger / Collins parser chain on the Treebank gold standard hints that these tools are capable of doing well, but that they each suffer (and the errors possibly compound) when they are applied out-of-domain. I suspect that tagging past participles properly in English is especially difficult in reduced passive cases; not only do many past participles look identical to the same verb's past tense form, but it is likely that taggers often disambiguate them by finding passive or perfect auxiliaries within the few words preceding the verb. Given that, where these auxiliaries are missing, the verb will often appear to be past tense – and past tense was the most common mistagging for both parsers. The next most common mistagging was to mislabel the reduced-passive verbs as adjectives.

Note also that these parsers may have had in their training sets some of the same documents that were in my WSJ gold standard, which could make their scores artificially high in that domain.

2.6.3 Full parser conclusion

Table 2.4 shows the scores for all three full parsers for both ordinary and reduced passive recognition. Comparing the scores for the Collins and Charniak parsers and MINIPAR, MINIPAR's scores were the lowest, implying that its "vrel" relationship is

Table 2.4: Performance of parsers on different types of passive voice constructions.

Parser	Type of	WSJ			MUC-4			PRO		
	Passives	R	P	F	R	P	F	R	P	F
Charniak	Ordinary	.95	.96	.95	.91	.96	.93	.93	.96	.94
Collins	Ordinary	.95	.95	.95	.91	.94	.92	.93	.94	.93
Charniak	Reduced	.90	.88	.89	.77	.71	.74	.77	.78	.77
Collins	Reduced	.85	.89	.87	.66	.67	.66	.64	.78	.70
MINIPAR	Reduced	.48	.57	.52	.51	.72	.60	.44	.68	.53

either inaccurately labeled or that MINIPAR has other ways of representing reduced passives that are harder to extract from its output data.

The Treebank parsers' performance demonstrates that these full parsers recognize reduced passives best in the domain on which they are trained. Their scores drop on other domains. To achieve similar levels of performance on other domains, they should be trained with data closely resembling the expected input data. Unfortunately, the Treebank parsers depend on heavily annotated training data, so retraining them for a new domain requires a large investment of effort by experts. This problem is common to all supervised-learning parsers.

2.7 Shallow parsers

Shallow parsers typically do not recognize or label verb voice at all, nor can voice be easily deduced from their output. Case studies of some shallow parsers follow, including the few that do label verb voice.

2.7.1 OpenNLP: Treebank-trained chunker

OpenNLP [2] is a Java-based NLP utility package consisting of several NLP modules, including a sentence splitter, a tokenizer, a part-of-speech tagger, and a shallow parser.

The shallow parser is really a *chunker* in the CoNLL 2000 mold [34]. The shared task description for CoNLL 2000 reads "Text chunking consists of dividing a text in syntactically correlated parts of words." Chunks are contiguous groups of tokens which correspond roughly to phrases, but there is no hierarchy – i.e., chunks cannot contain other chunks. Chunks are also non-overlapping; no word can belong to more than one chunk [34].

For instance, this sentence:

```
The dog was washed by Sam.
```

receives the following parse:

```
[NP The dog][VP was washed][PP by][NP Sam][O .]
```

where brackets delimit phrases, with the phrase type given after the left bracket. The phrase type "0" means the contained token, here a period, does not belong to any phrase. Note that the prepositional phrase (PP) does not contain the following NP.

The core of the OpenNLP chunker is a learned maximum entropy model written by Baldridge et al., based on the work of Adwait Ratnaparkhi [28, 29]. The documentation does not make clear what the training corpus was, though there are hints that it was the Penn Treebank.

The chunker takes parallel streams of tokens and POS tags as input. The output is a third parallel stream of labels. Each token is labeled as being the beginning of a chunk (B), a subsequent word in the chunk (I), or not part of a chunk (O). The chunker does not identify verbs' voice, nor does its output provide enough structure to deduce it easily. There are many ConLL-style chunkers and any that produce this type of output will have the same limitations.

2.7.2 CASS: Shallow parser with handwritten grammar

Steven Abney's CASS parser [1] is a well-known shallow parser with the ability to use custom-made grammars, including non-English grammars. CASS is distributed with a sophisticated hand-built English grammar. It is the combination of the CASS code and this grammar that I will refer to as "the CASS parser."

CASS is based on a cascaded finite-state transducer model. The guiding philosophy is "growing islands of certainty into larger and larger phrases" [1]. That is, it is designed to apply several small hand-written grammars successively to an input text, each concentrating on a small set of high-precision tasks. The output from one stage's grammar is used as input for the next. The first might recognize only very simple noun phrases and verb phrases; the second could assemble prepositional phrases, and so on. No recursion is allowed; an early stage cannot depend on constituents recognized by a later stage. I classify CASS as a shallow parser because, while it can assemble constituents into full sentences, it does not insist on that.

Output from CASS is configurable. The most informative style displays a hierarchical parse tree. It also provides annotation of syntactic roles and word or phrase attributes, such as *subject*, *object*, and *head*.

However, the CASS shallow parser as distributed does not identify verbs' voice directly, nor are there unambiguous structural clues to it.

2.7.3 FASTUS: Chunker with handwritten grammar

FASTUS [13] is a system similar to CASS in that it is based on the finite-state transducer model – it applies a series of processes to input data, each depending only on previous stages. It differs from CASS in that it is not simply a parser, but a full information extraction (IE) system. The first three of its five stages constitute a shallow parser tailored for IE.

After the first three stages have run, the input text has been grouped into contiguous, nonoverlapping, nonhierarchical segments much like the chunks output by the OpenNLP shallow parser. The FASTUS chunking is more sophisticated, including principles like attachment of appositives and certain prepositional phrases to adjacent noun phrases, yielding what the authors call *complex noun phrases.* Some noun phrases are labeled with semantic attributes relevant to the domain, such as whether the phrase is a location or a company name.

Verbs are grouped together with modals, auxiliaries, and other modifiers, so that "*manufactures*," "*will manufacture*," and so on up to complex groups like "*has announced plans to begin manufacturing*" are considered variations on a single meaning. Depending on their modifiers and auxiliaries, the verb groups are labeled with attributes, including voice. For example, "*was manufactured*" would be labeled passive.

The FASTUS documentation explicitly mentions the reduced-passive problem, noting that "Verbs are sometimes locally ambiguous between active and passive senses, as the verb *kidnapped* in the two sentences, '*Several men kidnapped the mayor today,*' '*Several men kidnapped yesterday were released today.*' " However, the system does not perform disambiguation, and such verbs are labeled as ambiguous between active and passive.

Starting with Stage 4, FASTUS begins to be an IE system. At this point the processing concentrates only on verb groups that signal important activities within the domain and noun phrases of important semantic types. For instance, when applied to extract information about corporate acquisitions, it might pay special attention to the verb "*to manufacture*" in order to find a description of a participating company. Input text is scanned for patterns of phrases with certain types of subject, verb, and object (S-V-O) triples, although here I believe the authors are conflating *subject* with *agent* and *object* with *theme*. Given a S-V-O triple of interest, the system generates extraction patterns, similar to AutoSlog's caseframes, for recognizing several different relationships of the noun phrases to the verb phrase, including passive and reduced passive constructions with the agent and theme displaced.

Thus, while FASTUS can sometimes recognize reduced passive constructions during information extraction, it does not do so at the parsing stage, and its recognition is limited to certain, user-defined verbs and noun phrases of interest for a particular IE task. It is not readily available for download, so I have not evaluated the postparse representation for effectiveness in identifying reduced passive constructions. Nor does the documentation mention the system's success with respect to that particular case.

2.7.4 Sundance: Shallow parser with heuristic grammar

The Sundance shallow parser [31] uses handwritten heuristics to build contiguous constituents similar to those produced by OpenNLP and FASTUS. Its grammar allows for some hierarchy, like the inclusion of a NP within a PP, and the grouping of phrasal constituents into clauses. It performs part-of-speech tagging based on a handwritten lexicon, resolving ambiguous tags at the same time as it performs syntactic parsing.

In addition to syntactic parsing, Sundance generates some annotations useful to downstream applications, such as labeling noun-phrase constituents with semantic classes; e.g. "Salt Lake City" might be labeled as a LOCATION. It also labels verb phrases with an attribute related to tense and voice, which can be active, passive, or infinitive. It recognizes passive auxiliaries other than *be*, such as *get*. It does not recognize reduced passives.

To evaluate its current performance in identifying ordinary passives, I wrote a program which used Sundance version 4.4 to find VPs with passive attributes and identify their head verbs as ordinary passives. Its scores are shown in Table 2.5.

The scores are roughly similar across domains but are noticeably better for MUC-4 and ProMED. This may be due to Sundance's ability to use a domain-specific lexicon in addition to its general lexicon; MUC-4 and ProMED each have a specialized lexicon. Overall, Sundance's performance on ordinary passives is comparable to other parsers'

Table 2.5: Sundance scores for ordinary passive recognition.

Parser	**Type of**	**WSJ**			**MUC-4**			**PRO**		
	Passives	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**
Sundance	Ordinary	.82	.87	.84	.87	.92	.89	.86	.94	.90

on the MUC-4 and ProMED corpora, but noticeably lower on Treebank. This is to be expected, given that the other parsers were trained on Treebank data.

2.8 Summary of prior work

To summarize, the Treebank-trained full parsers perform well at recognizing both ordinary and reduced passives in documents in the same domain as their training corpus. While their performance on ordinary passives is still good in other domains, their performance on reduced passives is much less so, and retraining them is expensive, requiring detailed annotation of large corpora. Shallow parsers are generally faster and more tolerant of ungrammatical input than full parsers. But currently, most shallow parsers do not recognize verb voice, and those that do don't recognize reduced passives or produce enough syntactic information to support recognition of reduced passives through minimal postprocessing.

My research hypothesis is that it is possible to recognize reduced passives using a shallow parser. This will provide voice labels (and thus, thematic role recognition) to applications that rely on shallow parsers. My thesis research concerned a classification approach for recognizing reduced passives: extracting feature vectors from a shallow parser's output and training a classifier with these vectors to distinguish reduced-passive verbs from other verbs. The next chapter discusses details of this approach.

CHAPTER 3

A CLASSIFIER FOR REDUCED PASSIVE RECOGNITION

Chapter 2 showed that full parsers can, with minimal postprocessing, perform reduced-passive recognition quite well. Shallow parsers, on the other hand, cannot; their grammars do not include specific structures corresponding to reduced passive constructions. However, there are applications in which shallow parsers are preferred and for which verb voice identification is important, such as information extraction or question answering. My research goal is to find a method for reduced passive recognition using a shallow parser.

I chose to view reduced passive recognition as a binary classification problem: that is, every verb in a given text can be viewed as belonging to one of two groups, REDUCED-PASSIVE or NOT-REDUCED-PASSIVE. Problems of this type can be solved with *learned classifiers*, which are programs that analyze a set of examples whose classifications are known and derive a model for classifying novel examples. The idea in a learned-classifier setting is that each verb should be represented by some set of *features* which encode relevant and important properties of that verb with respect to classifying it as reduced passive or not. The choice of meaningful features is the key to using a learned classifier successfully. I hypothesized that a shallow parser could provide enough information about verbs to describe and classify them in this manner.

Figure 3.1 shows the process for training and testing a reduced passive classifier. The "Data preparation" stage, at the top, consists of two tasks. First, shown in the upper left, we need to assemble a text corpus whose verbs are annotated, correctly labeled as REDUCED-PASSIVE or NOT-REDUCED-PASSIVE. The different approaches to doing this are detailed in Section 3.1. Second, shown in the upper right, we collect knowledge about verbs from a second corpus. This knowledge concerns the verb's transitivity and expected thematic roles and is described fully in Section 3.3.1. The second corpus does not need to be annotated in any way.

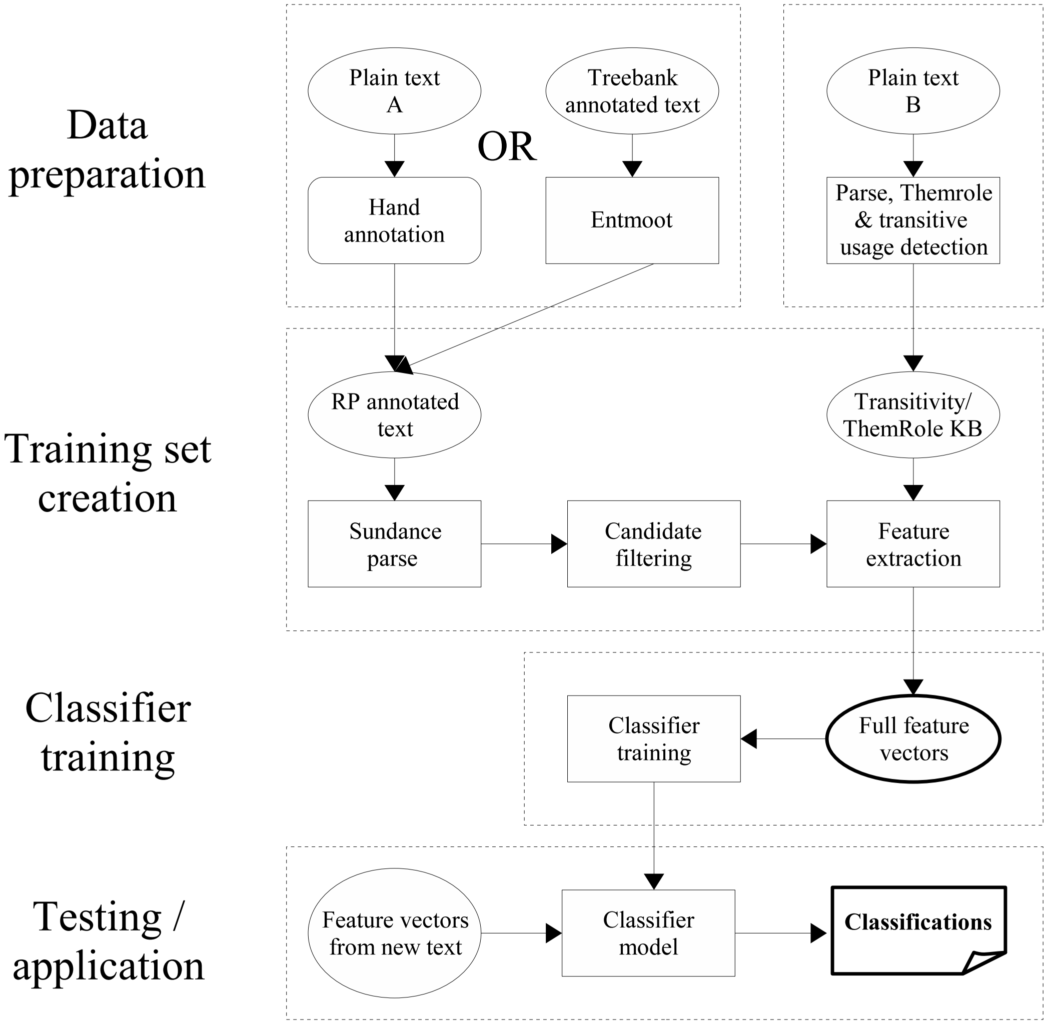

Figure 3.1: Complete classifier flowchart.

Next, in the "Training set creation" stage, the data gathered in the data preparation stage is processed to create a list of *feature vectors* corresponding to the possible reduced-passive verbs in the annotated text. Each feature vector is a set of feature values describing one verb in the text, paired with the verb's correct classification as REDUCED-PASSIVE or NOT-REDUCED-PASSIVE. Together, these feature vectors and classifications comprise the *training set*. Descriptions of the features and the motivations for them form the bulk of this chapter, including Sections 3.2 and 3.3.

Third, during the "Classifier training" stage, the classifier analyzes the training set and produces a model for classifying novel examples of verbs described in terms of their features. This is discussed in Section 3.5. Finally, in the "Testing / application" stage, the model is evaluated for its effectiveness in classifying reduced passives, or it is given new data for reduced-passive recognition. I will present experimental results on several test sets in Chapter 4.

3.1 Data sets

To be effective, a learned classifier needs a substantial amount of training data. In this case, part of that requirement is a large number of verbs correctly labeled as REDUCED-PASSIVE or NOT-REDUCED-PASSIVE. The representation I used for that is plain text with reduced passives marked, like the example given in Section 2.4:

```
The dog washed/RP yesterday was fed this morning.
```

Any verb marked /RP is REDUCED-PASSIVE and any other verb is assumed to be NOT-REDUCED-PASSIVE. I call this *RP-annotated text*.

There are two methods of creating RP-annotated text. The first, shown in Figure 3.2, is to annotate "by hand," reading the plain text in a text editor and adding the `/RP` notation to reduced passives. The problem is that in any supervised-learning task, the training set needs to be sufficiently large to support the creation of a comprehensive model. Hand-annotating enough documents would be expensive for a single researcher (though certainly not as bad as Treebank annotation; it would be feasible for a research group with a few dedicated annotators).

The second, shown in Figure 3.3, is to use Entmoot (see Section 2.3), which can convert existing Treebank data into RP-annotated text. Enough Treebank data already exists to create a reasonably-sized training set for a proof of concept. Further, while Entmoot's classifications aren't perfect, they are sufficiently accurate. The Entmoot classification of verbs to REDUCED-PASSIVE and NOT-REDUCED-PASSIVE are effectively the work of human experts, those who created the Treebank annotations. Agreement between Entmoot markup and my hand-annotated Treebank gold standard is high: for

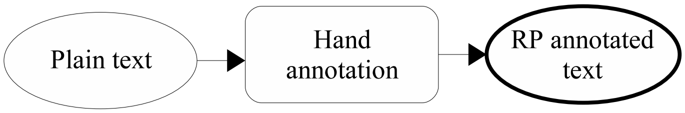

Figure 3.2: Manual RP-annotated text preparation process.

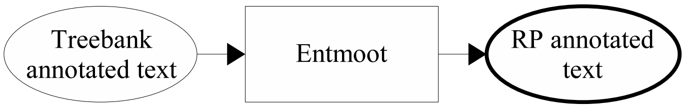

Figure 3.3: RP-annotated text preparation using Entmoot.

reduced passives, recall is 92.99% and precision is 92.78%, yielding an F-measure of 92.88%.

The Entmoot-derived training set, created from 2,069 *Wall Street Journal* articles, contains 27,863 verbs: 23,958 NOT-REDUCED-PASSIVE and 3,905 REDUCED-PASSIVE. I refer to this set of 2,069 documents as the *Entmoot corpus*.

For test data, I also used the hand-annotated gold standards described in Section 2.4: 50 *Wall Street Journal* articles from the Penn Treebank (none of which appear in the Entmoot corpus), 200 documents from the MUC-4 terrorism corpus, and 100 documents from the ProMED mailing list. These three corpora contained roughly the same number of REDUCED-PASSIVE verbs: 442 in the *Wall Street Journal* articles, 416 in the MUC-4 documents, and 463 in the ProMED documents.

3.2 Creating feature vectors

Figure 3.4 shows the process by which RP-annotated text is converted into training set feature vectors. First, each sentence in the RP-annotated corpus is processed by the Sundance shallow parser. This provides information about the constituent words and phrases of the sentence, including part-of-speech tagging, verb tense identification and association of verbs and their modifiers into contiguous VPs. By itself, this information is enough to disqualify a large fraction of verbs as reduced passives; the remaining verbs are possibly reduced passives and are called *candidate RP verbs*. Candidate RP verbs each then receive a feature vector derived from the Sundance parse and the classification they had in the RP-annotated text – those with an `/RP` label are classified REDUCED-PASSIVE and any without are classified NOT-REDUCED-PASSIVE.

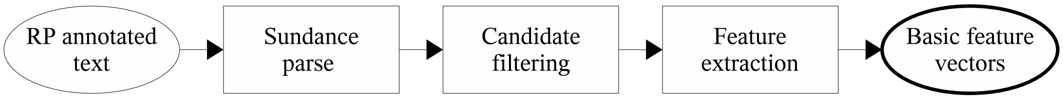

Figure 3.4: Basic feature vectors.

Candidate filtering is described in detail in Section 3.2.1, and the feature vector constructed for each candidate RP verb is discussed in Section 3.2.2.

3.2.1 Candidate filtering

Most verbs are clearly active voice or ordinary passive, and they can be immediately disqualified from being reduced passives. Those remaining, the candidate RP verbs, are *possibly* reduced passives. Candidate RP verbs are identified by seven characteristics easily found in a shallow parse:

POS-Tagged as verb: It is an implicit candidacy requirement that the part-of-speech tagger recognizes that the candidate is a verb. Any word not tagged as a verb is not considered for candidacy. Accordingly, POS-tagging errors can cause genuine reduced passives to fail candidacy filtering.

Past tense: All English passive voice verbs are past participles. However, Sundance does not distinguish past tense and past participle forms, considering both past tense. Any verb that is not past tense, however, cannot be passive, so we remove all non-past-tense verbs from consideration.

Not auxiliary verb: The verb in question should not be an auxiliary verb. In practice, this means that verbs that can act as auxiliaries - *be, have*, etc. - which are not the head verb in their VP are rejected.

The remaining candidacy requirements deal with premodifiers to the verb. That is, words that occur within the same base verb phrase, preceding the verb in question. Premodifiers of interest may be auxiliaries or modals; adverbs are ignored. Sundance does not allow VPs to contain other phrases, so intervening NPs or other phrases would break the premodifier relationship. For instance, the verb *has* in “Sam has the dog washed every Thursday” is not a premodifier to the verb *washed*, since the two would occur as separate VPs with an intervening NP (*the dog*).

Not ordinary passive: If a verb has a passive auxiliary, it cannot be a reduced passive. Rather, it is an ordinary passive, so any verbs with a passive auxiliary are rejected.

Not perfect: a perfect (*have*) auxiliary indicates an active-voice construction, such as “Sam *has washed* the dog.” Any verbs that have the perfect auxiliary are removed from consideration.

No “do” auxiliary: A *do* auxiliary, as in “Sam *did not* go to the store,” implies an active-voice usage. Since, as in this example, the auxiliary takes the past tense conjugation, it is unusual to find *do* auxiliaries with past tense verbs; it tends only to happen with head verbs that have past-tense forms the same as their uninflected forms, such as *let*, which the parser mistags as past tense. Verbs with a *do* auxiliary are rejected.

No modals: Similarly, modals preceding the verb in question imply active voice. Like the *do* auxiliary, cases with modals are unusual but do occur, particularly with ambiguously-conjugated verbs, as in "Mary *will let* the students out." It is possible for ordinary passives to have modals, as in "The apples will be eaten," but not reduced passives.

Any verb *not* rejected by any of these tests is a candidate RP verb. Only candidate RP verbs receive feature vectors in the classifier training and test data - though these candidates include both positive and negative examples, since the candidacy requirements are not a perfect determination of reduced-passive classification. Non-candidates are left out of the feature vector set and assumed to be NOT-REDUCED-PASSIVE.

3.2.2 Basic features

Table 3.1 shows the *basic* feature set, i.e., the features that can be extracted from a Sundance parse with no additional processing or external data.

The candidacy requirements might be thought of as features and were used as such in early experiments. Ultimately, however, I decided to use them as a filtering step rather than features since they were so successful in disqualifying nonreduced-passive verbs and they reduce the data set size and feature dimensionality. The basic features are not as definitive in their relationship to a verb's reduced-passive status as the candidacy requirements.

The Sundance parser provides more detailed analysis than most shallow parsers, though not as much as full parsers. I exploit three properties that it has that simple phrase chunkers do not:

1. It associates semantic classes with nominals.
2. It identifies clause boundaries.
3. It assigns syntactic roles to phrases such as subject, direct object, and indirect object.

These support a set of 23 features, which I categorize into the following groups: Lexical, Syntactic, Part-of-Speech, Clausal, and Semantic. In the following subsections, I describe each group of features in detail.

Table 3.1: Basic feature set.

Feature	Description
L1	Verb's root
S1	Does the verb have a (syntactic) subject?
S2	Root of subject NP head (if any)
S3	Is subject a nominative pronoun?
S4	Does verb have a following "by"-PP?
S5	Root of following "by"-PP NP head (if any)
S6	Does verb have a direct object?
S7	Does verb have an indirect object?
P1	Is verb followed by a verb, aux, or modal?
P2	Is verb followed by a preposition?
P3	Is verb preceded by a number?
P4	Part of speech of word preceding verb
P5	Part of speech of word following verb
C1	Is sentence multiclausal?
C2	Does sentence have multiple head verbs?
C3	Is verb followed by infinitive?
C4	Is verb followed by new clause?
C5	Is verb in the sentence's last clause?
C6	Is verb in the last VP in the sentence?
M1	Low-level semantic class of subject NP
M2	Low-level semantic class of nearby "by" PP
M3	Top-level semantic class of subject NP
M4	Top-level semantic class of nearby "by" PP

3.2.3 Lexical feature

The basic feature set contains one lexical feature, which is the root of the verb. Some verbs occur commonly or even predominantly in passive voice - for instance, it is very rare to see the verb *infect* in active voice in the ProMED corpus. For such verbs, the root alone can be a strong indicator that a given usage is a reduced passive since ordinary passives are rejected by the candidacy filter.

3.2.4 Syntactic features

While the Sundance shallow parser does not assign specific grammatical structures to reduced passives the way the Treebank grammar does, it supplies potentially useful syntactic information including phrase structure and syntactic role labels. This information can be used to extract features related to the verb's syntactic subject (if any), any prepositional phrase headed by the preposition *by* that closely follows the verb, and direct and indirect objects (if any) of the verb. Sundance recognizes that a single NP may play different syntactic roles with respect to multiple verbs. For instance, in the sentence "I fed the dog washed by Sam," *the dog* is the direct object of *fed* and the subject of *washed*. In these cases, Sundance inserts a copy of the NP before the second verb. It assigns the direct object role to the first copy of the NP and the subject role to the second. Table 3.2 lists the syntactic features, which I describe in detail below.

3.2.4.1 Subject features. generally, we expect NPs in the subject position with respect to a reduced passive verb to behave as themes. The subject features provide some clues about the verb's subject.

3.2.4.2 (S1): Does the verb have a (syntactic) subject? Presence or absence of a subject can be a clue to reduced passive constructions.

3.2.4.3 (S2): Root of Subject NP head (if any). it is possible that certain words occur exclusively, or nearly so, as themes of a verb. For instance, the training corpus might have had some form of *dog* as the theme of *walk* every time *walk* had a theme. If so, an occurrence of the word *dog* in subject position with respect to *walk* could be a strong indicator a of reduced passive.

3.2.4.4 (S3) Is subject (if any) a nominative pronoun? Since ordinary passive voice constructions have already been filtered, if a candidate verb has a nominative-case pronoun (e.g., *she*) as its subject, then it generally corresponds to an agent rather than a theme. Therefore, a nominative-case pronoun in the subject position is a very strong indication that a verb is in an agent-verb syntactic order, hence active voice.

3.2.4.5 Following "by"-PP features. A *following "by"-PP* is a prepositional phrase, headed by the preposition *by*, which occurs after the verb, either adjacent to the verb or separated by at most one other constituent. This is an approximation for PP attachment to the verb, which shallow parsers don't provide. Passive-voice verbs commonly have such PPs closely following them, often containing the agent NP (except in some cases such as when the *by* is locative).

Table 3.2: Syntactic features.

Feature	**Description**
S1	Does the verb have a (syntactic) subject?
S2	Root of subject NP head (if any)
S3	Is subject a nominative pronoun?
S4	Does verb have a following "by"-PP?
S5	Root of following "by"-PP NP head (if any)
S6	Does verb have a direct object?
S7	Does verb have an indirect object?

3.2.4.6 (S4) Does verb have a following "by"-PP? The simple presence of a nearby following "by"-PP is strong evidence of passive voice.

3.2.4.7 (S5) Root of following "by"-PP NP head (if any). as feature S2 does with potential themes, this feature can capture words that commonly occur as agents for certain verbs.

3.2.4.8 Object features. generally, we do not expect NPs in the direct and indirect object positions with respect to reduced passive verbs. The object features describe the verb's direct and indirect objects.

3.2.4.9 (S6) Does verb have a direct object? Passive-voice verbs generally do not have direct objects. However, ditransitive verbs can have them in reduced passive constructions. For example:

`The students given/RP` *`books`* `went to the library.`

In this case, the subject (*the students*) fills the *recipient* thematic role, not the *theme*, but *given* is still a reduced passive. Consequently, the presence of a direct object is not by itself strong enough to disqualify a verb as a reduced passive but it may be a valuable clue.

3.2.4.10 (S7) Does verb have an indirect object? Sundance's definition of indirect objects does not include those found in prepositional phrases, such as *these students* in the sentence "The books given to these students were new." Rather, indirect objects are always NPs following a verb, like *Mary* in the sentence "John gave Mary the book." Given this definition, an indirect object is very unlikely to occur after a passive-voice verb.

3.2.5 Part-of-speech features

Reduced passive constructions may be signaled by the parts of speech of words that immediately precede or follow the verb. Table 3.3 shows the features that relate to these adjacent POS tags.

3.2.5.1 (P1) Is verb followed by another verb, auxiliary, or modal? A verb followed by another verb, a modal, or an auxiliary verb may be a reduced passive in a reduced relative clause. For example, the reduced passive verb *interviewed* in this sentence is followed by the verb *felt*:

`The students interviewed/RP` *`felt`* `that the test was fair.`

Table 3.3: Part-of-speech features.

Feature	**Description**
P1	Is verb followed by another verb, aux, or modal?
P2	Is verb followed by a preposition?
P3	Is verb preceded by a number?
P4	Part of speech of word preceding verb
P5	Part of speech of word following verb

This feature is similar to the clausal features described in Section 3.2.6.

3.2.5.2 (P2) Is verb followed by a preposition? From inspection of RP-annotated text, it happens quite often that reduced passives are followed by prepositions, as in:

`The soap used/RP` *`for`* `washing dogs smells good.`

or

`Six packages found/RP` *`in`* `the house contained cheese.`

Of course, if the preposition is *by* it is a special case, as noted before, but other prepositions commonly occur after reduced passives as well.

3.2.5.3 (P3) Is verb preceded by a number? Numbers in isolation – that is, not modifying a noun – sometimes act as implied subjects to verbs. One case of this is in parallel constructions such as conjoined VPs, when the object being enumerated is mentioned in the first VP and elided from subsequent VPs. For example:

`Authorities reported 20 people killed/RP and` *`100`* `injured/RP in the attack.`

In cases like these, numbers immediately before candidate RP verbs can indicate reduced passives.

3.2.5.4 (P4) Part of speech of word preceding verb. Features P1-3 represent specific part-of-speech collocations that seem useful for identifying reduced passives, based on my inspection of reduced passives in texts from the *Wall Street Journal*, MUC-4, and ProMED corpora. Other such collocations may exist that I have not observed, especially in novel corpora. This feature simply records the part of speech for the word preceding the candidate RP verb. If there are unobserved correspondences between reduced passives and a certain part of speech for the word preceding a candidate RP verb, this feature may enable the classifier to learn them.

3.2.5.5 (P5) Part of speech of word following verb. This is the same as P4, but for the word following the verb.

3.2.6 Clausal features

Reduced passives commonly occur in sentences with multiple clauses. Unlike many shallow parsers, the Sundance parser identifies clause boundaries,[1] which can be useful clues for finding reduced passives because a common type of reduced passive is that which occurs in a reduced relative clause, like the verb *washed* in the sentence:

`The dog` ***`washed/RP by Sam`*** `was brown.`

Alternatively, the reduced relative clause may be shifted to the beginning of the sentence for emphasis, ending up in a separate clause from its theme. For example, the italicized clause in the following sentence has been shifted:

`Stung/RP by a thousand bees,` `Maynard cursed furiously.`

Because reduced passives occur in multiclausal sentences, the basic feature set includes features related to the clausal structure of the verb's sentence, which are listed in Table 3.4.

3.2.6.1 (C1) Is sentence multiclausal? This feature says whether the containing sentence has multiple clauses by Sundance's clausal-boundary standards.

3.2.6.2 (C2) Does sentence have multiple head verbs? The Sundance clause boundaries do not always identify structures like reduced relative clauses because it may collect multiple successive verbs, or verbs followed by infinitive-*to* and a further verb, into a single VP instead of dividing them with a clause boundary. This feature reports whether the containing sentence has multiple non-auxiliary verbs, which is a finer-grained approximation of multiclausality.

3.2.6.3 Following-clause features. The previously mentioned clausal features attempt to confirm the verb's occurrence in a multiple-clause sentence. However, certain

[1] These clause boundaries are "shallow" clauses, approximations that do not address some subtleties like embedded clauses.

Table 3.4: Clausal features.

Feature	**Description**
C1	Is sentence multiclausal?
C2	Does sentence have multiple head verbs?
C3	Is verb followed by infinitive?
C4	Is verb followed by new clause?
C5	Is verb in the sentence's last clause?
C6	Is verb in the last VP in the sentence?

clausal structures may argue against a verb's being a reduced passive. Clause boundaries occurring immediately after a verb may indicate clausal complements. For example:

`Sam believed` ***`that dogs should be washed.`***

Here *believed* is in active voice, with the following italicized clause behaving as a complement (argument) of *believed*. Verbs whose subcategorization frames include a clausal complement often appear in the active voice without a direct object, so recognizing the clausal complement may help the classifier to identify these cases. The remaining features point out the possible existence of this kind of clausal complement.

3.2.6.4 (C3) Is verb followed by infinitive? Infinitives following verbs can be infinitive complements:

`Sam tried` *`to wash the dog.`*

3.2.6.5 (C4) Is verb followed by new clause? This feature points out the existence of a following clause.

3.2.6.6 (C5) Is verb in the sentence's last clause? Verbs occurring in the last clause of a sentence cannot have following clauses.

3.2.6.7 (C6) Is verb in the last VP in the sentence? This is the same as feature C5 except, instead of using the parser's clause boundaries, it uses the assumption used by feature C2, that head verbs may define clause boundaries that Sundance does not recognize. Thus, if the verb is the last nonauxiliary verb in the sentence, this feature considers it to be in the last clause.

3.2.7 Semantic features

Sundance's parser assigns semantic tags to nouns according to *semantic classes*. A semantic class is a label associated with a given noun in the Sundance dictionary. For instance, the dictionary entry for the word *dog* associates it with the part of speech NOUN and the semantic class label ANIMAL. Other words, like *pig* and *walrus* could have the ANIMAL label as well, and therefore belong to the same semantic class. Noun phrases receive the semantic class of their head noun. Semantic classes are also related to one another through a *semantic hierarchy* like the one shown in Figure 3.5. The hierarchy is a tree whose nodes are semantic classes and whose edges denote a subclass relationship; for example, ANIMAL is a subclass of ANIMATE. Figure 3.5 shows a small sample hierarchy. The complete hierarchy I used for my experiments is shown in Appendix B.

The following two example sentences show the verb *washed* first in active voice and then as a reduced passive. The agent and theme NPs are shown with their semantic classes.

`Sam(HUMAN)` *`washed`* `the dog(ANIMAL).` : ACTIVE

`The dog(ANIMAL)` *`washed`* `by Sam(HUMAN) was brown.` : REDUCED-PASSIVE

A human reader would have a sense, based on world knowledge, that dogs do not wash anything; in general, animals do not wash things. On reading the the second sentence, they would assume that *the dog* is not the agent of *washed*, and therefore that *washed* is not in active voice. The basic feature set's semantic features, shown in Table 3.5, attempt to capture this kind of world knowledge.

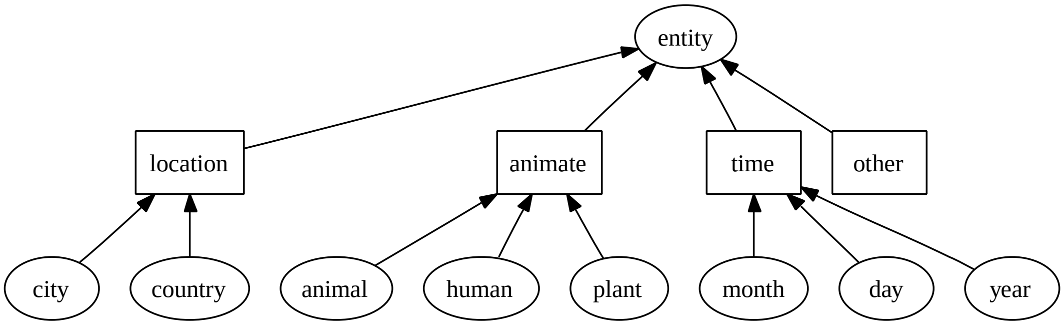

Figure 3.5: A sample semantic hierarchy.

Table 3.5: Semantic features.

Feature	Description
M1	Low-level semantic class of subject NP
M2	Low-level semantic class of nearby "by" PP
M3	Top-level semantic class of subject NP
M4	Top-level semantic class of nearby "by" PP

Recall that it is important to recognize reduced passives because of their displacing effect on agents and themes from their expected syntactic positions. The semantic features provide a clue that such displacements have taken place around a given verb. Accordingly, the semantic classes of certain syntactic positions associated with agents and themes are of special interest; these are the subject and nearby following "by"-PP positions.

One potential problem is that semantic class associations may be sparse because too few sentences contain words of a certain class together with a given verb root. This is the motivation for including *top-level* semantic features, as well as *low-level* semantic features. Top-level semantics are found by following the semantic hierarchy from the low-level semantic class toward the root, stopping one step before the root itself. In Figure 3.5 the top-level classes are shown in rectangles. They are the most general semantic classes. In contrast, the low-level semantic class is the specific semantic tag assigned to a word in the dictionary.

3.2.7.1 (M1) Low-level semantic class of subject NP. This is the semantic class of the subject NP. If Sundance's dictionary does not have a semantic class for the NP's head noun, this feature gets a special NONE value.

3.2.7.2 (M2) Low-level semantic class of nearby following "by"-PP. Semantic class of the NP in a nearby following "by"-PP as defined in Section 3.2.4, or NONE.

3.2.7.3 (M3) Top-level semantic class of subject NP. Top-level semantic class of subject NP (or NONE).

3.2.7.4 (M4) Top-level semantic class of nearby "by"-PP. Top-level semantic class of nearby following "by"-PP (or NONE).

The semantic features are the last of the five subsets of the basic feature set. The basic feature set provides several kinds of information about a verb, some intended to reinforce the notion that the verb is a reduced passive, others to suggest that it is not. As I will show in Chapter 4, the basic features perform fairly well on their own but there are other features that can be added to the feature set with a small amount of additional data preparation that may also be helpful. In the next section, I describe these: transitivity and thematic role features.

3.3 Transitivity and thematic role features

While the basic features capture a wide variety of properties about a verb, there are other properties of passive-voice verbs that they do not encode and which can be represented with a small amount of data preparation. There are two of these: transitivity and thematic role semantics.

Since passive verbs require themes, only transitive verbs can be used in passive voice. If we know that a verb is intransitive, then we can disqualify it from consideration as a reduced passive. Unfortunately, many verbs can be used either transitively or intransitively, so the fact that a verb can be used intransitively is not a strong enough reason to disqualify it. Still, an estimate of how likely the verb is to be transitive might be useful knowledge.

Second, the purpose of recognizing reduced passives is to avoid misconstruing their thematic roles. If it is possible to discern that a verb has likely agent and theme NPs occupying passive-voice syntactic positions, we can propose that the verb is passive. To do this, we need to know what kind of NPs typically act as agents and themes for the given verb. The basic feature set addresses this indirectly through some of the subject and *by*-PP features mentioned in Sections 3.2.4 and 3.2.7. The root-of-NP-head features can capture commonly-occurring nouns, and the semantic features can identify the general semantics of NPs that appear in those positions.

It would, however, be useful to have features that state explicitly whether the subject is a likely agent or theme, and whether a *by*-PP contains a likely agent. These could help the classifier recognize when the verb's agent and theme are in passive-voice syntactic positions.

Both transitivity and expected thematic role fillers are properties that require knowledge about verbs beyond what is found in Sundance's dictionary and the equivalent resources used by other shallow parsers. The next section describes how I created a knowledge base to supply it.

3.3.1 Transitivity and thematic role knowledge

Figure 3.6 shows how the knowledge base is built. An unannotated text corpus is parsed by the Sundance shallow parser, which identifies constituents and assigns syntactic

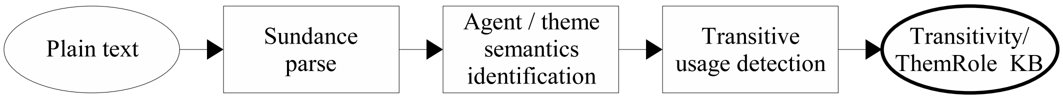
Plain text
Sundance parse
Agent / theme semantics identification
Transitive usage detection
Transitivity/ ThemRole KB

Figure 3.6: Building transitivity / thematic role knowledge base (KB).

and semantic labels to each NP. Then, the knowledge base tool examines each verb. It estimates the percentage of the verb's occurrences that are transitive.

The tool considers the verb together with its context to decide whether it is being used transitively in each case. This transitivity detection is similar to that of [21]. A verb usage is considered transitive if:

- the verb has no passive auxiliary but does have both a subject and a direct object (active voice), OR

- the verb is in past tense and has a passive auxiliary (ordinary passive).

Second, the tool estimates each verb's thematic role semantics. The detection is similar to that used by [30] and [5], but less sophisticated. It looks for two cases similar to the transitivity cases, one for active voice and one for passive voice. Active voice verbs are expected to follow the "obvious" S-V-O syntactic order, such that their agents are in subject position and themes are in direct object position. Ordinary passives are assumed to have their themes in subject position and agents in a following "by"-PP.

More specifically, the thematic role semantics are detected using these rules:

1. If the verb is in active voice: If it has both a subject and a direct object, the subject's top-level semantic class is recorded as an agent and that of the direct object is counted as a theme.

2. If the verb is an ordinary passive: If it has a subject, the top-level semantic class of the subject is added as a theme type. If it has an adjacent following *by*-PP, the top-level semantic class of the NP within the *by*-PP is added as an agent type unless the class is LOCATION, BUILDING, or TIME. These classes are excluded because NPs of these types are very likely to be locative or temporal entities.

I used only top-level semantics because I believed that low-level semantics would be too sparse. Using low-level semantics is a possible direction for future research.

Once the entire corpus has been processed, the resulting knowledge base contains three types of information for each verb root: 1. how many times it occurred in total; 2. how many times it appeared to be used transitively; and 3. a list of the top-level semantic classes that occurred as apparent agents and themes for that verb. The collection of these lists for all verbs encountered in the corpus comprises the Transitivity / Thematic Role knowledge base (*Transitivity / ThemRole KB* or simply *KB*).

Note that this tool has no awareness of reduced passives; most likely, those would be misconstrued as active voice. Since reduced passives typically do not have direct objects and the tool requires active-voice verbs to have direct objects to be considered transitive, reduced passives would probably be considered intransitive usages. Consequently, the transitivity counts in the knowledge base are, for this and other reasons such as misidentification of direct objects, properly considered to be a lower bound on the verb's true transitivity. For the same reason, reduced passives are also unlikely to have an effect on thematic role semantics; they would appear to be active-voice verbs with no direct object, and semantics are only recorded for active-voice verbs that do have direct objects. However, as I described in Section 3.2.4, there are cases such as ditransitives where reduced passives have direct objects, so there is some noise in this data.

The following sections describe the classifier features that are based on the KB.

3.3.2 Transitivity feature

3.3.2.1 (TRANS1) Transitivity rate for verb root. The transitivity rate for a verb is the number of transitive occurrences recorded for its root divided by its total number of occurrences. In theory, this rate should correspond to the degree to which the verb is transitive in the domain used to build the knowledge base; verbs with low transitivity would be less likely to be in passive voice.

This feature is a "binned" representation of the transitivity rate which is converted into one of six values: 0%–20%, 20%–40%, 40%–80%, 80%–100%, and Unknown (verb root does not occur in knowledge base). This is for the benefit of classifier algorithms that require discrete symbolic values and to avoid undue complexity in decision-tree models.

3.3.3 Thematic role features

Table 3.6 shows the features related to thematic role semantics.

The thematic role features rely on a notion of "plausibility" with respect to agent and theme semantic classes, similar to that expressed in [24]. The emphasis is not on finding which semantic classes are *likeliest* to be agents or themes for a given verb; rather, it is on finding those that are *possible* agents or themes. The knowledge base records all the top-level semantic classes encountered for a verb root in apparent agent and theme

Table 3.6: Thematic role features.

Feature	**Description**
THEM1	Is subject a Plausible Theme?
THEM2	Is “by” PP a Plausible Agent?
THEM3	Is subject a Plausible Agent?
THEM4	Frequency of verb root

positions. From these we can get a sense of whether a given semantic class could be an agent or theme for the verb. The criterion that I use to determine plausibility is that the semantic class should comprise not less than 1% of all agent or theme semantic types for the verb.

For instance, assume the verb *wash* had 1000 identifiable agents in the training corpus, 555 of which were ANIMATE, 444 of which were BUILDING, and 1 that was a LOCATION. There's a good chance that that LOCATION agent instance was the result of a misparse, dictionary shortcoming, or other “noise”; the 1% plausibility criterion would reject LOCATION as an agent class because it only accounts for 0.1% of agents seen during the construction of the knowledge base. ANIMATE, at 55.5% of occurrences, and BUILDING, at 44.4%, would both be considered plausible.

The four thematic role features are:

3.3.3.1 (THEM1) Is subject a Plausible Theme? Is the top-level semantic class of the subject a plausible theme for the verb? The theory is that a verb is likelier to be passive if a plausible theme is its subject.

3.3.3.2 (THEM2) Is “by” PP a Plausible Agent? Is the top-level semantic class of the “by”-PP pass a plausible agent for the verb? This feature looks for support of a passive-voice interpretation of the verb by seeing if the NP contained within a nearby following “by”-PP is a semantically plausible agent. If so, that should be evidence of passive voice, since the “by”-PP is the agent position for passive verbs.

3.3.3.3 (THEM3) Is subject a Plausible Agent? Is the top-level semantic class of the subject a plausible agent for the verb? A plausible agent in subject position is evidence that the verb is in active voice, since the subject is the agent position for active verbs.

3.3.3.4 (THEM4) Frequency of verb root. This feature is intended as a reliability indicator for the thematic role features. It is a binned representation of the verb's root's total number of occurrences in the training corpus; presumably, the agent and theme counts are more meaningful if the verb is more common. Low-frequency verbs may be less reliable due to the higher impact noisy data can have on them. Here, the bins are roughly logarithmic: 0 occurrences, 1–10, 11–100, or more than 100.

3.4 Full feature vector data

Figure 3.7 shows the process for constructing the full-feature training and test data. This consists of two steps: first, a Transitivity / ThemRole KB is built, as discussed in Section 3.3.1. Once the KB is finished, the second step is to extract feature vectors from the RP-annotated text for the training set. The KB supplies the knowlege needed for the transitivity and thematic role features. The same process and the same KB are applied to the gold standard RP-annotated text for the test set.

In theory, it would be useful if the text used to build the knowledge base were from the same corpus as the test set – since, it could be assumed, the verb usages would be more similar. Since the corpus used to generate transitivity and thematic role data doesn't need to be annotated, we can generate the knowledge base data from any text collection. For my research, I needed to use the Entmoot corpus (2069 annotated *Wall Street Journal* articles from the Penn Treebank) to train the classifier, because otherwise I would have needed to hand-annotate reduced passive verbs in an equivalent amount of text, which is prohibitively expensive for the scope of this research. Because the knowledge base training texts do not need annotation, however, I was able to build knowledge bases from

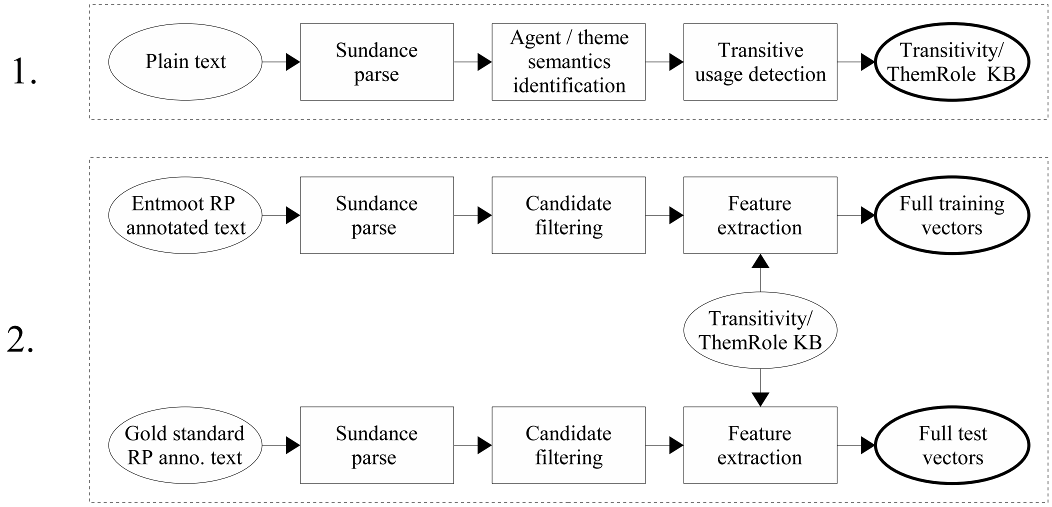

Figure 3.7: The process for creating feature vectors.

domain-specific corpora in the gold standards' domains.

I ran one set of experiments using knowledge bases from the same domain as the test set. I used the Entmoot corpus for the *Wall Street Journal* gold standard, 1225 MUC-4 documents for the MUC-4 gold standard, and 4959 ProMED documents for the ProMED gold standard. In every case the knowledge base training corpus had no documents in common with the test set. I called these the *Domain-specific Knowledge Base* or *DSKB* experiments.

A different argument could be made that it would be best to use the same texts both for the training set and for the knowledge base generation, and I performed experiments in this way as well. For these the knowledge base was built from the Entmoot corpus for each of the three gold standards, with the only difference being the Sundance domain-specific dictionaries that were used for each domain (in the same way as it was for building the basic feature vectors).[2] I called the experiments using this corpus for the knowledge base the *Wall Street Journal Knowledge Base* or *WSJKB* experiments.

Knowledge bases for the tenfold cross-validation were built from the 90% of the Entmoot corpus used for training in each fold, to avoid any unfair biasing by including the test data in any stage of training.

3.5 Training the classifier

Once the training feature vectors have been created, training a classifier is simply a matter of processing the training vectors with a machine learning (ML) algorithm to create a classification model. Figure 3.8 shows this process. For these experiments I used two different types of classifiers.

The first type was the *J48 decision tree (Dtree).* J48 is a version of C4.5 included in the Weka [38] machine learning package. Decision tree classifiers have the advantage that the models they produce are human-readable.

The second type was the *Support vector machine (SVM).* SVMs are a state-of-the-art machine learning method that have achieved good results for many NLP problems. I performed two sets of experiments using the SVM^{light} support vector machine software [14]: one with its default settings using a linear kernel, and one with its default settings

[2] Sundance can use different hand-built dictionaries, semantic lexicons, and lists of common phrases for different domains.

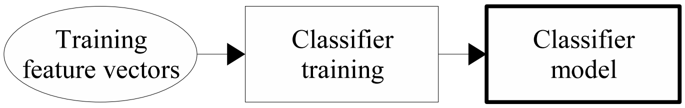

Figure 3.8: Classifier training.

using a polynomial kernel, which specify a degree 3 polynomial. I found that the polynomial kernel performed the best overall.

The classification model can then be applied to novel feature vectors to classify the corresponding verbs as REDUCED-PASSIVE or NOT-REDUCED-PASSIVE.

To summarize, treating reduced passive recognition as a verb classification problem requires that verbs be represented by a set of features that describe it in relevant ways. I chose several features that describe some properties of the verb itself, such as its root, and others which describe its context, such as syntactic, part-of-speech, and semantic properties of nearby words and phrases extracted from a shallow parse of the verb's sentence. Further useful features related to a verb's transitivity and its commonly occurring agent and theme semantic classes can be derived from a set of texts.

I created two tools to generate training data for a learned classifier model. One creates the transitivity and thematic role knowledge base from unannotated text. The other converts plain text with the reduced passives labeled into a set of feature vectors, each of which represents one verb together with its correct classification as REDUCED-PASSIVE or NOT-REDUCED-PASSIVE. I conducted experiments with several different classifier models using different learning algorithms and variations on the feature set, and Chapter 4 presents the results.

CHAPTER 4

EXPERIMENTAL RESULTS

This chapter presents the results of my experiments using variations on the feature set described in Chapter 3 and three different classifier models. There were two main feature set variations: the *basic feature set* which used only features that could be derived from a Sundance shallow parse, and the *full feature set* which used the transitivity and thematic role features discussed in Section 3.3 in addition to the basic feature set. I also performed ablation tests in order to observe the contributions made by small subsets of the full feature set and see which particular features were the most useful in reduced passive recognition.

Section 4.1 describes the data sets on which the experiments were performed. Section 4.2 presents four simple tests used to create baseline scores to which the classifiers' scores were compared. Section 4.3 discusses the performance of the basic feature set, and Section 4.4 presents the results of using the full feature set. Finally, Section 4.5 discusses ablation testing and analysis of the errors arising from my classification approach.

4.1 Data sets

I performed experiments on four different data sets:

XVAL: 10-fold cross validation over the Entmoot corpus of 2,069 *Wall Street Journal* articles from the Penn Treebank.

WSJ: The training set is the 2,069-document Entmoot corpus. The test set consists of the 50-document hand annotated *Wall Street Journal* gold standard.

MUC: The training set is the 2,069-document Entmoot corpus. The test set consists of the 200-document hand annotated MUC-4 gold standard.

PRO: The training set is the 2,069-document Entmoot corpus. The test set consists of the 100-document hand annotated ProMED gold standard.

Throughout this chapter, I present the experiments' scores using the short names given in boldface above for the data sets. For comparison, I also present four *baseline* scores for each data set, which I describe in the next section.

4.2 Baseline scores

I ran four different "baseline" experiments to see if simple heuristics were sufficient to recognize reduced passives. Their scores are given in Table 4.1. The four different tests were these:

Candidacy: For this test, every verb which passed the candidacy requirements outlined in Section 3.2.1 was labeled a reduced passive.

+MC+NoDO: Any candidate RP verb whose sentence was multiclausal and which did not have a direct object was labeled a reduced passive.

+ByPP: Any candidate RP verb which had a following "by"-PP was labeled a reduced passive.

+All: Finally, this test combined all the requirements of the previous three tests: candidacy, multiclausal sentence, no direct object, and following "by"-PP.

What the baseline tests show is that it is easy to get high recall or precision without using any sophisticated features or a learned classifier, but not both.

Table 4.1: Recall, precision, and F-measure results for cross-validation on Treebank texts (XVAL), and separate evaluations on WSJ, MUC4, and ProMED (PRO) test sets.

	XVAL			**WSJ**			**MUC4**			**PRO**		
Classifier	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**
Simple Baselines												
Candidacy	.86	.14	.24	.86	.15	.25	.85	.17	.28	.86	.25	.38
+MC+NoDO	.77	.23	.36	.74	.26	.38	.77	.25	.38	.78	.36	.50
+ByPP	.13	.68	.21	.13	.69	.22	.09	.75	.16	.10	.90	.17
+All	.12	.82	.20	.12	.81	.22	.09	.80	.16	.09	.93	.16
Classifier with basic feature set												
Dtree	.53	.82	.64	.48	.80	.60	.52	.78	.62	.43	.80	.56
LSVM	.55	.86	.67	.48	.86	.62	.47	.81	.59	.38	.80	.51
PSVM	.60	.87	.71	.53	.84	.65	.54	.83	.65	.42	.81	.55
Parsers												
Charniak				.90	.88	.89	.77	.71	.74	.77	.78	.77
Collins				.85	.89	.87	.66	.67	.66	.64	.78	.70
MINIPAR				.48	.57	.52	.51	.72	.60	.44	.68	.53

4.3 Classifier resuts for the basic feature set

The first section of Table 4.1 shows the baseline scores. The second section shows scores achieved by classifiers using the basic feature set compared with baseline results. Its columns contain the recall (R), precision (P), and F-measure (F) scores. In the lowest section, the full parser scores from Chapter 2 are shown again for comparison purposes.

The basic feature set performs much better than the baselines with all three learning algorithms: J48 decision tree (Dtree), linear-kernel SVM (LSVM) and degree-3 polynomial kernel SVM (PSVM). For every corpus other than ProMED, the PSVM scored highest, with 53% to 60% recall and 83% to 87% precision. For ProMED the Dtree was the best, with 43% recall and 80% precision.

The basic classifier also outperformed MINIPAR in all three gold-standard corpora. It did not do as well when compared to the Treebank-trained full parsers, though its MUC-4 F-measure is very close to that of the Collins parser.

In short, the basic feature set worked reasonably well. I hypothesized that transitivity and thematic role features might better describe the verbs' properties. Together with the basic feature set, these additional features form the *Full feature set*, which is evaluated in the next section.

4.4 Full feature set performance

Table 4.2 extends Table 4.1 to include the full feature set scores. The full feature set was used in two experiments: one with a transitivity / thematic role knowledge base (KB) trained on the same domain as the *Wall Street Journal* training set, and one with a KB trained on different domain-specific texts. Domain-specific knowledge base (DSKB) scores are given only for the MUC-4 and ProMED gold standards because the WSJ and XVAL knowledge bases were already from the same domain as their test sets. The best F-measures were achieved by the full feature set classifiers using the *Wall Street Journal* knowledge base (WSJKB). For the ProMED corpus, the linear kernel SVM achieved the highest F-measure, and the polynomial kernel SVM classifier performed the best for all other corpora. For every corpus, the full feature set / WSJKB classifier achieved precision of 80% or higher; recall was 60% or higher for every corpus except ProMED, where the best classifier scored 58%.

Generally, the full feature set is a clear improvement over the basic feature set: F-measures for the best-performing classifier climb by two points in the cross-validation, by 4 points for the WSJ and MUC-4 gold standards, and by 12 points in ProMED.

Table 4.2: Recall, precision, and F-measure results for cross-validation on Treebank texts (XVAL), and separate evaluations on WSJ, MUC4, and ProMED (PRO) test sets.

	XVAL			**WSJ**			**MUC4**			**PRO**		
Classifier	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**
Simple Baselines												
Candidacy	.86	.14	.24	.86	.15	.25	.85	.17	.28	.86	.25	.38
+MC+NoDO	.77	.23	.36	.74	.26	.38	.77	.25	.38	.78	.36	.50
+ByPP	.13	.68	.21	.13	.69	.22	.09	.75	.16	.10	.90	.17
+All	.12	.82	.20	.12	.81	.22	.09	.80	.16	.09	.93	.16
Classifier with basic feature set												
Dtree	.53	.82	.64	.48	.80	.60	.52	.78	.62	.43	.80	.56
LSVM	.55	.86	.67	.48	.86	.62	.47	.81	.59	.38	.80	.51
PSVM	.60	.87	.71	.53	.84	.65	.54	.83	.65	.42	.81	.55
Classifier with full feature set - WSJKB												
Dtree	.53	.82	.64	.49	.81	.61	.52	.78	.62	.43	.81	.56
LSVM	.62	.82	.71	.58	.79	.67	.61	.76	.67	.58	.81	.68
PSVM	.65	.84	.73	.60	.82	.69	.60	.80	.69	.54	.82	.65
Classifier with full feature set - DSKB												
Dtree							.49	.78	.61	.43	.81	.56
LSVM							.61	.77	.68	.56	.81	.66
PSVM							.61	.80	.69	.54	.81	.65
Parsers												
Charniak				.90	.88	.89	.77	.71	.74	.77	.78	.77
Collins				.85	.89	.87	.66	.67	.66	.64	.78	.70
MINIPAR				.48	.57	.52	.51	.72	.60	.44	.68	.53

Comparing F-measures, the full feature set PSVM classifier outperforms the Collins parser in the MUC-4 domain, and the LSVM classifier with WSJKB comes close to Collins in the ProMED domain. Charniak's parser is still the clear leader, but the classifier does outperform it in precision. With improvement in recall, the classifier could be competitive with the full parsers in the non-WSJ domains.

Contrary to my expectations, there does not appear to be any reason to prefer the domain-specific knowledge base (DSKB) over the Entmoot corpus knowledge base (WSJKB). Its scores are very close to the equivalent WSJKB scores, and where the difference is greater than one percentage point the DSKB is worse. My hypothesis had been that domain would affect the tendency of certain verbs to be used as reduced passives, possibly as a result of more common transitive senses of them; for instance, the verb *walk* in a corpus about dogs would be more likely to be transitive than in texts about different topics. Because of that, *walk* would more often occur as both ordinary and reduced passive. I also believed that genre would affect the frequency and context of reduced passive occurrences. These properties may be true, but have a smaller impact than I expected. It is also possible that the DSKB would work better if the training set were also from the same domain as the test set.

To determine why the classifier worked as well as it did, and where improvements might be made, I conducted several experiments with different subsets of the full feature set and examined specific cases of verbs for which the classifier succeeded or failed. The next sections present these experiments and an analysis of the classfier's performance.

4.5 Analysis

The scores in Table 4.2 show that the classifier with full feature set far outperforms the simple baselines presented in Section 4.2 and comes close to the performance of the Treebank-trained full parsers in non-Treebank domains. This chapter deals with an analysis of which features contributed most to the classifier's success, which if any hindered it, and a study of particular cases in which the classifier succeeded or failed.

Section 4.5.1 discusses the performance of classifiers using the full feature set with different subsets of features withheld, and Section 4.5.2 shows how those subsets perform on their own. Section 4.5.3 discusses some specific situations in which the classifier performed especially well or poorly.

4.5.1 Ablation testing

Ablation testing consists of peeling off, or *ablating*, parts of the feature vector to train a classifier with a subset of the original features. These tests are aimed at finding the contributions of various subsets of the full feature set toward the classifier's overall scores. I performed ablation studies on each subset identified in Chapter 3: lexical (LEX), syntactic (SYN), part-of-speech (POS), clausal (CLS), semantic (SEM), transitivity (TRN), and thematic (THM). Each test consisted of training and scoring the classifier twice – once with only the features in that subset, and once with all the features *except* that subset. The subset-only score provides some sense of the discriminating power of the subset itself, i.e., how much of the reduced-passive recognition problem it can solve on its own. The full-minus-subset score shows the impact of removing the subset, *i.e.*, whether other features can compensate for it or be sufficient without it. Since there can be synergistic effects or overlap among features, the two scores give a useful view when taken together. Since the polynomial SVM classifier performed the best overall, I used it to evaluate these ablated feature sets.

Table 4.3 shows the ablation testing results for feature sets consisting of the full feature set *except* each of the seven subsets. The top block of scores shows the full and basic feature set scores for comparison. The second block lists the ablation scores in approximate order of descending F-measure. Since the basic feature set does not include any of the features dependent upon the transitivity / thematic role knowledge base, WSJKB and DSKB scores are the same, and the knowledge base is identified as "None."

Interestingly, the features whose omission causes the greatest harm are some of the simplest: lexical (LEX) and part-of-speech (POS). The impact is both on recall and precision. This implies that these features account for a relatively large number of cases that are not captured by the rest of the feature vector. Close behind POS and LEX is the transitivity feature (TRN), whose omission caused a noticeable drop in recall in all domains, including a precipitous 13% in the ProMED domain using the WSJKB.

On the other hand, the semantic subset (SEM) appears to be detrimental in every domain except for the cross-validation; that is, removing these features improves scores. This is especially dramatic in the DSKB tests, where it costs three points of recall in the MUC-4 domain and four in the ProMED domain with no compensating increase in precision. Interestingly, the thematic features (THM), which are effectively a more

Table 4.3: Full-minus-subset ablation scores using PSVM classifier.

		XVAL			**WSJ**			**MUC4**			**PRO**		
Features	**KB**	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**
Full	WSJ	.65	.84	.73	.60	.82	.69	.60	.80	.69	.54	.82	.65
	DS							.61	.80	.69	.54	.81	.65
Basic	None	.60	.87	.71	.53	.84	.65	.54	.83	.65	.42	.81	.55
Ablation tests													
Full - SEM	WSJ	.65	.84	.73	.61	.82	.70	.60	.80	.68	.55	.81	.66
	DS							.63	.80	.70	.58	.81	.68
Full - CLS	WSJ	.64	.84	.73	.59	.83	.69	.59	.81	.68	.54	.83	.65
	DS							.58	.80	.67	.55	.83	.66
Full - THM	WSJ	.64	.84	.73	.59	.81	.68	.60	.81	.69	.54	.80	.64
	DS							.58	.80	.67	.52	.81	.64
Full - SYN	WSJ	.62	.83	.71	.58	.81	.68	.60	.81	.69	.52	.81	.63
	DS							.60	.81	.69	.52	.81	.63
Full - TRN	WSJ	.60	.86	.70	.53	.83	.65	.54	.80	.65	.41	.81	.54
	DS							.54	.80	.65	.44	.80	.57
Full - POS	WSJ	.60	.83	.69	.50	.79	.61	.55	.77	.64	.42	.79	.55
	DS							.55	.76	.64	.45	.82	.58
Full - LEX	WSJ	.54	.78	.64	.49	.76	.60	.57	.78	.66	.45	.77	.57
	DS							.56	.78	.65	.49	.78	.60

sophisticated form of the semantic features, contribute more to the overall score in the DSKB tests than they do in the WSJKB tests. This could mean that the thematic features are of higher quality when trained on texts in the same domain as the test set, which is consistent with expectations, although they still did not contribute much to overall performance.

The clausal features (CLS) also appear to contribute little, even causing a small amount of harm in the ProMED domain.

Overall, the scores do not drop greatly, regardless of which subset of features is withheld. This suggests a large degree of overlap among the subsets. The next section shows how well each subset performs on its own.

4.5.2 Single-subset testing

Table 4.4 shows the results of testing each of the seven feature subsets in isolation. Again, the table includes the full feature set and basic feature set for comparison purposes.

While none of the individual feature sets come close to the performance of the full feature set, the part-of-speech features (POS) perform surprisingly well by themselves. In F-measure terms they outperform all of the baselines. In the ProMED domain they come close to MINIPAR's performance. Likewise, the lexical feature (LEX) shows that just knowing the root of the verb is valuable. Its performance is noticeably better in the cross-validation and WSJ corpora, where the training and test sets belong to the same domain.

Table 4.4: Single-subset scores using PSVM classifier.

		XVAL			**WSJ**			**MUC4**			**PRO**		
Features	**KB**	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**	**R**	**P**	**F**
Full	WSJ	.65	.84	.73	.60	.82	.69	.60	.80	.69	.54	.82	.65
	DS							.61	.80	.69	.54	.81	.65
Basic	None	.60	.87	.71	.53	.84	.65	.54	.83	.65	.42	.81	.55
Ablation tests													
POS Only	None	.43	.60	.51	.36	.56	.44	.53	.57	.55	.43	.63	.51
SYN Only	None	.27	.78	.40	.26	.83	.40	.30	.82	.44	.18	.84	.29
LEX Only	None	.35	.71	.47	.26	.60	.36	.22	.54	.31	.20	.61	.30
THM Only	WSJ	.22	.72	.34	.23	.77	.36	.29	.75	.42	.16	.84	.26
	DS							.29	.82	.43	.16	.84	.27
SEM Only	None	.11	.74	.19	.11	.79	.20	.23	.87	.37	.11	.57	.19
TRN Only	WSJ	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
	DS							.00	.00	.00	.00	.00	.00
CLS Only	None	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00

This suggests that usage of particular verbs as reduced passives is domain-dependent.

The thematic features (THM) do some good on their own, clearly outperforming their less sophisticated counterparts, the semantic features (SEM). This makes sense because they combine knowledge of the verb and its semantic expectations, where the semantic features alone do not consider the verb. It is interesting to note that they do slightly better with the domain-specific knowledge base than with the WSJKB, including a seven-point gain in precision in the MUC-4 domain. This supports the observation in Section 4.5.1 that these features may be stronger if the knowledge base is derived from documents in the same domain as the test set.

The most interesting single subset of the features is the transitivity feature (TRN). By itself, it provides insufficient knowledge for the classifier to discriminate between reduced passives and other verbs. With every learning algorithm, the resulting classifier defaulted to classifying every verb as NOT-REDUCED-PASSIVE. However, as Table 4.3 shows, it makes an important contribution when combined with all of the other features.

Overall, the feature subsets tend to be low-recall and (relatively) high-precision. This is good if they account accurately for nonoverlapping subsets of the reduced passives in the test data. However, from the full feature set experiments we saw that there is a good deal of overlap, so the benefits of using all the features is not strictly additive.

In order to find more specific shortcomings of the feature set and possible directions for new feature development, I also examined some particular cases that performed well or poorly. The next section discusses these observations.

4.5.3 Error analysis

I examined output from the classifier to determine whether there were particular problems that might be solved by adding features or changing the parser or its supporting data. To do this, I collected the feature vectors and sentences for all the candidates in the gold standard corpora, associating each with its classification and the predictions made by the PSVM classifier using the full feature set and all the ablated versions shown in the previous sections. I also made observations about the reduced passives that were erroneously rejected by candidacy filtering and bad part-of-speech tagging. The following sections present my findings.

4.5.4 Incorrect candidate filtering

Probably the single biggest problem was Sundance's mistagging of reduced passives with nonverb parts of speech. Only verbs can qualify as candidates, so any mistagged reduced passive verbs are effectively false negatives. The impact is significant; 41 of 442 reduced passives in the WSJ gold standard, or 9% of them, were mistagged as non-verbs. The effect is even more severe in the other corpora: 62 of 416 (15%) of reduced passives were mistagged in the MUC-4 gold standard, and 50 of 464 (11%) in the ProMED gold standard.

About 90% of the mistagged verbs were tagged as adjectives. For some words, the *usage* as a verb or adjective is a difficult distinction to make, especially in constructions like this:

`3 *arrested* men were taken to the police station.`

In this sentence, *arrested* clearly behaves like an adjective modifying *men*, though it also conceptually behaves like a verb implying some event in which men were arrested. While this is not a reduced passive,[1] many reduced passive verb usages look enough like it that Sundance mistakenly tags them as adjectives.

Among the reduced passives correctly tagged as verbs, candidacy filtering incorrectly removes another fraction. This was negligible in the MUC-4 gold standard, but the WSJ gold standard lost a further 21 (nearly 5%) of its verbs and ProMED lost 21 as well (about 4%). There were several reasons for this, none of which seemed to be clearly dominant, though it was common for the verbs not to be recognized as past tense and therefore rejected. Altogether, incorrect candidate filtering incurs a serious recall penalty, since it rejects about 15% of the reduced passives in all three corpora.

Conversely, some precision loss was due to erroneous taggings of nonverbs as verbs, though again the most common case was the difficult distinction between verbs and adjectives (e.g., *"60 infected cows"*). There were also verbs that were wrongly considered to be candidates; many of these were ordinary passive constructions where the auxiliary verb was separated from the main verb far enough that Sundance did not put the auxiliary in the same VP. Others were active voice verbs in odd constructions such as the verb *set* in this sentence:

[1] Some linguists might view *arrested* as a verb in this case, which would make it a reduced passive; I view it as an adjective and therefore, since I have limited my research to verbs, not a reduced passive.

> Prosecutors, in an indictment based on the grand jury's report, maintain that at various times since 1975, he owned a secret and illegal interest in a beer distributorship; plotted hidden ownership interests in real estate that presented an alleged conflict of interest; set up a dummy corporation to buy a car and obtain insurance for his former girlfriend (now his second wife); and maintained 54 accounts in six banks in Cambria County.

4.5.5 Specialized word usages

Though a wide variety of verbs appeared as reduced passives, some verbs were consistently used that way. For example, the word *based* occurred 13 times in the WSJ gold standard and 10 of those were reduced passives (and they were all correctly classified as such). The following sentence is an example of a typical case:

> He founded Arkoma Production Corp., an oil and gas exploration company *based/RP* in Little Rock.

Since *based* is very commonly used in the passive voice, occurrences of it among candidate RP verbs will mostly be reduced passives. This may bias the classifier toward treating *based* as a reduced passive (though it is also likely that features besides the lexical feature will suggest it is reduced passive as well).

Similarly, some verbs occur very commonly as active voice or ordinary passive in the training set, which may bias the classifier against reduced passives with that root. For example, this reduced passive instance of *reported*, from the ProMED gold standard, was misclassified as active voice:

> No more deaths had occurred as a result of the outbreak beyond the two *reported/RP* yesterday.

The *Wall Street Journal* texts in the Entmoot training corpus seemed to use *reported* almost exclusively in active voice, which is plausible in newspaper writing because news articles will often credit other news agencies for content, e.g., "The AP reported a drop in new home sales." In the ProMED corpus, though, usages of *reported* were commonly reduced passives as in the sentence above. This may hint that common verbs are susceptible to domain-specific bias in usage. The classifier experiments showed that the classifier did get the specialized usages right if the specialized usage was consistent between the training and test sets.

4.5.6 Ditransitive and clausal-complement verbs

A source of errors was ditransitive verbs, which probably confuse the system because an indirect object can move into the subject position in the passive voice. For example, consider these uses of the verb *send*:

1. `I sent the newspaper a letter.` (active voice)
2. `A letter was sent to the newspaper.` (passive voice)
3. `The newspaper was sent a letter.` (passive voice)

In sentence 3, the direct object (*a letter*) is still present even though the verb is in passive voice. The sentence below shows a similar case that was mislabeled by the classifier:

`The Urdu-language newspaper was one of 3 institutions` ***`sent/RP`***
`letters containing white powder.`

In this sentence, the subject of *sent* (*3 institutions*) is not the theme of *sent*, but the recipient. The ditransitive verb, with its different argument structure, allows for a different syntactic transformation than we find with ordinary transitive verbs.

Though this was not a common problem, it does expose a shortcoming in our treatment of verbs' argument structures. Clausal-complement verbs may behave similarly; as I noted in Section 3.2.6, clausal complements can occur with active-voice verbs:

`Sam believed` ***`that dogs should be washed.`***

In this case, *believe* is in the active voice but does not take a direct object. Hoever, the requirement of a clausal complement means that the verb's argument structure is, like a ditransitive, different from what my experimental model expected.

CHAPTER 5

CONCLUSIONS

Approaching reduced passive voice recognition as a classification problem is promising as a technique for finding reduced passives in shallow parsing environments. While the Treebank-trained full parsers still perform better, many NLP systems use shallow parsers for reasons of speed or robustness and could benefit from this approach.

First, I summarize the benefits of this approach. Next, I discuss some potentially fruitful directions for future research to take, suggested by the error analysis section in the previous chapter.

5.1 Benefits of the classifier approach for reduced passive recognition

One of the chief benefits of the classifier approach for reduced passive recognition is that it uses a shallow parser rather than a full parser. While full parsers can achieve good reduced passive recognition, shallow parsers tend to be faster and more robust when given ungrammatical input. This makes shallow parsers more suitable for NLP tasks, such as information extraction, which involve processing large volumes of text which may be informally written or poorly formatted. Shallow parsers are not able to recognize reduced passives by themselves as full parsers can, so systems based on them may incorrectly assign thematic roles due to the effects that passive voice verbs have on the mapping of syntactic roles to thematic roles. The classification approach to reduced-passive recognition offers a solution to this problem for systems that use shallow parsing.

The classification approach has other benefits as well:

1. **Minimal training resources:** The existing Entmoot corpus automatically derived from the Penn Treebank seems to be an acceptable training set. However, even constructing a new training corpus of RP-annotated data would be inexpensive compared to sophisticated annotation efforts like the Penn Treebank and would require less advanced expertise on the part of the annotators. Furthermore, the

domain-specific corpora used to build the transitivity/thematic role knowledge base requires no annotation at all.

2. **High precision:** The classifier currently shows precision of 80% or higher in reduced passive classification, across each of the domains tested.

5.2 Future Work

The clear first priority for improvement is in recall, with current scores as low as 54% even from the best classifier. This should not come at the expense of precision, though, since our ultimate aim is to improve thematic role identification by increasing the accuracy of verb voice recognition.

5.2.1 Parser and data preparation improvements

In Section 4.5.4, I identified erroneous part-of-speech tags as a major problem. About 10% of reduced passive verbs are mistagged as parts of speech other than verbs, and are therefore never even considered to be candidate RP verbs. Most, about 90%, of these were mistagged as adjectives. Therefore, recall may be substantially improved by applying the classifier not only to words tagged as verbs but also to words ending in "-ed" that were tagged as adjectives. Adjectives of this type constituted over 90% of the reduced passives that were mistakenly rejected by the data preparation process. If feature vectors could be built for them reliably, the classifier could conceivably improve its recall by about 10% in each of the domains. In some ways this will be difficult, since some features depend on Sundance's labeling of nearby NPs as subjects or objects, which will be inaccurate with respect to words it does not consider verbs, so some workaround will be necessary.

Alternatively, it may be possible to correct this problem by improving Sundance's POS tagging or using a separate POS tagger that works better. Improvement of the support data that Sundance uses might also help; for instance, creating more detailed part-of-speech and semantic dictionaries for the existing domains.

5.2.2 Improvements to existing features

The semantic features are currently a handicap to the classifier, however justified they may seem in theory. It is possible that they would perform better if more detailed semantic dictionaries were created for Sundance.

The transitivity feature, though useful in its current form, might be improved if the knowledge base tool were able to distinguish different senses of verbs. For instance, the verb *to run* has two common senses, one of which is intransitive and one of which is transitive, illustrated in these sentences:

1. `The marathon racers` ***`ran`*** `yesterday.` (intransitive)
2. `Bill` ***`ran`*** `the restaurant for 40 years.` (transitive)

If the knowledge base and feature extraction accounted for verb sense, multiple-sense verbs could be more accurately classified.

The full feature set's thematic role features are crude in their current form. Word sense disambiguation might be useful for it as well. Further, it may help to count occurrences of low-level semantic types as agents and themes in addition to the top-level semantics currently used. Alternatively, a more sophisticated method of judging agent/theme likelihood for a verb given an NP's semantic class might help, such as the methods described in [30] and [5].

5.2.3 Additional features

The problems with ditransitive and clausal-complement verbs mentioned in Section 4.5.6, though not common, suggest that features addressing additional argument structures could be helpful. The tool that builds the Transitivity / ThemRole KB might also accumulate a clausal-complement rate, similar to the transitivity rate, for verbs. In general, methods for finding verbs' subcategorization frames, such as those describe in [36, 21], could be incorporated into this approach.

5.2.4 Improved choice or application of learning algorithm

It might be useful to attempt an ensemble solution: that is, have different classifiers trained on different subsets of the feature vector and let them "vote" on how to classify a given verb. However, the ablation tests in Section 4.5.1 suggest that this may not be very useful with the current feature vector. New features may improve the prospects for this kind of approach.

5.3 Future work summary and conclusion

The classifier approach to reduced passive recognition clearly has room to grow. In its present form it demonstrates that lexical knowledge about particular verbs, together with syntactic and semantic properties of verbs' context, can be used to distinguish reduced passives from other verbs. Supplementary knowledge gleaned from unannotated text improves its performance. Improved shallow parsing and more sophisticated supplementary knowledge may allow the classifier to achieve even better performance in the future.

APPENDIX A

ENTMOOT RULES

Tables A.1 and A.2 show the six rules used by Entmoot to recognize ordinary and reduced passives in Treebank-style parse trees:

Table A.1: Rules for finding reduced passives in Treebank parse trees.

1	- Parent and any nested ancestors are VPs - None of VP ancestors' preceding siblings is verb - Parent of oldest VP ancestor is NP Ex: "The man, it seems, has a Lichtenstein corporation, *licensed* in Libya and *sheltered* in the Bahamas."
2	- Parent is a PP Ex: "Coke introduced a caffeine-free sugared cola *based* on its original formula in 1983."
3	- Parent is VP and Grandparent is Sentence (clause) - Great-grandparent is clause, NP, VP, or PP Ex: "But there were fewer price swings than *expected*."
4	- Parent (and nested ancestors) is ADJP - None of oldest ADJP ancestor's preceding siblings is a determiner - None of oldest ADJP ancestor's following siblings is a noun or NP Ex: "Two big stocks *involved* in takeover activity saw..."

Table A.2: Rules for finding ordinary passives in Treebank parse trees.

1	- Parent is a VP - Starting with parent and climbing nested VP ancestors, the closest verb sibling before any VP ancestor is a passive auxiliary Ex: "He was *fined* $25,000."
2	- Parent (and nested ancestors) is ADJP - Oldest ADJP ancestor's parent is VP - Closest verb sibling before oldest ADJP ancestor is a passive auxiliary Ex: "The move had been widely *expected*."

APPENDIX B

FULL SEMANTIC HIERARCHY

Figure B.1 shows the complete semantic hierarchy for nouns that I used when conducting my experiments. Top-level semantic classes are shown in rectangular nodes, and low-level semantic classes in elliptical nodes.

Most of the classes are self-explanatory, though in the DISEASE_OR_ORGANISM subtree there are some abbreviations: ACQABN for "acquired abnormality", BIOACT for "bioactive substance", and RICKCHLAM for "rickettsia-chlamydia".

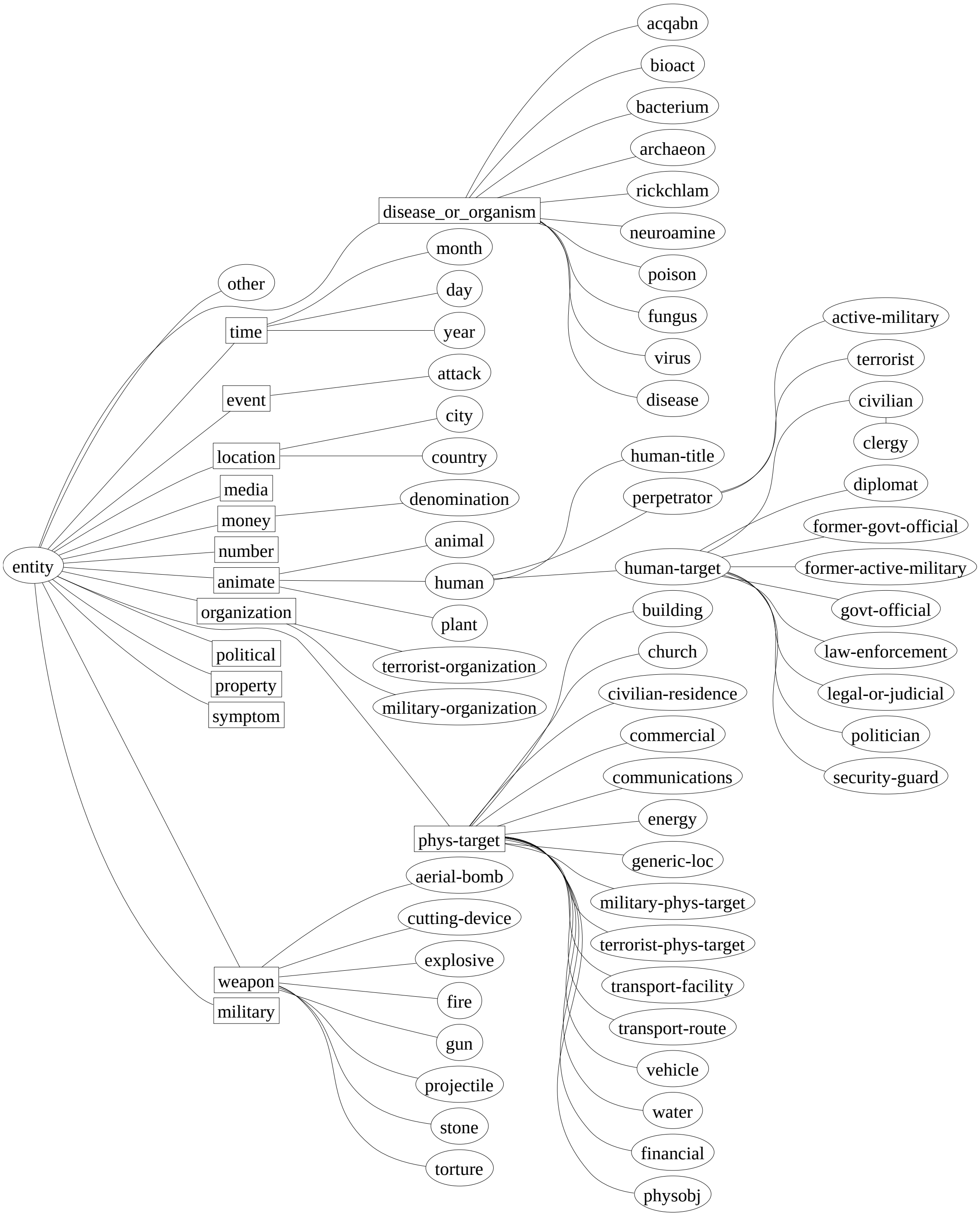

Figure B.1: Complete semantic hierarchy for nouns.

REFERENCES

[1] Abney, S. Partial parsing via finite-state cascades. Workshop on Robust Parsing, 8th European Summer School in Logic, Language and Information, Prague, Czech Republic, 1996.

[2] Baldridge, J., Morton, T., and Bierner, G. OpenNLP Maximum Entropy package and tools API, 2005.

[3] Bethard, S., Yu, H., Thornton, A., Hatzivassiloglou, V., and Jurafsky, D. Automatic extraction of opinion propositions and their holders. In *Computing Attitude and Affect in Text: Theory and Applications*. Springer, 2005.

[4] Bies, A., Ferguson, M., Katz, K., and MacIntyre, R. Bracketing guidelines for Treebank II style Penn Treebank Project. Technical Report, Department of Computer and Information Science, University of Pennsylvania, 1995.

[5] Brockmann, C., and Lapata, M. Evaluating and combining approaches to selectional preference acquisition. In *Proceedings of the Tenth Conference on European Chapter of the Association For Computational Linguistics, Volume 1* (Budapest, Hungary, 2003), pp. 27–34.

[6] Charniak, E. A maximum-entropy-inspired parser. In *Proceedings of the 2000 Conference of the North American Chapter of the Association for Computational Linguistics* (2000).

[7] Charniak, E., Goldwater, S., and Johnson, M. Edge-based best-first chart parsing. In *Proceedings of the Sixth Workshop on Very Large Corpora* (1998), pp. 127–133.

[8] Choi, Y., Cardie, C., Riloff, E., and Patwardhan, S. Identifying sources of opinions with conditional random fields and extraction patterns. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing* (2005), pp. 355–362.

[9] Collins, M. *Head-Driven Statistical Models for Natural Language Parsing.* PhD thesis, University of Pennsylvania, 1999.

[10] Gildea, D., and Jurafsky, D. Automatic labeling of semantic roles. *Computational Linguistics 28*, 3 (2002), 245–288.

[11] Haegeman, L. *Introduction to Government and Binding Theory.* Basil Blackwell Ltd, 1991.

[12] Haghighi, A., Toutanova, K., and Manning, C. A joint model for semantic role labeling. In *Proceedings of the Annual Conference on Computational Natural Language Learning (CoNLL)* (2005), pp. 173–176.

[13] Hobbs, J., Appelt, D., Bear, J., Israel, D., Kameyama, M., Stickel, M., and Tyson, M. FASTUS: A cascaded finite-state transducer for extracting information from natural-language text. In *Finite-State Language Processing*, E. Roche and Y. Schabes, Eds. MIT Press, Cambridge, MA, 1997.

[14] Joachims, T. Making large-scale SVM learning practical. In *Advances in Kernel Methods - Support Vector Learning*, B. Scholkopf, C. Burges, and A. Smola, Eds. MIT Press, Cambridge, MA, 1999.

[15] Kim, S., and Hovy, E. Extracting opinions, opinion holders, and topics expressed in online news media text. In *Proceedings of ACL/COLING Workshop on Sentiment and Subjectivity in Text* (2006).

[16] Lin, D. Dependency-based evaluation of MINIPAR. In *Proceedings of the LREC Workshop on the Evaluation of Parsing Systems* (Granada, Spain, 1998), pp. 48–56.

[17] Lin, D. LaTaT: Language and text analysis tools. In *Proceedings of the First International Conference on Human Language Technology Research* (2001).

[18] Marcus, M., Santorini, B., and Marcinkiewicz, M. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics 19*, 2 (1993), 313–330.

[19] Melli, G., Shi, Z., Wang, Y., Liu, Y., Sarkar, A., and Popowich, F. Description of SQUASH, the SFU Question Answering Summary Handler for the DUC-2006 summarization task. In *Proceedings of the Document Understanding Conference 2006 (DUC-2006)* (2006).

[20] Merlo, P., and Stevenson, S. What grammars tell us about corpora: the case of reduced relative clauses. In *Proceedings of the Sixth Workshop on Very Large Corpora* (Montreal, 1998), pp. 134–142.

[21] Merlo, P., and Stevenson, S. Automatic verb classification based on statistical distribution of argument structure. *Computational Linguistics 27*, 3 (2001), 373–408.

[22] Miller, G. WordNet: An on-line lexical database. *International Journal of Lexicography 3*, 4 (1991).

[23] MUC-4 Proceedings. Proceedings of the Fourth Message Understanding Conference (MUC-4). Morgan Kaufmann, 1992.

[24] Pado, U., Crocker, M., and Keller, F. Modelling semantic role plausibility in human sentence processing. EACL, Trento, 2006.

[25] ProMED-mail. http://www.promedmail.org/, 2006.

[26] Punyakanok, V., Roth, D., Yih, W., Zimak, D., and Tu, Y. Semantic role labeling via generalized inference over classifiers (shared task paper). In *Proceedings of the Annual Conference on Computational Natural Language Learning (CoNLL)* (2004), H. Ng and E. Riloff, Eds., pp. 130–133.

[27] Ratnaparkhi, A. A maximum entropy model for part-of-speech tagging. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-96)* (1996).

[28] Ratnaparkhi, A. A simple introduction to maximum entropy models for natural language processing. Technical Report 97-08, Institute for Research in Cognitive Science, University of Pennsylvania, 1997.

[29] Ratnaparkhi, A. *Maximum Entropy Models for Natural Language Ambiguity Resolution.* PhD thesis, University of Pennsylvania, 1998.

[30] Resnik, P. Selectional constraints: An information-theoretic model and its computational realization. *Cognition 61* (November 1996), 127–159.

[31] Riloff, E., and Phillips, W. An introduction to the Sundance and AutoSlog systems. Technical Report UUCS-04-015, School of Computing, University of Utah, 2004.

[32] Riloff, E., and Schmelzenbach, M. An empirical approach to conceptual case frame acquisition. In *Proceedings of the Sixth Workshop on Very Large Corpora* (1998), pp. 49–56.

[33] Sakai, T., Saito, Y., Ichimura, Y., Koyama, M., Kokubu, T., and Manabe, T. ASKMi: A Japanese question answering system based on semantic role analysis. In *RIAO-04* (2004).

[34] Sang, E., and Buchholz, S. Introduction to the CoNLL-2000 shared task: Chunking. In *Proceedings of CoNLL-2000 and LLL-2000* (Lisbon, Portugal, 2000).

[35] Stenchikova, S., Hakkani-Tur, D., and Tur, G. QASR: Question Answering using Semantic Roles for speech interface. In *Proceedings of ICSLP-Interspeech* (2006).

[36] Stevenson, S., Merlo, P., Kariaeva, N., and Whitehouse, K. Supervised learning of lexical semantic verb classes using frequency distributions. In *Proceedings of SigLex99: Standardizing Lexical Resources* (College Park, Maryland, 1999).

[37] Sudo, K., Sekine, S., and Grishman, R. An improved extraction pattern representation model for automatic IE pattern acquisition. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL-03)* (2003).

[38] Witten, I., and Frank, E. *Data Mining: Practical Machine Learning Tools and Techniques, 2nd Edition.* Morgan Kaufmann, San Francisco, 2005.

[39] Yi, S., and Palmer, M. The integration of syntactic parsing and semantic role labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL)* (2005).