Extraction of semantic relations from bioscience text

by

Barbara Rosario

GRAD. (University of Trieste, Italy) 1995

A dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Information Management and Systems

and the Designated Emphasis

in

Communication, Computation and Statistics

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:

Professor Marti Hearst, Chair
Professor John C. I. Chuang
Professor Dan Klein

Fall 2005

The dissertation of Barbara Rosario is approved:

Professor Marti Hearst, Chair	Date
Professor John C. I. Chuang	Date
Professor Dan Klein	Date

University of California, Berkeley

Fall 2005

Extraction of semantic relations from bioscience text

by

Barbara Rosario

Abstract

Extraction of semantic relations from bioscience text

by

Barbara Rosario

Doctor of Philosophy in Information Management and Systems

and the Designated Emphasis in Communication, Computation and Statistics

University of California, Berkeley

Professor Marti Hearst, Chair

A crucial area of Natural Language Processing is semantic analysis, the study of the meaning of linguistic utterances. This thesis proposes algorithms that extract semantics from bioscience text using statistical machine learning techniques. In particular this thesis is concerned with the identification of concepts of interest ("entities", "roles") and the identification of the relationships that hold between them. This thesis describes three projects along these lines.

First, I tackle the problem of classifying the semantic relations between nouns in noun compounds, to characterize, for example, the "treatment-for-disease" relationship between the words of *migraine treatment* versus the "method-of-treatment" relationship between the words of *sumatriptan treatment*. Noun compounds are frequent in technical text and any language understanding program needs to be able to interpret them. The task is especially difficult due to the lack of syntactic clues. I propose two approaches to this problem. Second, extending the work to the sentence level, I examine the problem of distinguishing among seven relation types that can occur between the entities "treatment" and "disease" and the problem of identifying such entities. I compare five generative graphical models and a neural network, using

lexical, syntactic, and semantic features. Finally, I tackle the problem of identifying the interactions between proteins, proposing the use of an existing curated database to address the problem of the lack of appropriately labeled data. In each of these cases, I propose, design and implement state-of-the art machine learning algorithms.

The results obtained represent first steps on the way to a comprehensive strategy of exploiting machine learning algorithms for the analysis of bioscience text.

Professor Marti Hearst, Chair Date

Acknowledgements

I would like to thank my advisor Marti Hearst, who helped and encouraged me throughout my graduate studies and who gave me the freedom to pursue my intellectual dreams.

I had a terrific learning and personal experience at SIMS, and I am grateful for this to all the professors (in particular to John Chuang), the wonderful staff and all my fellow students, especially Michal Feldman for being such a good friend.

I am very grateful to my dissertation committee, Marti Hearst, John Chuang and Dan Klein, for having helped me so insightfully and under such time-pressure, thanks! I want to thank Charles Fillmore and Chris Manning, as well, for being in my qualifying exams and because I learned so much from them.

I would like to thank Nu Lai, Kaichi Sung and for their help in labeling examples and other biological insights.

This work was supported in part by NSF award number IIS-9817353, by a grant from the ARDA AQUAINT program, NSF DBI-0317510, and a gift from Genentech.

My life would be much sadder without my dearest friends, Dino, Bruno, Jane, Francesca and Annalisa.

My families: my Italian family, my mother, sisters Nicoletta and Rossana and little Riccardo that I am going to meet tomorrow for the first time; my American family, Carolyn, Carol, Brad and little Nico: thanks for your love and support. My father, who taught me a lot (despite himself) and whom (for the better or the worse) I look so much alike.

Lara, my love, for making me so happy (and sometimes unhappy), like anyone else. But above all, Michael, for your love, support, help, humor and patience.

To Michael and Lara.

E papa'.

Contents

Chapter 1

Introduction

1.1 Extracting semantics from bioscience text

A crucial area of Natural Language Processing is semantic analysis, the study of the meaning of linguistic utterances. This thesis is part of a larger effort to investigate what can be called "statistical semantic parsing," that is, the attempt to extract semantics from text and to build a knowledge representation of the concepts expressed in the text, using statistical machine learning techniques (for an overview, see Grishman, 1986).

This thesis proposes algorithms that extract semantics in terms of "entities" and "relations" from bioscience text. As an example, I envision a system that when asked the question "What are the treatments of cervical carcinoma" identifies the following sentence as containing an answer: *"Stage Ib and IIa cervical carcinoma can be cured by radical surgery or radiotherapy"* and extracts the text strings *radical surgery* and *radiotherapy* to provide the specific answers. Similarly, I envision a system that returns *"intranasal migraine treatment"* to a question such as "What are the methods of administration of headache treatment," or that, when given the question "What

are the side effects of extracorporeal circulation in infants?," extracts *hearing loss* from the following sentence: *"Neonatal BAEP threshold recordings were of limited value for predicting subsequent hearing loss common in ECMO-treated survivors."*

Such a system needs to engage in a form of inductive reasoning to infer that *migraine* and *headache* are closely related, that *radiotherapy* is a treatment, that *ECMO* is a type of extracorporeal circulation and that *neonatal* is used to talk about infants. It needs to understand what the semantic relations are that hold between the concepts (to distinguish between a *cure* kind of relation in the "migraine" sentence and a *side effect* in the "neonatal" sentence) and it needs to identify the specific strings of text that contain the answers to the queries.

Some of the core issues of "understanding language" are the identification of concepts of interest and the identification of the relationships that hold between them. In this thesis I address these problems.

I position my work to bridge the fields of statistical machine learning and biomedical natural language processing. I propose state-of-the art machine learning algorithms; in particular, I design and implement graphical models specifically for the problems tackled. Graphical models (Jordan, 2004) represent a marriage between probability theory and graph theory. They are playing an increasingly important role in the machine learning field. I propose applying these models (and others) to practical but very difficult and important problems in the bioscience domain.

I have chosen the bioscience application area for several reasons. First, very useful applications can be developed in this domain; as an example, the identification of the interactions between proteins is one of the most important challenges in modern biology; thousand of articles are published every year on this subject, most of which are available electronically but only in unstructured text format. Automatic mechanisms are needed to extract the information from these articles, to help researchers find what they need, but also more ambitiously, to make inferences about propositions

that hold between scientific concepts.

Another reason for tackling bioscience text is that it may be easier to process automatically than ordinary text. It is less ambiguous, and the concepts and processes it describes are more "mechanical" and therefore easier to represent by a computer.

Finally, there are many resources for this domain that we can exploit, including journal articles, domain-specific ontologies and manually curated databases.

1.2 Outline of this thesis

This thesis is structured as follows:

- In Chapter 2 I describe two approaches that I developed for classifying the semantic relations between nouns in noun compounds, to characterize, for example, the "treatment-for-disease" relationship between the words of *migraine treatment* versus the "method-of-treatment" relationship between the words of *sumatriptan treatment*. Noun compounds are very frequent in technical text and any automatic parsing program needs to be able to interpret them. The task is especially difficult due to the lack of syntactic clues. In the first approach, I propose a classification algorithm that achieves accuracies as high as 62% for the classification of noun compounds into one out of eighteen semantic relations. The second approach is linguistically motivated and explores the use of a lexical hierarchy for the purpose of placing words from a noun compound into categories, and then using this category membership to determine the relation that holds between the nouns.

  Most related work relies on hand-written rules of one kind or another (Finin, 1980; Vanderwende, 1994; Rindflesch et al., 2000b) or tackles easier problems (Lapata, 2000, for example, addresses a binary classification problem).

- In Chapter 3 I examine the problem of distinguishing among seven relation types that can occur within a sentence between the entities "treatment" and "disease," as well as the problem of identifying such entities (role extraction). I compare five generative graphical models and a neural network, using lexical, syntactic, and semantic features.

  While there is much prior work on role extraction, little work has been done for relationship recognition. Moreover, many papers that claim to be doing relationship recognition in reality address the task of role extraction: (usually two) entities are extracted and the relationship is *implied* by the co-occurrence of these entities or by the presence of some linguistic expression (Agichtein and Gravano, 2000; Zelenko et al., 2002); in the related work there is, to the best of my knowledge, no attempt to distinguish between *different* relations that can occur between the *same* semantic entities. Moreover, most of the related work on relationship extraction assumes the entity extraction task is performed by another system and the entities of interests therefore are given as input. The models I propose in this thesis do not make this assumption and indeed perform role and relation extraction simultaneously.

- In Chapter 4 I apply the statistical models proposed in Chapter 3 to another important application, the identification of the interactions between proteins in bioscience text. A major impediment to such work (and in general to the development of many statistical methods) is the lack of appropriately labeled data. Labeling is a very time-consuming and subjective process; moreover, especially for "semantic tasks," different sets of labeled data are needed for each domain and perhaps for each application. I propose the use of existing curated database, the HIV-1 Human Protein Interaction Database to serve as a proxy for training data.

In the bioscience domain there have recently been many attempts to automatically extract protein-protein interactions, however, the work on relation classification is primary done through hand-built rules. Some approaches simply report that a relation exists between two proteins but do not determine which relation holds (Bunescu et al., 2005; Marcotte et al., 2001; Ramani et al., 2005), while most others start with a list of interaction verbs and label only those sentences that contain these trigger verbs (Blaschke and Valencia, 2002; Blaschke et al., 1999a; Thomas et al., 2000; Ahmed et al., 2005; Phuong et al., 2003; Pustejovsky et al., 2002).

In this thesis, the statistical methods proposed determine the interaction *types* and do not use trigger words explicitly.

These projects constitute a significant step toward the goal of extracting propositional information from text.

Chapter 2

Noun Compounds

2.1 Introduction

One of the important challenges of biomedical text, along with most other technical text, is the proliferation of noun compounds. A typical article title is shown below; it consists of a cascade of four noun phrases linked by prepositions:

> *Open-labeled long-term study of the efficacy, safety, and tolerability of subcutaneous sumatriptan in acute migraine treatment.*

A language understanding program needs to be able to interpret the noun compounds (NCs) in order to ascertain sentence meaning. NCs present challenges for natural language processing, such as syntactic attachment and semantic interpretation. I argue that the real concern in analyzing such a title is in determining the relationships that hold between different concepts, rather than on finding the appropriate attachments.[1] For example, we want to characterize the "treatment-for-disease" rela-

[1] Or at least this is the first step that must be taken. If we can determine the semantic relation between the nouns in a two-word noun compound, and if we also know how to parse longer noun compounds, then we can fully interpret them (assuming also that we know how to combine the meanings of the multiple relations).

tionship between the words of *migraine treatment* versus the "method-of-treatment" relationship between the words of *sumatriptan treatment*. These relations are intended to be combined to produce larger propositions that can then be used in a variety of interpretation paradigms, such as abductive reasoning or inductive logic programming.

Interpretation of noun compounds is highly dependent on lexical information; it is the meaning of the words *migraine* and *sumatriptan* that determines their different relations with the word *treatment*.

I present here two lines of research that tackle this problem. In the first approach (Rosario and Hearst, 2001), I describe a classification algorithm for identifying the types of possible relationships between two-word noun compounds. In the second approach (Rosario et al., 2002) I use a lexical hierarchy and show that mere membership within a particular sub-branch of the hierarchy is sufficient in many cases for assignment of the appropriate semantic relation.

The remainder of this chapter is organized as follows: Section 2.2 discusses noun compounds from a linguistic point of view, Section 2.3 describes my collection of NCs in the biomedical domain, as well as the semantic relations that I identified for this collection. Section 2.4 describes in some detail two linguistic theories of the semantics of NCs and shows how theses theories are not appropriate for my collection and for the relation identification task as I define it. Section 2.5 discusses related work and finally in Sections 2.6 and 2.7 I describe the two different approaches I propose for the semantic classification of NCs.

2.2 Noun compounds in the linguistic literature

There are a multitude of possible definitions for NCs. The most popular are (from Lauer, 1995b):

1. **Noun premodifier**: Any constituent can appear before a noun to form a NC: *out-in-the-wilds cottages* is therefore considered a NC.

2. **Compounds** (phonological definition due to Chomsky, for English compounds): words preceding a noun form a compound if they receive primary stress, thus *blackboard* is a compound, while *black board* is not.

3. **Complex Nominals**: Levi (1978) chooses to include certain adjectives along with nouns as possible compounding elements that she calls Complex Nominals. The adjectives she includes are non-predicative adjectives as in *electrical engineer* or *presidential refusal*.

4. **Noun Noun Compounds**: any sequence of nouns that itself functions as a noun.

5. **Noun Noun Compounds**: any sequence of nouns at least two words in length that itself functions as a noun, but which contains no genitive markers and is not a name.

In my work I use definition 4 (but I do not consider compounds with genitive markers) and I do not include dvandva compounds.[2]

We can think of five different structures for NCs, at least for the English language (Warren, 1978):

1. **Morphological Structure**

   Warren says that the most important point is that, as a rule, the first noun

[2]Dvandva compounds are compounds with hyphens, such as *poet-painter*. Warren (1978) says that these compounds are different from the others in that the modifier does not predicate something about the second constituent, nor does it modify the meaning of the second constituent. The motivation for combining *poet* and *painter* in *poet-painter* is different from the motivation of combining *girl* and *friend* in *girlfriend*. In the former case, we want to convey that someone is not only a *poet* but also a *painter*, and we are not trying to identify which *painter*, or to define what kind of *painter* we are talking about; in the latter case, we want to make our reference more precise: *poet-painter* has an expansion of reference scope, while *girlfriend* narrows the reference scope.

does not take inflectional endings, such as plural or genitive suffices, but then she lists exceptions to this rule: plural first-words (*sports center, narcotics law*) and first-words with genitive -s (*driver's seat, women's colleges*).[3]

2. **Syntactic Structure**

   Warren discusses several ways of parsing NCs with more than two constituents (that she calls compounds-within-compounds) and shows the frequency and distributions of various combinations. For example *[[[spider leg] pedestal] table]* is left-branching within a left-branching structure, whereas *[home [workshop tool]]* is right-branching within a left-branching structure.

3. **Phonological Structure**

   Two fundamental stress patterns are identified: fore-stress (or first-stress) and double-stress. Fore-stress involves primary stress on the first noun and secondary stress on the second noun (*cowboy*). Double-stress involves heavy stress on both the first and second constituents: *stone wall*. Warren observes that certain semantic relations are connected with certain stress patterns and that stress may be used to signal syntactic and semantic differences: for example, compounds expressing material-artifact are usually pronounced with double stress (*bronze screws, brass wire*) and compounds expressing Purpose or Origin are pronounced with fore-stress.

4. **Information Structure**

   Elements in a sentence are not equally important. Some elements are, communicatively, more important than others and have therefore a different degree of "communicative dynamism." Warren suggests that the same analysis is possi-

[3]Warren includes a short discussion about the controversy regarding such constructs and she identifies the semantic relations of these compounds as either Purpose (*driver's seat*) or Possession (*pastor's cap*).

ble for NCs and that the two constituents of a NC do not normally have equal informative force, and that in NCs the first position is reserved to the element with the highest degree of communicative dynamism.

5. **Semantic Structure**

NCs can be described as having a bi-partite structure, in that they consist of a topic and a comment (Warren, 1978). The topic is what is being talked about (represented by the head of the NC) and the comment is what is said about the comment. The function of the comment is to make the reference of the topic more precise. It restricts the reference scope of the topic by either classifying the intended topic, or by identifying which of a number of known topics are being referring to. The first case refers to NCs in which the comment has *classifying function* (paraphrasable by: "that kind of -topic- that (verb) -comment-"), as in *bilge water* (that kind of water that comes from bilge). The second case refers to NCs in which comment has *identifying function* (paraphrasable by: "that -topic- that (verb) (some specific) -comment-"), as in *bathroom door* (the door that is attached to the bathroom). In other words, in the first case the comment specifies *which type* of topic, in the second case, it specifies *which instance* of the topic.

Within the category of semantic structure, Warren lists **Idiomatization**, **Established** (those combinations that have become the generally accepted word for their referent, like *toothpaste* instead of *mouth hygiene paste*) and **Novel Compounds** (combinations that have been created for the occasion) and finally **Participant Roles** in which the constituents of the NC are semantically related but there are several different semantic relations that can hold between them. It is indeed the purpose of Warren (1978) to establish whether or not

there is a limited number of possible types of semantic relations between the constituents of compounds, and what these semantic relations are. According to Warren, there are six major types of semantic relations than can be expressed in compounds: Constitute, Possession, Location, Purpose, Activity-Actor, Resemblance. In Section 2.4.2 I describe in more detail Warren's semantic relations for NCs.

There are other important issues about NCs such as lexicalization, linguistic function, grammar (nominalizations vs. non-verbal compounds), and ambiguity. The semantic structure of NCs is the focus of this chapter.

2.3 Noun compound collection and relations

To create a collection of noun compounds, I performed searches from MEDLINE, which contains references and abstracts from thousands of biomedical journals.[4] I used several query terms, intended to span across different subfields. I retained only the titles and the abstracts of the retrieved documents. On these titles and abstracts I ran a part-of-speech tagger (Brill, 1995) and a program that extracts those sequences of adjacent tokens tagged as nouns by the tagger (constituents).

As an example, the following sentence was extracted from an abstract of a medical article:

```
The best multivariate predictors of influenza infections were
cough and fever with a positive predictive value of 79%.
```

[4]MEDLINE is the NLM's premier bibliographic database covering the fields of medicine, nursing, dentistry, veterinary medicine, the health care system, and the preclinical sciences. MEDLINE contains bibliographic citations and author abstracts from more than 4,800 biomedical journals published in the United States and 70 other countries. The database contains over 12 million citations dating back to the mid-1960's. Coverage is worldwide, but most records are from English-language sources or have English abstracts.
http://www.nlm.nih.gov/pubs/factsheets/medline.html.

The tagger returns the following tags (the format being "word_TAG"):

```
The_DT best_JJS multivariate_JJ predictors_NNS of_IN
influenza_NN infections_NNS were_VBD cough_NN and_CC
fever_NN with_IN a_DT positive_JJ predictive_JJ value_NN
of_IN 79_CD %_NN ._.
```

from which I extract ‘‘`influenza infections`’’ because the tags of these words are both nouns (one singular and one plural).

I extracted NCs with up to 6 constituents, but for this work I consider only NCs with 2 constituents.

For this collection of NCs, I manually developed a set of semantic relations that hold between the constituent nouns. In this work I aim for a representation that is intermediate in generality between standard case roles (such as Agent, Patient, Topic, Instrument, Fillmore, 1968, 1977), and the specificity required for information extraction. I have created a set of relations, shown in Table 2.1, that are sufficiently general to cover a significant number of noun compounds, but that can be domain-specific enough to be useful in analysis.

The problem remains of determining what the appropriate relations are. I wanted to support relationships between entities that are shown to be important in cognitive linguistics. In theoretical linguistics, there are contradictory views regarding the semantic properties of noun compounds. As described in Section 2.4, Levi (1978) argues that there exists a small set of semantic relationships that NCs may imply. In contrast, Dowing (1977) argues that the semantics of NCs cannot be exhausted by any finite listing of relationships. Between these two extremes lies Warren's taxonomy of six major semantic relations organized into a hierarchical structure (Warren, 1978).

I tried to produce relations that correspond to linguistic theories such as those of Levi and Warren, but in many cases these are inappropriate. Levi's classes are

too general for my purposes; for example, she collapses the "location" and "time" relationships into one single class "In" and therefore *field mouse* and *autumnal rain* belong to the same class. Warren's classification schema is much more detailed, and there is some overlap between the top levels of Warren's hierarchy and my set of relations. For example, my "Cause (2-1)" for *flu virus* corresponds to her "Causer-Result" of *hay fever*, and my "Person Afflicted" (*migraine patient*) can be thought as Warren's "Belonging-Possessor" of *gunman*. Warren differentiates some classes also on the basis of the semantics of the constituents, so that, for example, the "Time" relationship is divided up into "Time-Animate Entity" of *weekend guests* and "Time-Inanimate Entity" of *Sunday paper*. (These issues are described in more detail in Section 2.4.)

My classification is based on the kind of relationships that hold between the constituent nouns rather than on the semantics of the head nouns. By inspecting the examples, I identified the 38 relations shown in Table 2.1.

For the automatic classification task described in Section 2.6, I used only the 18 relations (indicated in bold in Table 2.1) for which an adequate number of examples were found in the NC collection. Many NCs were ambiguous, in that they could be described by more than one semantic relationship. In these cases, I simply multi-labeled them: for example, *cell growth* is both "Activity" and "Change," *tumor regression* is "Ending/reduction" and "Change" and *bladder dysfunction* is "Location" and "Defect." My approach handles this kind of multi-labeled classification.

Two relation types are especially problematic. Some compounds are non-compositional or lexicalized, such as *vitamin k* and *e2 protein*; others defy classification because the nouns are subtypes of one another. This group includes *migraine headache*, *guinea pig*, and *hbv carrier*. I placed all these NCs in a catch-all category. I also included a "wrong" category containing word pairs that were incorrectly labeled as NCs.[5] (I did

[5]The percentage of the word pairs extracted that were not true NCs was about 6%; some examples

not remove them because I envision my classification system as part of a pipeline: if my system can recognize that certain NCs are "wrong", then we could try to recover from the erroneous tagging. In any case, keeping the "wrong" class implies that we have an additional class, and should we have a perfect tagger, we would not need this class any more and the classification problem would be easier.)

The relations were found by iterative refinement based on looking at 1880 extracted compounds (described in the next section) and finding commonalities among them. Labeling was done by the author and by a biology student. The NCs were classified out of context, thus making resolution of ambiguities difficult in some cases.

These relations are intended to be combined into larger propositions. For example, consider *migraine treatment*, to which I have assigned the relation "Purpose." This can be described by the proposition *Purpose(treatment, migraine)*. We can then add semantic information for the two consituent nouns: migraine is a Disease and treatment falls into Therapeutic Techniques (see Section 2.6). This allows us to infer the following proposition: *Purpose(Therapeutic Techniques, Disease)*; we can also use various degree of generality, and decide, for example, that we want to be more specific and describe migraine as a Nervous System Disease instead: we would have then *Purpose(Therapeutic Techniques, Nervous System Diseases)*. A representation such as this allows for flexibility in the detail of the semantics. The end goal is to combine these relationships in NCs with more than two constituent nouns, as in the example *intranasal migraine treatment*.

The collection of NCs is publicly available at:
http://biotext.berkeley.edu/data/nc_data.html.

Section 2.4 below describes in detail some linguistic theories. The reader not interested in a linguistic discussion can go directly to Section 2.5. I briefly summarize

are: *treat migraine*, *ten patient*, *headache more*. I do not know, however, how many NCs I missed. The errors occurred when the wrong label was assigned by the tagger.

Name	N	Examples
Wrong parse (1)	109	exhibit asthma, ten drugs
Subtype (4)	393	hbv carrier, t1 tumour
Activity/Physical process (5)	59	bile delivery, virus reproduction
Ending/reduction	8	migraine relief, headache resolution
Beginning of activity	2	headache induction, headache onset
Change	26	papilloma growth, tissue reinforcement
Produces (7)	47	polyomavirus genome, actin mrna
Cause (1-2) (20)	116	asthma hospitalizations, aids death
Cause (2-1)	18	flu virus, influenza infection
Characteristic (8)	33	receptor hypersensitivity, cell immunity
Physical property	9	blood pressure, artery diameter
Defect (27)	52	hormone deficiency, csf fistulas
Physical Make Up	6	blood plasma, bile vomit
Person afflicted (15)	55	aids patient, headache group
Demographic attributes	19	childhood migraine, infant colic
Person/center who treats	20	headache specialist, children hospital
Research on	11	asthma researchers, headache study
Attribute of clinical study (18)	77	headache parameter, attack study
Procedure (36)	60	tumor marker, brain biopsy
Frequency/time of (2-1) (22)	25	headache interval, influenza season
Time of (1-2)	4	morning headache, weekend migraine
Measure of (23)	54	asthma mortality, hospital survival
Standard	5	headache criteria, society standard
Instrument (1-2) (33)	121	aciclovir therapy, laser irradiation
Instrument (2-1)	8	vaccine antigen, biopsy needle
Instrument (1)	16	heroin use, internet use, drug utilization
Object (35)	30	bowel transplantation, kidney transplant
Misuse	11	drug abuse, acetaminophen overdose
Subject	18	headache presentation, glucose metabolism
Purpose (14)	61	headache drugs, hiv medications
Topic (40)	38	vaccination registries, health education
Location (21)	145	brain artery, tract calculi, liver cell
Modal	14	emergency surgery, trauma method
Material (39)	28	formaldehyde vapor, aloe gel, latex glove
Bind	4	receptor ligand, carbohydrate ligand
Activator (1-2)	6	acetylcholine receptor, pain signals
Activator (2-1)	4	headache trigger, headache precipitant
Inhibitor	11	adrenoreceptor blockers
Defect in Location (21 27)	157	lung abscess, artery aneurysm, brain disorder
Total number of labeled NCs	**1880**	

Table 2.1: The semantic relations defined via iterative refinement over a set of noun compounds. The relations shown in boldface are those used in the experiments described in Section 2.6. Relation ID numbers (used in Section 2.6) are shown in parentheses by the relation names. The second column shows the number of labeled examples for each class; the last row shows a class consisting of compounds that exhibit more than one relation. The notation (1-2) and (2-1) indicates the directionality of the relations. For example, Cause (1-2) indicates that the first noun causes the second, and Cause (2-1) indicates the converse.

the discussion of Section 2.4 at the beginning of next section.

2.4 Semantic theories in the linguistic literature

As mentioned in Section 2.3, I would like my classification schema for the NC relations to correspond to linguistic theories. In this section, I describe some of these theories and show how they are not appropriate for my collection of NCs.

The semantic properties of NCs have been hotly debated in linguistics, with numerous contradictory views being proposed. On one end of the scale, Levi (1978) suggests that that there exists a *very* small set of possible semantic relationships that NCs may imply. In contrast, Dowing (1977) performed a series of psychological experiments and concluded that the semantics of NCs cannot be exhausted by any finite listing of relationships (however, she acknowledges that some relations are more common than others). Between the two ends of this scale, there exists a range of semantic theories. Lauer (1995b) lists the following in order of the degree to which they constrain the possible semantics: Leonard (1984); Warren (1978); Finin (1980).

From a computational point of view, if noun compound interpretation is entirely pragmatic and context-dependent, as Dowing (1977) implies, then our task is indeed very difficult, if not hopeless. If on the other hand, it is the case that there is a finite small set of possible relations, then we have a basis for computational interpretation.

In the following subsections, I analyze two such theories: Levi (1978) and Warren (1978). Levi (1978) is cited in virtually every paper on computational analysis on NCs; her theory that all NCs can express only a very small set of semantic relations (13) is very attractive from a computational point of view. It is also notable that her theory derives from a theory of language and not from empirical investigation. I decided to analyze Warren (1978) for exactly the opposite reason, that is, her work *"is primarily a report of the results of an empirical investigation. In other*

words, my main interest has been directed towards finding out facts about the semantic patterns of a particular type of compound, rather than towards the question of how to account for the compounding process in accordance with a particular theory of language" (Warren, 1978). Warren's approach is very similar to mine: she analyzed 4557 different compounds (while Levi's collection consists only of about 200 examples) and classified them according to the covert semantic relations they expressed.

(Note that both Levi and Warren also include orthographically joined morphemes such as *gunboat* for Levi and *armchair, seafood* for Warren. They do not differentiate the semantics of continuous compounds vs. the semantics of compounds of separate words. I consider only sequences of separate nouns, assuming that the words in orthographically joined morphemes are joined because they are sufficiently common to warrant inclusion in the lexicon, and thus do not require dynamic processing.)

2.4.1 Judith N. Levi: The Syntax and Semantics of Complex Nominals

From page 5 of Levi (1978):

> *"The principal purpose of this book is the exploration of the syntactic and semantic properties of complex nominals (CNs) in English, and the elaboration of detailed derivations for these forms within a theory of generative semantics. The book also represents an attempt to incorporate into a grammar of English a model of the productive aspects of complex nominal formation".*

The emphasis is therefore on the **formation** of NCs. On page 6 Levi claims:

> *"One of the most important claims of the present work is that all CNs must be derived by just two syntactic processes: predicate nominalization*

> *and predicate deletion. The analysis of CNs derived by the latter process will entail the proposal of a set of Recoverably Deletable Predicates (RDPs) representing the only semantic relations which can underlie CNs. This set whose members are small in number, specifiable, and probably universal, consists of these nine predicates: CAUSE, HAVE, MAKE, USE, BE, IN, FOR, FROM and ABOUT."*

The analysis of nominalization brings out four types of nominalized verbs (AGENT, PATIENT, PRODUCT and ACT) derived from two types of underlying structures (Objective and Subjective).

The claims are large: only two syntactic processes and a very small set of universal relationships. Levi suggests that her principles are likely to reflect linguistic universals that constrain the use and interpretation of NCs in *all* human languages. Moreover, she dismisses ambiguity as being resolved by semantic, lexical and pragmatic clues: *"The fact that a given CN may be derived by the deletion of any of these predicates, or by the process of nominalization, means that CNs typically exhibit multiple ambiguity. This ambiguity is, however, reduced to manageable proportions in actual discourse by semantic, lexical and pragmatic clues"*. This could be achievable, in theory, by automatic systems, but in practice this is very hard to to do (and indeed ambiguity resolution is a core, unsolved problem of computational linguistics).

2.4.1.1 Derivation of CNs by Predicate Deletion

Levi claims that the variety of semantic relationships between head nouns and prenominal modifiers in NCs is confined within a very limited range of possibilities. A first derivation of CNs by predicate deletion is proposed to account for the larger part of these possible semantic relationships for those NCs whose two elements are derived from the two arguments of an underlying predicate, as in *apple pie* or *malarial*

mosquitoes. This small set of specifiable predicates that are recoverably deletable in the process of CN formation is made up of nine predicates. These predicates, and only these predicates, may be deleted in the process of transforming an underlying relative clause construction into the typically ambiguous surface configuration of the CN.

The NCs derived by predicate deletion are derived by a sequence of transformation in which a predicate was deleted to form a NC: for example, for *viral infection* (CAUSE) we have the following set of transformations:

virus causes infection
infection is caused by virus
infection is virus-caused
infection virus-caused
virus-caused infection
viral infection

Similarly, *marginal note* was derived by *note in margin*. Examples of CNs derived by deletion of these nine predicates are given in Table 2.2.

2.4.1.2 Derivation of CNs by Predicate Nominalization

The second major group is composed of CNs whose head noun is a nominalized verb, and whose prenominal modifier is derived from either the underlying subject or the underlying direct object of this verb.

Levi classifies the CNs first according to categories determined by the head noun, and second according to categories determined by the prenominal modifier. For the first case, Levi proposes four nominalization types: **Act Nominalizations** (*parental refusal, birth control*), **Product Nominalizations** (*musical critiques, royal orders*),

RDP	**N1 is direct object of relative clause**	**N1 is subject of relative clause**
CAUSE (causative)	**CAUSE1** tear gas disease germ malarial mosquitoes traumatic event	**CAUSE2** drug death birth pains nicotine fit viral infection
MAKE (productive, compositional)	**MAKE1** honeybee silkworm musical clock sebaceous glands	**MAKE2** daisy chains snowball consonantal patterns molecular chains
HAVE (possessive, dative)	**HAVE 1** picture book apple cake gunboat musical comedy	**HAVE 2** government land lemon peel student power reptilian scales
USE (instrumental)	voice vote steam iron manual labor solar generator	
BE (essive, appositional)	soldier ant target structure professorial friends mammalian vertebrates	
IN (locative-spatial or temporal)	field mouse morning prayers marine life marital sex	
FOR (purposive, benefactive)	horse doctor arms budget avian sanctuary aldermanic salaries	
FROM (source, ablative)	olive oil test-tube baby apple seed rural visitors	
ABOUT (topic)	linguistic lecture tax law price war abortion vote	

Table 2.2: Levi's classes of NCs derived by deletion of predicates. RDP stands for Recoverably Deletable Predicate. In parenthesis, more traditional terms for the relationships expressed by the RDPs.

Agent Nominalizations (*city planner, film cutter*) and **Patient Nominalizations** (*student inventions, mammalian secretions*).

Act Nominalizations can be paraphrased by "the act of (parents refusing)" while Product Nominalizations seem to represent some object (usually though not always tangible) that is produced as the result of a specific action or event (outcome) and can be paraphrased by "that which is produced by the act of (criticizing music)." To clarify the distinction between Product Nominalizations and Patient Nominalizations, Levi presents the following examples, in which (a) are products and (b) patients:

(a) The managerial appointments infuriated the sales staff.
(b) The managerial appointees infuriated the sales staff.
(a) We expect to receive the senatorial nominations shortly.
(b) We expect to receive the senatorial nominees shortly.

CNs whose head nouns are derived by nominalization (NOM CNs) may also be classified according to the syntactic source of their prenominal modifier(s). In this way the data may be divided into two major categories: **Subjective NOM CNs**, whose prenominal modifier derives from the underlying subject of the nominalized verb and **Objective NOM CNs**, whose prenominal modifier derives instead from that verb's underlying direct object (see Table 2.3). This classification is based on the meaning and source of the head noun alone and on the syntactic source of their prenominal modifier; it does not depend on the semantic relationships between the two nouns.

2.4.1.3 Exclusion of the preposition "OF"

I was surprised by Levi's exclusion of the preposition "OF" from the set of recoverably deletable predicates, especially because this rule seemed to be the reason I was unable

	SUBJECTIVE NOM CNs	**OBJECTIVE NOM CNs**
Act	parental refusal manager attempt scell decomposition judicial betrayal	birth control heart massage subject deletion musical criticism
Product	clerical error peer judgments faculty decisions papal appeals	oceanic studies chromatic analyses stage designs tuition subsidies
Agent	-	draft dodger urban planner acoustic amplifier electrical conductor
Patient	royal gifts student inventions presidential appointees city employees	-

Table 2.3: Levi's classes of NCs derived by nominalizations. The modifier derives from the underlying subject or object of the nominalized verb.

to classify about 250 of my NCs. On page 97, Levi (1978) lists some examples of CNs derived by FOR deletion (*bird sanctuary* from *sanctuary FOR birds*, *nose drops* from *drops FOR nose*) and on page 95, examples of CNs derived by IN deletion (*office friendships* from *friendships IN the office*, *marginal note* from *note IN the margin*). It's not clear to me why she excludes CNs derived by OF deletion: *headache activity* from *activity OF headache*, *brain function* from *function OF the brain*, *cell growth* from *growth OF the cell*. On page 161, she discusses the membership requirements for the RDP set but she mainly defends her choice for the nine predicates rather than explaining the reasons for the exclusion of others (she never explicitly mentions the preposition "of").

Levi claims that the predicates she proposes *"seem to embody some of the most rock-bottom-basic semantic relationships expressible in human language; assuming that there are such things as semantic primes, these nine are surely outstanding candi-*

dates." Levi goes on saying that all the nine predicates manifest "surface invisibility" and "grammaticality." She defines "surface invisibility" as a characterization of those predicate *"that can be either deleted or incorporated before they reach the surface without leaving a surface trace."* So for example the sign *employees only* means *FOR employees only*, *Adar is as talented as Ariella* means *Adar is as talented as Ariella IS TALENTED* and *Max wants a lollipop* means *Max wants (TO HAVE) a lollipop.*

One example comes to mind regarding the proposition "of": in a medical laboratory with plates containing biopsies, a label *bowel* would mean *biopsy OF bowel*, and the label *brain* would stand for *biopsy OF brain.* The notion of grammaticality means that these predicates *"are often expressed not by independent lexical items but by bound grammatical morphemes."* For example, some predicates in the RDP set are grammatized into marking of nouns, i.e., into case endings. For example, in the Proto-Indo-European system, CAUSE and FROM act as ablative, FOR as dative, USE as instrumental, IN as locative, MAKE as accusative, BE as nominative and HAVE as possessive genitive. It is important to note that in Levi (1978) the genitive case has been described only as possessive genitive that can indeed be paraphrased by HAVE.[6] I argue, however, the genitive is not used only to show possession but can also be indefinite, objective, partitive, descriptive in which cases the preposition is OF. For example:[7]

- genitive as charge: *He is guilty of murder*
- genitive as indefinite: *A box of [some] weight*
- genitive as description: *He was a man of no character*
- genitive as material: *A statue made of silver*

[6]And this was in fact the case for the classes "Characteristic of," "Physical property," "Defect" for which both HAVE2 and OF are good descriptions

[7]From Latin Dictionary and Grammar Aid: http://www.nd.edu/ archives/gen.htm

- genitive as objective: *He had no fear of death*
- genitive as partitive: *One out of a million*

Thus I would think of *bowel biopsy* and *acetaminophen overdose* as objective genitives and of *asthma mortality* as a description genitive.

For these reasons, I argue that the exclusion of "OF" is not well motivated.

2.4.1.4 Problems with Levi's classification system

In this section I show how Levi's classification system is not appropriate for the classification of my collection of NCs in the biomedical domain. (See Section 2.6.2 and Table 2.1 for a more detailed description of my classification.)

There are two main reasons why Levi's classification is inappropriate: 1) for many of the NCs in my collection there is no appropriate class under Levi's schema and 2) there is a many-to-many mapping between Levi's and my classification: NCs with the same semantic relation belong to different classes under Levi's classification and the same class under Levi's schema correspond to several classes of my schema. In other words, in certain cases, Levi's classification does not have the specificity required for the NCs of my collection, in others, it is instead too specific.

Table 2.4 shows the mapping from Levi's schema to mine, and below I expand on these concerns.

- **No corresponding class in Levi's classification**
  - Lexicalized NCs.

    My collection of NCs includes many non-compositional or lexicalized NCs, such as *vitamin k* and *e2 protein*. As lexicalized NCs do not represent regular grammatical process and must be learned on an item-by-item basis, Levi

does not include them in her analysis. Her study is restricted to endocentric complex nominals and does not consider, for example, metaphorical, lexicalized or idiomatic meanings.[8]

– Exclusion of "OF." Levi's system does not have an appropriate class for many NCs such as *tissue atropy* ("Change"), *artery diameter* ("Physical property"), *headache period, influenza season* ("Time of"), *asthma mortality, abortion rate* ("Measure of"), *cell apoptosis, infant death.* I argue that these NCs could be be described well by the (deleted) proposition "OF" that Levi does not include, as described in the previous section.

- **Multiple Levi classes for a single semantic relation: many-to-one mapping**

  For these cases, the main difference between my classification and Levi's is that my classification does not depend on whether or not nouns are derived from a verb by nominalization; only the semantic relationship between the nouns is taken into account.

  In my classification schema, the NCs *anti-migraine drugs, flu vaccine, asthma therapy, influenza vaccination, asthma treatment, tumor treatment* all fall under the "Purpose" semantic class. Under Levi's system, *anti-migraine drugs, flu vaccine, asthma therapy, influenza vaccination* would be "RDP, for" but *asthma treatment, tumor treatment* would be "NOM, Objective Act," since the noun *treatment* is derived by the verb *treat* by nominalization. Similarly, in

[8] Chapter 7 of Levi (1978) deals with exception classes, i.e., types of NCs that are exceptions to her theory and therefore not included, for example, CNs whose modifiers denote units of measure, and CNs whose adjectival modifiers must be derived from adverbs rather than nouns. Other examples whose meanings (and hence, derivations) her theory does not predict are: cat/morning/tennis person, volleyball/poetry freak, lunar/solar/fiscal/academic year, vodka/shopping binge, lunar/solar/fiscal academic year, dental/medical appointment, iron/bronze/industrial age, diamond/textile heir.
Her claim is that these NCs are formed not by the syntactic processes of predicate deletion or predicate nominalization but rather by a morphological process equivalent to suffixation.

my view all the following NCs contain an "Instrument" kind of relation: *injection method, laser surgeries, aciclovir therapy, vibration massage, lamivudine therapy, chloroquine treatment, ginseng treatment, laser treatment* but following Levi's schema they would be divided into two classes: "RDP, use" (*injection method, laser surgeries, aciclovir therapy, vibration massage, lamivudine therapy*) and "NOM, Subjective Act" (*chloroquine treatment, ginseng treatment, laser treatment*).

- **Single Levi's class corresponding to several semantic relations: one-to-many mapping**

  Most of Levi's classes correspond to several different categories in my classification schema. For example "Demographic attributes," "Located in," "Is part of" and "Time of" all correspond to "RDP, IN." This is due to the fact that the purposes of the two classification schemas are very different; Levi justifies this vagueness on three major grounds (as she writes on page 82) as follows:

  > *"First, the identification of the RDP set which they constitute allows us to make highly accurate predictions about which semantic structures can underlie CNs and which can not; second, the specification of general predicates serves to include a number of more specific relations that might otherwise have to be enumerated individually with a concomitant loss of generality; third, this analysis creates no more confusion or vagueness than is observable in periphrastic equivalents with overt lexicalizations of the RDPs (e.g., x has y, x is in y, x is for y), where some fundamental distinctions are made but where a great deal is typically left unspecified."*

  She lists some of the possible more specific relationships for IN: inhabits, grow-

in, according-to, during, found-in and occur-in. She claims that:

> *"The positing of six or more separate groups, however, would obscure the generalization that is captured in the present theory by positing a single predicate IN for both spatial and temporal, both concrete and abstract location. [...] we must recognize that (a) it cannot be accidental that all of these paraphrases must in any case include the locative preposition IN, and (b) the more specific verbs are wholly predictable on the basis of the semantic characteristics of the surface components of the CN. For example, almost any CN whose head noun denotes an animate creature and whose modifier is a place noun will be a suitable member for the INHABIT group. [...] Since, however, these differences not only are predictable on independent grounds but also seem to have no effect whatsoever on the formation of the corresponding CNs, our description need not include this of specificity."*

The important point is that the purpose of her work is the formation of CNs, while mine is their automatic classification. Under Levi's system, in order to predict the specific relationships on the basis of the semantics of the nouns, I would need to define the specific readings and add rules such as:

- IF the relationship is IN
- IF the first noun belongs to the class A and the second noun to the class B
- Then: IN means INHABITS

Rules such as these are brittle and difficult to construct and require the identification of the semantic classes A and B, as well as finding a way to assign

each noun to the corresponding semantic class. For my purposes, an approach in which NCs are directly mapped to more specific categories is preferable.

One might ask why I consider Levi's work if its goals are so different from mine. The reason is that I wanted to analyze an important linguistic theory on this subject, and, as already mentioned, most of the research in computational linguistics for the analysis of NCs cite this work. More importantly, some of them justify their classification schemas as based on Levi's (see, e.g., Vanderwende, 1994). However, based on the analysis above, I argue that despite this precedence, a different classification schema is needed for my task.

2.4.2 Beatrice Warren: Semantic Patterns of Noun-Noun Compounds

Warren's theory (Warren, 1978) is far less constraining than Levi's. Her purpose was to determine whether there exists a limited number of semantic relations between the constituents of NCs, and to determine the nature of these relations. Warren (1978) is primarily a report of the results of an empirical investigation; this work is a comprehensive study of 4557 manually extracted compound nouns from the Brown corpus.[9] Warren developed a taxonomy of implicit semantic relations, with four hierarchical levels of abstraction.

According to Warren (1978), there are six major types of semantic relations that can be expressed in compounds (in the following definitions A and B indicate, respectively, the first and the second noun):

- **Constitute**: A is something that wholly constitutes B, or vice versa: this class is divided into Source-Result, Result-Source and Copula classes. Some

[9] In particular, her collection comes from "The standard Corpus of Present-Day Edited American English" assembled at Brown University during 1963 and 1964.

Levi's classification	**My classification**
CAUSE1	Cause 1-2
CAUSE2	Cause 2-1
MAKE1	Produce
MAKE2	Physical make up
HAVE1	Person afflicted Location Material
HAVE2	Characteristic Physical property Defect
USE	Instrument 1-2
BE	Subtype Modal
IN	Demographic attributes Location Time of 1-2
FOR	Purpose Person/center who treats Instrument 2-1 Bind Activator 2-1
FROM	Material
ABOUT	Research on Attribute of clinical study Procedure Topic Standard Misuse
NOM: Subjective Act	Activity/physical process Change Instrument 1-2 Subject
NOM: Subjective Product	Ending/reduction Activator 1-2
NOM: Subjective Patient	-
NOM: Objective Act	Purpose Misuse Object
NOM: Objective Product	-
NOM: Objective Agent	Activator 2-1 Inhibitor

Table 2.4: For each of the Levi's classes, the semantic relations of Section 2.3 that correspond to them. Note that very different semantic relations fall under the same class under Levi'schema.

examples: *metal coupon, paste wax, student group.*

- **Possession** if A is something of which B is a part or a feature or vice versa; divided into Part-Whole, Whole-Part and Size-Whole classes. Examples: *board member, apple pie.*

- **Location**: A is the location or origin of B in time or space. Participant roles are Place-OBJ, Time-OBJ, Origin-OBJ. Examples: *coast road, Sunday school, seafood.*

- **Purpose**: A indicates the "purpose" of B. Participant roles: Goal-Instrumental. Examples: *pie tin, tablecloth.*

- **Activity-Actor**: A indicates the activity or interest with which B is habitually concerned (*cowboy, fire department*).

- **Resemblance**: A indicates something that B resembles. Participant roles: Comparant-Compared. Examples: *egghead, bullet head.*

The semantic relations can often be paraphrased (although paraphrasing does not work with idiomatization.) In Table 2.5 are shown the major classes and subclasses, the paraphrases, and some examples of Warren's classification of noun compounds.

These classes are further subdivided into a hierarchical organization. I do not describe the classes in detail (there is a whole chapter for each class in Warren, 1978) but in Figures 2.1, 2.2, 2.3 and 2.4 one can see Warren's semantic relations further subdivided into several levels of sub-classes.

2.4.2.1 Is Warren's classification system appropriate?

There is a fairly good correspondence between my relations and Warren's. For example, my "Cause (2-1)" for *flu virus* corresponds to her "Causer-Result" of *hay fever*,

MAJOR CLASS	SUB CLASSES	PARAP.	EXAMPLES
Contitute	Source-Result	OF	*metal sheet*
	Result-Source	IN	*sheet metal*
	Copula	-	*girl friend*
Possession	Part-Whole	OF	*eggshell*
	Whole-Part	WITH	*armchair*
Location	Place-OBJ	IN, AT, ON	*coast road*
	Time-OBJ	IN, AT, ON	*Sunday school*
	Origin-OBJ	FROM	*seafood*
Purpose	Goal-Instrumental	FOR	*pie tin*
Activity-Actor	Activity-Obj	-	*cowboy*
Resemblance	Comparant-Compared	LIKE	*cherry bomb*

Table 2.5: Warren semantic relations for NCs

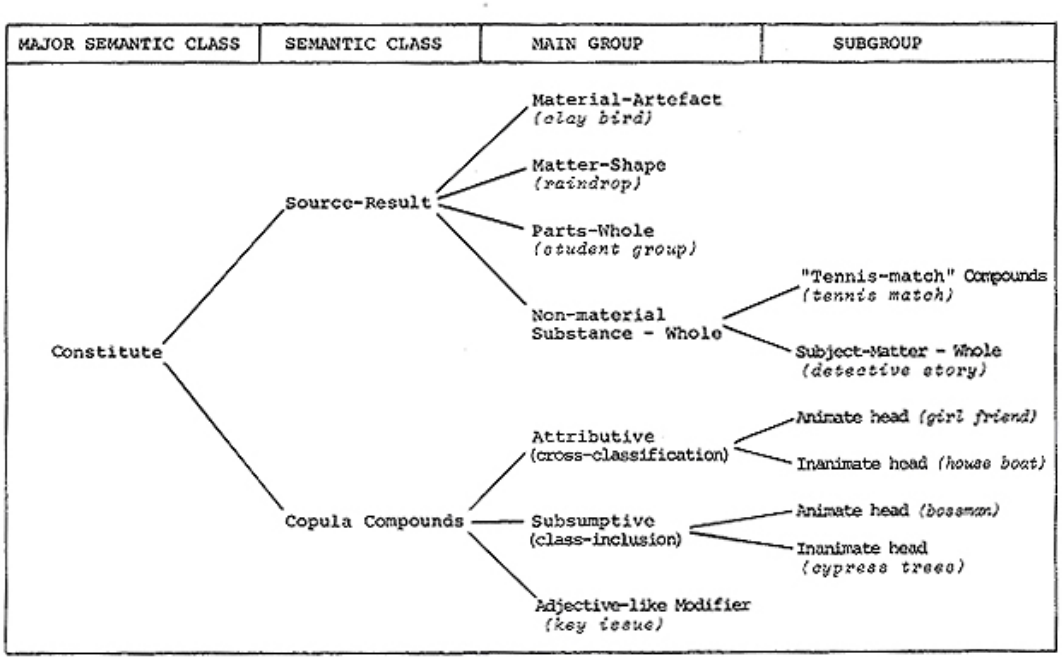


Figure 2.1: Warren's Constitute semantic relation (Warren, 1978).

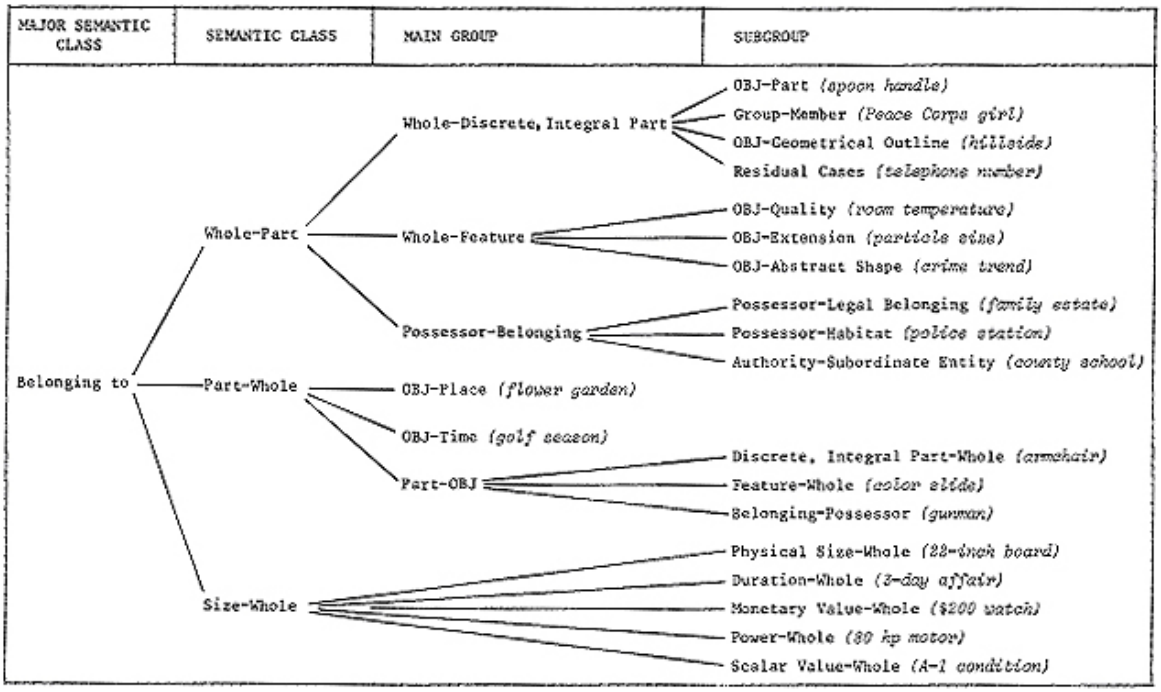


Figure 2.2: Warren's Possession semantic relation (Warren, 1978).

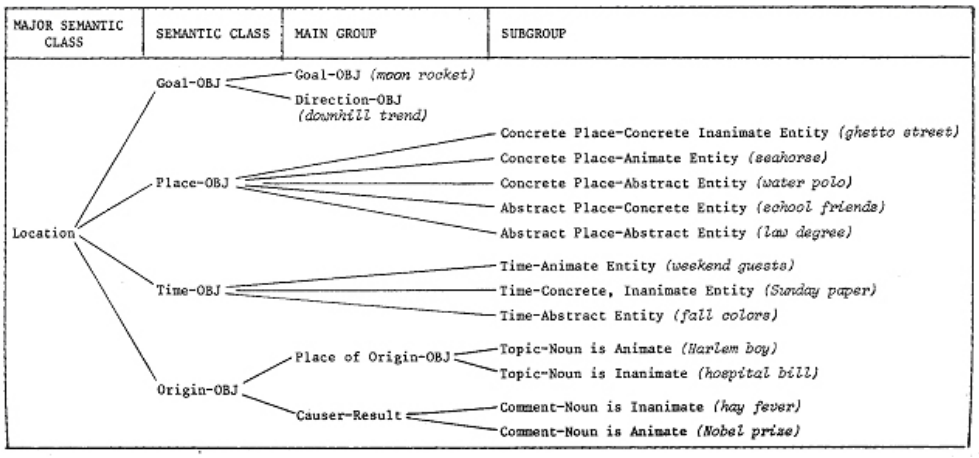


Figure 2.3: Warren's Location semantic relation (Warren, 1978).

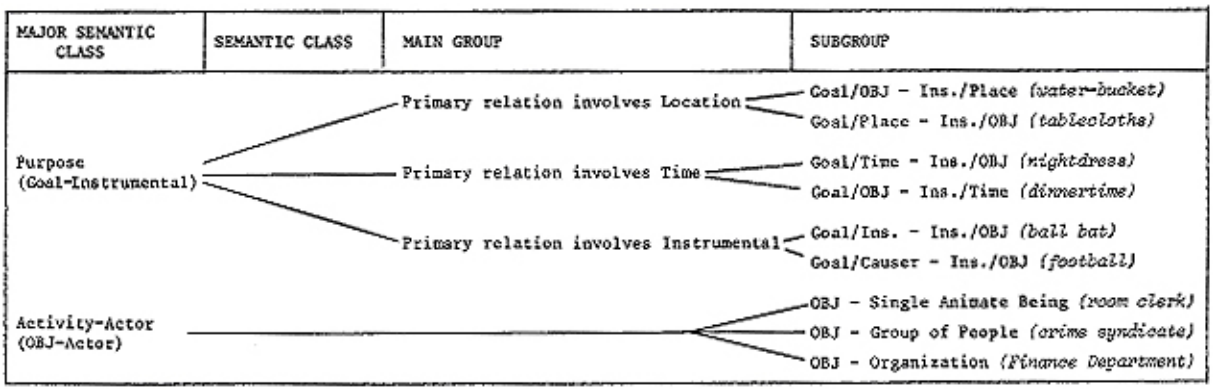


Figure 2.4: Warren's Purpose and Activity-actor semantic relations (Warren, 1978).

my "Topic" (*health education*) to her "Subject-Matter-Whole" (*detective story*) and my "Person Afflicted" (*migraine patient*) can be thought as Warren's "Belonging-Possessor" of *gunman*.

Warren's investigation is empirical, and it may well be that some relations that I found in my collection never occurred in hers. For example, I was not able to find an appropriate relation in Warren's schema for *heroin use, internet use, drug utilization* that I classified as "Instrument 1." Perhaps the closest class is "Material-artifact" of *clay bird, brass wire* except that *use, utilization* are not "artifacts." Also, I was not able to classify under Warren's schema NCs for "Misuse" (*drug abuse, acetaminophen overdose*), "Binds" (*receptor ligand*), "Activator 2-1" (*headache trigger*), "Inhibitor" (*receptor blockers*), but again these NCs are very domain specific.

More importantly, many of the NCs for which I could not find a corresponding relation in Warren's schema are nominalizations (NCs in which one noun is deverbal) that Warren does not include; some examples are: *tissue reinforcement, tumor development, ventricle enlargement* ("Change"), *migraine relief, headache decrease* ("Ending/reduction"), *kidney transplant, liver transplantation* ("Object").

Warren's motivation for excluding nominalizations is that a main goal of her work

is to determine which semantic relations between two nouns can be left unexpressed, or which verbs connecting the two nouns may be discarded. In a compound containing a deverbal noun there is no verb deleted and the type of semantic relation that exists between the constituents is explicit, while this relation is implicit in non-verbal compounds. An important consequence of this distinction is that nominalizations do NOT appear to be restricted as to the type of relations that may exists between the constituents. Indeed, there is no need for such a restriction since the type of relation between the nouns is explicitly indicated by the deverbal noun. In contrast, there are a limited number of relations that are left unexpressed for non-verbal NCs.

I argue that Warren's motivation for excluding nominalizations is not appropriate for the problem tackled here – the automatic classification of NCs into semantic classes. In theory, we could first divide NCs into verbal and non-verbal (with a binary classification system perhaps) and then analyze the two classes separately. This would make the system more complicated, but it would be useful if the number of possible types of relations in verbal NCs is indeed unconstrained and explicit and if we knew how to infer the relation given the verb (which is another problem in it's own right).[10] Moreover, we would still want to infer the same relation for *headache transformation* (verbal) and *tissue atrophy* (non-verbal), for example. Warren acknowledges this fact and notes that her distinction would bring her to include *sugar bowl* while excluding *sugar container* as a verbal combination in spite of the fact that the two NCs express the same semantic relationship.

Warren's classification is very detailed and her definition of subclasses is well-aligned with my tasks. However, the organization of her classes sometimes does not reflect the similarities of the underlying relations.

As an example, *3-day affair* and *75-minute concert* are "Duration-whole" in the

[10]Lapata (2000) proposes algorithms for the classification of nominalizations according to whether the modifier is the subject or the object of the underlying verb expressed by the head noun.

"Possession" class because the comment indicates the duration of the topic, while *morning concert* and *Sunday paper* are "Time-Object" in "Location" because comment indicates when the topic takes place; these NCs end up in two entirely different semantic classes, obscuring therefore the similarity of the semantic relations. Similarly, "Belonging-Possessor" (*gunman*) and "Possessor-Belonging" (*family estate*) are opposite classes within the "Possession" major class (see Figure 2.2, in both cases there is a "own" relationship, but in *gunman* the topic owns the comment while in *family estate* it's the comment that owns the topic). "Belonging-Possessor" is sub-class of "Part-Whole" while "Possessor-Belonging" is sub-class of "Whole-Part." While this is logical, it results in two classes that are natural opposites of each other being "located" in two positions in the hierarchy that are far apart. Just by looking at the hierarchy it is not immediately evident that these two classes are so closely related.

If we do not use the hierarchy but instead use only the classes in their own this concern can be ignored. However, if we do want to take advantage of the hierarchical structure, the fact that similar concepts like *3-day affair* are a "Possession" while *weekend affair* is a "Location" could be problematic.

Warren differentiates some classes on the basis of the semantics of the constituents, so that, for example, the "Time" relationship is divided up into "Time-Animate Entity" of *weekend guests* and "Time-Inanimate Entity" of *Sunday paper*. By contrast, my classification is based on the kind of relationships that hold between the constituent nouns rather than on the semantics of the head nouns.

To summarize this section, the problem of determining what the appropriate semantic relations are is a complex one. A detailed review of the linguistic literature did not suggest a set of relations appropriate for my collection of NCs. The empirical analysis I performed on my collection has a similar goal to that of Warren (1978) and the semantic patterns I identified do correspond, to some extent, to those of Warren

(1978). The different characteristics of the two collections, however, require different relations and it may be the case that this is true in general, that is, new sets of semantic relations may be needed for each domain. Moreover, the intended use of these semantic relations may also affect their choice.

2.5 Related work

Several approaches have been proposed for empirical noun compound interpretation. Lauer and Dras (1994) point out that there are three components to the problem: identification of the compound from within the text, syntactic analysis of the compound (left versus right association), and the interpretation of the underlying semantics. Several researchers have tackled the syntactic analysis (Lauer, 1995a; Pustejovsky et al., 1993; Liberman and Church, 1992), usually using a variation of the idea of finding the subconstituents elsewhere in the corpus and using those to predict how the larger compounds are structured.

2.5.1 Noun Compound Relation Assignment

Most related work on the interpretation of the semantics relies on hand-written rules of one kind or another. Finin (1980) examines the problem of noun compound interpretation in detail, and constructs a complex set of rules. Vanderwende (1994) uses a sophisticated system to extract semantic information automatically from an online dictionary, and then manipulates a set of hand-written rules with hand-assigned weights to create an interpretation. Rindflesch et al. (2000b) use hand-coded rule based systems to extract the factual assertions from biomedical text.

Lapata (2000) classifies nominalizations according to whether the modifier is the subject or the object of the underlying verb expressed by the head noun (see Section

2.4.1 for a discussion about nominalizations.) She reports an accuracy of 80% for the easier problem of binary classification.

Barker and Szpakowicz (1998) describe noun compounds as triplets of information: the first constituent, the second constituent, and a marker that can indicate a number of syntactic clues. Relations are initially assigned by hand, and then new ones are classified based on their similarity to previously classified NCs. However, similarity at the lexical level means only that the same word occurs; no generalization over lexical items is made. The algorithm is assessed in terms of how much it speeds up the hand-labeling of relations. Barrett et al. (2001) have a somewhat similar approach, using WordNet and creating heuristics about how to classify a new NC given its similarity to one that has already been seen.

2.5.2 Using Lexical Hierarchies

Many approaches attempt to automatically assign semantic roles (such as case roles) by computing semantic similarity measures across a large lexical hierarchy; primarily using WordNet (Fellbaum, 1998). Budanitsky and Hirst (2001) provide a comparative analysis of such algorithms.

However, it is uncommon to simply use the hierarchy directly for generalization purposes. Many researchers have noted that WordNet's words are classified into senses that are too fine-grained for standard NLP tasks. For example, Buitelaar (1997) notes that the noun *book* is assigned to seven different senses, including *fact* and *section, subdivision*. Thus most users of WordNet must contend with the sense disambiguation issue in order to use the lexicon.

There have been several efforts to incorporate lexical hierarchies for the problem of prepositional phrase (PP) attachment. The current standard formulation is: given a verb followed by a noun and a prepositional phrase, represented by the tuple

$v, n1, p, n2$, determine which of v or $n1$ the PP consisting of p and $n2$ attaches to, or is most closely associated with. As an example, consider the following minimal pair:

(i) *eat spaghetti with a fork*

(ii) *eat spaghetti with sauce*

In the PP attachment problem, one has to determine which is a more likely association: fork and eat, or fork and spaghetti.

Because the data is sparse, empirical methods that train on word occurrences alone have been supplanted by algorithms that generalize one or both of the nouns according to class-membership measures (Resnik, 1993; Resnik and Hearst, 1993; Brill and Resnik, 1994; Li and Abe, 1998), but the statistics are computed for the particular preposition, verb and noun. Resnik (1993, 1995) uses classes in Wordnet and a measure of conceptual association to generalize over the nouns. Brill and Resnik (1994) use Brill's transformation-based algorithm along with simple counts within a lexical hierarchy in order to generalize over individual words.

It is not clear how to use the results of such analysis after they are found; the semantics of the relationship between the terms must still be determined. In my framework we would cast this problem as finding the relationship $R(p, n2)$ that best characterizes the preposition and the NP that follows it, and then seeing if the categorization algorithm determines if there exists any relationship $R'(n1, R(p, n2))$ or $R'(v, R(p, n2))$.

One difficulty with the standard PP attachment problem formulation is that fork/spaghetti and sauce/eat are both related to each other, but they are related to each other in two different ways. Instead, I ask the question: what is the relationship between fork and spaghetti and between sauce and spaghetti (contrasting the noun pairs, as opposed to the verb-noun pairs).

The most closely related use of a lexical hierarchy that I know of is that of Li and Abe (1998), which uses an information-theoretic measure to make a cut through the top levels of the noun portion of WordNet. This is then used to determine acceptable classes for verb argument structure, as well as for the prepositional phrase attachment problem.

My approach (described in Section 2.6) differs from these in that I use machine learning techniques to determine which level of the lexical hierarchy is appropriate for generalizing across nouns.

2.6 Classifying the semantic relations

In the previous sections I introduced my collection of NCs, the semantic relations that I identified, and some of the linguistic theories for this domain. In this section, I present my work on the automatic classification of the NCs of my collection using a neural net approach (Rosario and Hearst, 2001). In the following section I will describe an alternative method called "descent of hierarchy" (Rosario et al., 2002) that does not make use of machine learning.

I have found that I can use a standard machine learning classification technique to classify relationships between two-word noun compounds with a surprising degree of accuracy. A one-out-of-eighteen classification using a neural net achieves accuracies as high as 62%; this result can be compared with the baseline accuracies of 5% of chance and 30% obtained with logistic regression. By taking advantage of lexical ontologies, I achieve strong results on noun compounds for which neither word is present in the training set. Thus, I think this approach is promising for a variety of semantic labeling tasks.

Section 2.6.1 describes the ontologies used; in Section 2.6.2 I describe the method for automatically assigning semantic relations to noun compounds, and report the

results of experiments using this method; finally Section 2.6.3 discusses this work.

2.6.1 Lexical Resources

The Unified Medical Language System (UMLS) is a biomedical lexical resource produced and maintained by the National Library of Medicine (Humphreys et al., 1998). I use the MetaThesaurus component to map lexical items into unique concept IDs (CUIs).[11] The UMLS also has a mapping from these CUIs into the MeSH lexical hierarchy; I mapped the CUIs into MeSH terms.

MeSH (Medical Subject Headings)[12] is the National Library of Medicine's controlled vocabulary thesaurus; it consists of set of main terms (as well as additional modifiers) arranged in a hierarchical structure. There are 15 main sub-hierarchies (trees) in MeSH, each corresponding to a major branch of medical terminology. For example, tree A corresponds to Anatomy, tree B to Organisms, tree C to Diseases and so on. Every branch has several sub-branches; Anatomy, for example, consists of Body Regions (A01), Musculoskeletal System (A02), Digestive System (A03) etc. I refer to these as "Model 1" or "level 1" categories.

These nodes have children, for example, Abdomen (A01.047) and Back (A01.176) are level 2 children of Body Regions. The longer the ID of the MeSH term, the longer the path from the root and the more precise the description. For example migraine is C10.228.140.546.800.525, that is, C (a disease), C10 (Nervous System Diseases), C10.228 (Central Nervous System Diseases) and so on. There are over 35,000 unique IDs in MeSH 2001. Many words are assigned more than one MeSH ID and so occur in more than one location within the hierarchy; thus the structure of MeSH can be interpreted as a network.

[11] In some cases a word maps to more than one CUI; for the work reported here I arbitrarily chose the first mapping in all cases.

[12] http://www.nlm.nih.gov/mesh/meshhome.html. The work reported in this paper uses MeSH 2001.

I use the MeSH hierarchy for generalization across classes of nouns; I use it instead of the other resources in the UMLS primarily because of MeSH's hierarchical structure. For these experiments, we considered only those noun compounds for which both nouns can be mapped into MeSH terms, resulting in a total of 2245 NCs.

2.6.2 Method and Results

Because I have defined noun compound relation determination as a classification problem, I can make use of standard classification algorithms; in particular, I used neural networks.

I ran experiments creating models that used different levels of the MeSH hierarchy. For example, for the NC *flu vaccination*, *flu* maps to the MeSH term D4.808.54.79.429.154.349 and *vaccination* to G3.770.670.310.890. *Flu vaccination* for Model 4 would be represented by a vector consisting of the concatenation of the two descriptors showing only the first four levels: D4.808.54.79 G3.770.670.310 (see Table 2.6). When a word maps to a general MeSH term (like treatment, Y11) zeros are appended to the end of the descriptor to stand in place of the missing values (so, for example, treatment in Model 3 is Y 11 0, and in Model 4 is Y 11 0 0, etc.).

The numbers in the MeSH descriptors are categorical values; I represented them with indicator variables. That is, for each variable I calculated the number of possible categories c and then represented an observation of the variable as a sequence of c binary variables in which one binary variable was one and the remaining $c-1$ binary variables were zero.

I also used a representation in which the words themselves were used as categorical input variables (I call this representation "lexical"). For this collection of NCs there were 1184 unique nouns and therefore the feature vector for each noun had 1184 components. In Table 2.7 I report the length of the feature vectors for one noun for

	flu vaccination
Model 2	D 4 G 3
Model 3	D 4 808 G 3 770
Model 4	D 4 808 54 G 3 770
Model 5	D 4 808 54 79 G 3 770 670
Model 6	D 4 808 54 79 429 G 3 770 670 310

Table 2.6: Different lengths of the MeSH descriptors for the different models

Model	Feature Vector
2	42
3	315
4	687
5	950
6	1111
Lexical	1184

Table 2.7: Length of the feature vectors for different models.

each model. The entire NC was described by concatenating the feature vectors for the two nouns in sequence.

The NCs represented in this fashion were used as input to a neural network. I used a feed-forward network trained with conjugate gradient descent. The network had one hidden layer, in which a hyperbolic tangent function was used, and an output layer representing the 18 relations (in bold in Table 2.1). A logistic sigmoid function was used in the output layer to map the outputs into the interval $(0, 1)$.

The number of units of the output layer was the number of relations (18) and therefore fixed. The network was trained for several choices of numbers of hidden units; I chose the best-performing networks based on training set error for each of the models. I subsequently tested these networks on held-out testing data.

I compared the results with a baseline in which logistic regression was used on the lexical features. Given the indicator variable representation of these features, this

logistic regression essentially forms a table of log-odds for each lexical item. I also compared to a method in which the lexical indicator variables were used as input to a neural network. This approach is of interest to see to what extent, if any, the MeSH-based features affect performance. Note also that this lexical neural-network approach is feasible in this setting because the number of unique words is limited (1184) – such an approach would not scale to larger problems.

Multi-class classification is a difficult problem (Vapnik, 1998). In this problem, a baseline in which the algorithm guesses yields about 5% accuracy. In Table 2.8 and in Figure 2.5 I report the results from these experiments. These results show that my method is a significant improvement over the tabular logistic-regression-based approach, which yields an accuracy of only 31 percent. Additionally, despite the significant reduction in raw information content as compared to the lexical representation, the MeSH-based neural network performs as well as the lexical-based neural network. (And I again stress that the lexical-based neural network is not a viable option for larger domains.) The neural network using only lexical features yields 62% accuracy on average across all 18 relations. A neural net trained on Model 6 using the MeSH terms to represent the nouns yields an accuracy of 61% on average across all 18 relations. Note that reasonable performance is also obtained for Model 2, which is a much more general representation. Table 2.8 shows that both methods achieve up to 78% accuracy at including the correct relation among the top three hypothesized.

Figure 2.6 shows the results for each relation. MeSH-based generalization does better on some relations (for example 14 and 15) and Lexical on others (7, 22). It turns out that the test set for relationship 7 ("Produces on a genetic level") is dominated by NCs containing the words *alleles* and *mrna* and that *all* the NCs in the training set containing these words are assigned relation label 7. A similar situation is seen for relation 22, "Time(2-1)." In the test set examples the second noun is either *recurrence*, *season* or *time*. In the training set, these nouns appear *only* in NCs that

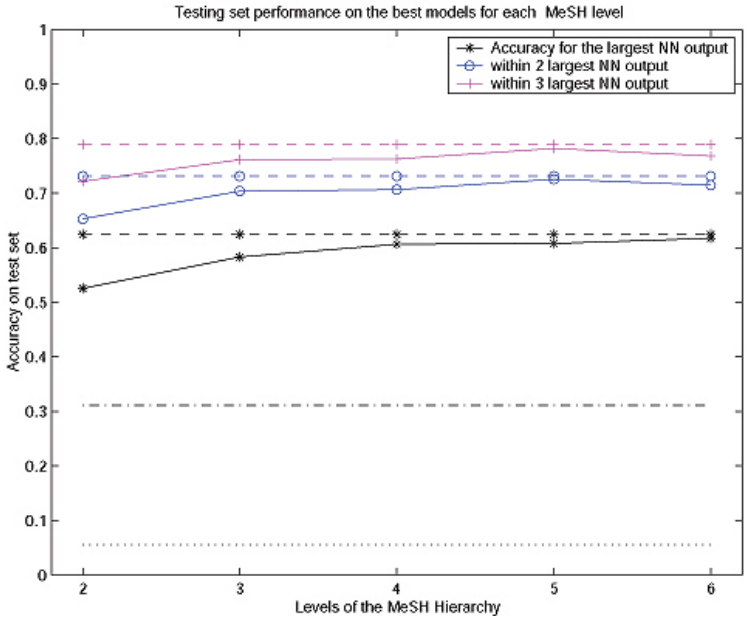


Figure 2.5: Accuracies on the test sets for all the models. The dotted line at the bottom is the accuracy of guessing (the inverse of the number of classes). The dash-dot line above this is the accuracy of logistic regression on the lexical data. The solid line with asterisks is the accuracy of my representation, when only the maximum output value from the network is considered. The solid line with circles if the accuracy of getting the right answer within the two largest output values from the neural network and the last solid line with the plus signs is the accuracy of getting the right answer within the first three outputs from the network. The three flat dashed lines are the corresponding performances of the neural network on lexical inputs.

Model	Acc1	Acc2	Acc3
Lexical: Log Reg	0.31	0.58	0.62
Lexical: NN	0.62	0.73	0.78
2	0.52	0.65	0.72
3	0.58	0.70	0.76
4	0.60	0.70	0.76
5	0.60	0.72	0.78
6	0.61	0.71	0.76

Table 2.8: Test accuracy for each model, where the model number corresponds to the level of the MeSH hierarchy used for classification. Lexical NN is Neural Network on Lexical and Lexical: Log Reg is Logistic Regression on NN. Acc1 refers to how often the correct relation is the top-scoring relation, Acc2 refers to how often the correct relation is one of the top two according to the neural net, and so on. Uniform guessing would yield a result of 0.077.

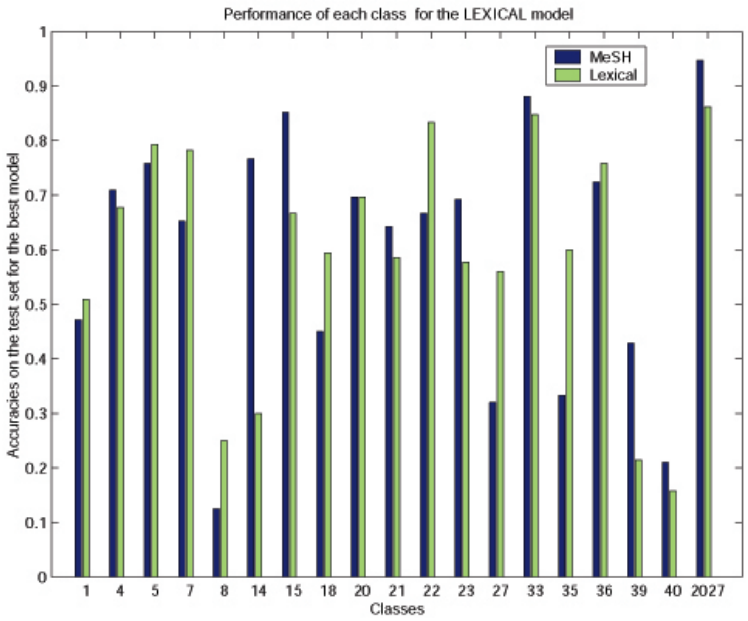


Figure 2.6: Accuracies for each class. The numbers at the bottom refer to the class numbers in Table 2.1. Note the very high accuracy for the "mixed" relationship 20-27 (last bar on the right).

have been labeled as belonging to relation 22.

On the other hand, looking at relations 14 and 15, there is a wider range of words, and in some cases the words in the test set are not present in the training set. In relationship 14 ("Purpose"), for example, *vaccine* appears 6 times in the test set (e.g., *varicella vaccine*). In the training set, NCs with *vaccine* in it have also been classified as "Instrument" (*antigen vaccine*, *polysaccharide vaccine*), as "Object" (*vaccine development*), as "Subtype of" (*opv vaccine*) and as "Wrong" (*vaccines using*). Other words in the test set for 14 are *varicella* which is present in the training set only in *varicella serology* labeled as "Attribute of clinical study," *drainage* which is in the training set only as "Location" (*gallbladder drainage* and *tract drainage*) and "Activity" (*bile drainage*). Other test set words such as *immunisation* and *carcinogen* do not appear in the training set at all.

In other words, it seems that the MeSH-based categorization does better when generalization is required. Additionally, this data set is "dense" in the sense that very few testing words are not present in the training data. The results reported in Table 2.8 and in Figure 2.5 were obtained splitting the data into 50% training and 50% testing for each relation with a total of 855 training points and 805 test points. Of these, only 75 examples in the testing set consisted of NCs in which both words were not present in the training set.

This is of course an unrealistic situation and so I tested the robustness of the method in a more realistic setting. I was also interested in seeing the relative importance of the first versus the second noun. Therefore, I split the data into 5% training (73 data points) and 95% testing (1587 data points) and partitioned the testing set into 4 subsets as follows (the numbers in parentheses are the number of points for each case):

- Case 1: NCs for which the first noun was not present in the training set (424)

Model	All test	Case 1	Case 2	Case 3	Case 4
Lexical: NN	0.23	0.54	0.14	0.33	0.08
2	0.44	0.62	0.25	0.53	0.38
3	0.41	0.62	0.18	0.47	0.35
4	0.42	0.58	0.26	0.39	0.38
5	0.46	0.64	0.28	0.54	0.40
6	0.44	0.64	0.25	0.50	0.39

Table 2.9: Test accuracy for the four sub-partitions of the test set.

- Case 2: NCs for which the second noun was not present in the training set (252)
- Case 3: NCs for which both nouns were present in the training set (101)
- Case 4: NCs for which both nouns were not present in the training set (810).

Table 2.9 and Figures 2.7 and 2.8 present the accuracies for these test set partitions. Figure 2.7 shows that the MeSH-based models are more robust than the lexical when the number of unseen words is high and when the size of training set is (very) small. In this more realistic situation, the MeSH models are able to generalize over previously unseen words. For unseen words, lexical reduces to guessing.[13]

Figure 2.8 shows the accuracy for the MeSH based-model for the four cases of Table 2.9. It is interesting to note that the accuracy for Case 1 (first noun not present in the training set) is much higher than the accuracy for Case 2 (second noun not present in the training set). This seems to indicate that the second noun is more important for the classification that the first one.

2.6.3 Conclusions about the Neural Net Approach

I have presented a simple approach to corpus-based assignment of semantic relations for noun compounds. The main idea is to define a set of relations that can hold be-

[13]Note that for unseen words, the baseline lexical-based logistic regression approach, which essentially builds a tabular representation of the log-odds for each class, also reduces to guessing.

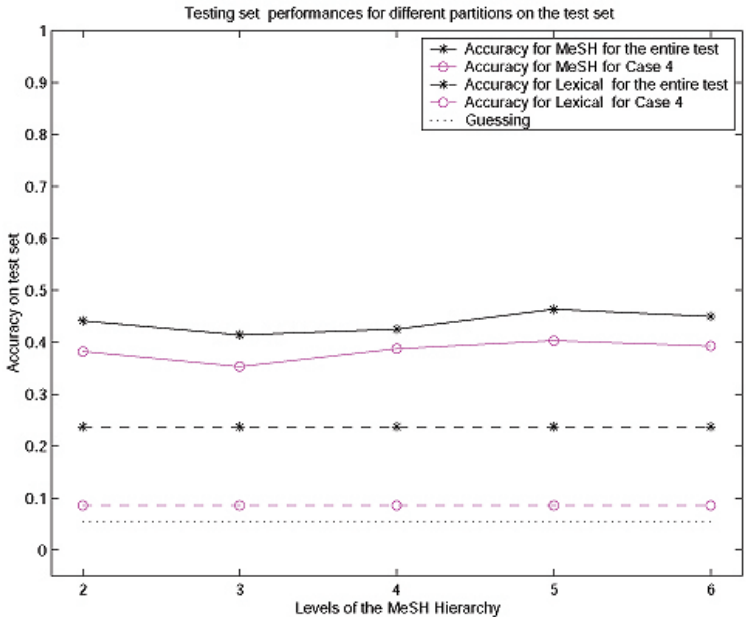


Figure 2.7: Comparing original test set with Case 4 in which none of the nouns in the test set were present in the training set. The unbroken lines represent the MeSH models accuracies (for the entire test set and for case 4) and the dashed lines represent the corresponding lexical accuracies. The accuracies are smaller than the previous case of Table 2.8 because the training set is much smaller, but the point of interest is the difference in the performance of MeSH vs. lexical in this more difficult setting. Note that lexical for case 4 reduces to random guessing.

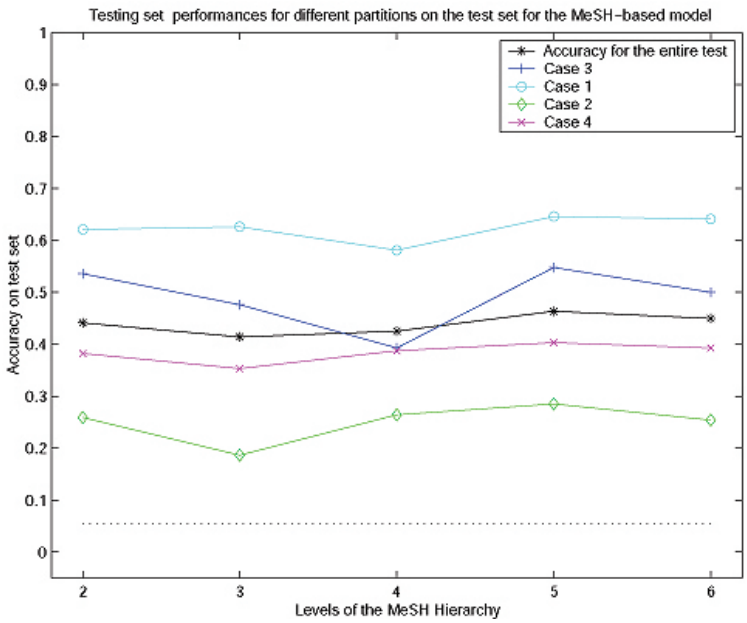


Figure 2.8: Accuracy for the MeSH based-model for the four cases. All these curves refer to the case of getting exactly the right answer. Note the difference in performance between case 1 (first noun not present in the training set) and case 2 (second noun not present in training set).

tween the terms and use standard machine learning techniques and a lexical hierarchy to generalize from training instances to new examples.

In this task of multi-class classification (with 18 classes) I achieved an accuracy of about 60%. These results can be compared with Vanderwende (1994) who reports an accuracy of 52% with 13 classes and Lapata (2000) whose algorithm achieves about 80% accuracy for a much simpler binary classification.

I have shown that a class-based representation performes as well as a lexical-based model despite the reduction of raw information content and despite a somewhat errorful mapping from terms to concepts. I have also shown that representing the nouns of the compound by a very general representation (Model 2) achieves a reasonable performance of about 52% accuracy on average. This is particularly important in the case of larger collections with a much bigger number of unique words for which the lexical-based model is not a viable option. Our results seem to indicate that I do not lose much in terms of accuracy using the more compact MeSH representation.

I have also shown how MeSH-based models outperform a lexical-based approach when the number of training points is small and when the test set consists of words unseen in the training data. This indicates that the MeSH models can generalize successfully over unseen words. My approach handles "mixed-class" relations naturally. For the mixed class Defect in Location, the algorithm achieved an accuracy around 95% for both "Defect" and "Location" simultaneously. My results also indicate that the second noun (the head) is more important in determining the relationships than the first one.

Future work could include training the algorithm to allow different levels for each noun in the compound, comparing the results to the tree cut algorithm reported in Li and Abe (1998), which allows different levels to be identified for different subtrees, and tackling the problem of noun compounds containing more than two terms.

2.7 The descent of hierarchy

2.7.1 Introduction

This section describes a second approach for the semantic analysis of NCs (Rosario et al., 2002).

As seen in the previous sections, interpretation of noun compounds is highly dependent on lexical information. In this section, I explore the use of a large corpus (MEDLINE) and a large lexical hierarchy (MeSH) for the purpose of placing words from a noun compound into categories, and then using this category membership to determine the relation that holds between the nouns. Surprisingly, I find that I can simply use the juxtaposition of category membership within the lexical hierarchy to determine the relation that holds between pairs of nouns. For example, for the NCs *leg paresis, skin numbness,* and *hip pain*, the first word of the NC falls into the MeSH A01 (Body Regions) category, and the second word falls into the C10 (Nervous System Diseases) category. From these I can declare that the relation that holds between the words is "located in." Similarly, for *influenza patients* and *aids survivors*, the first word falls under C02 (Virus Diseases) and the second is found in M01.643 (Patients), yielding the "afflicted by" relation. Using this technique on a subpart of the category space, I obtain 90% overall accuracy.

In some sense, this is a very old idea, dating back to the early days of semantic nets and semantic grammars. The critical difference now is that large lexical resources and corpora have become available, thus allowing some of those old techniques to become feasible in terms of coverage. However, the success of such an approach depends on the structure and coverage of the underlying lexical ontology. Since lexical hierarchies are not necessarily ideally suited for this task, I also pose the question: how far down the hierarchy must the algorithm descend before all the terms within the subhierarchy

behave uniformly with respect to the semantic relation in question? I find that the topmost levels of the hierarchy yield an accurate classification, thus providing an economic way of assigning relations to noun compounds.

In the following sections, I discuss the linguistic motivations behind this approach, the characteristics of the lexical ontology MeSH, the use of a corpus to examine the problem space, the method of determining the relations, the accuracy of the results, and the problem of ambiguity.

2.7.2 Linguistic Motivation

One way to understand the relations between the words in a two-word noun compound is to cast the words into a head-modifier relationship, and assume that the head noun has an argument structure, much the way verbs do, as well as a qualia structure in the sense of Pustejovsky (Pustejovsky, 1995). Then the meaning of the head noun determines what kinds of things can be done to it, what it is made of, what it is a part of, and so on.

For example, consider the noun *knife*. Knives are created for particular activities or settings, can be made of various materials, and can be used for cutting or manipulating various kinds of things. A set of relations for knives, and example NCs exhibiting these relations is shown below:

kitchen knife, hunting knife: "Used-in"

steel knife, plastic knife: "Made-of"

carving knife: "Instrument-for"

meat knife, putty knife: "Used-on"

chef's knife, butcher's knife: "Used-by"

Some relationships apply to only certain classes of nouns; the semantic structure of the head noun determines the range of possibilities. Thus if we can capture regularities about the behaviors of the constituent nouns, we should also be able to predict which

relations will hold between them.

I propose using the categorization provided by a lexical hierarchy for this purpose. Using a large collection of noun compounds, I assign semantic descriptors from the lexical hierarchy to the constituent nouns and determine the relations between them. This approach avoids the need to enumerate in advance all of the relations that may hold. Rather, the corpus determines which relations occur.

2.7.3 The Lexical Hierarchy: MeSH

As mentioned in Section 2.6.1, MeSH (Medical Subject Headings) is the National Library of Medicine's controlled vocabulary thesaurus; it consists of terms arranged in a hierarchical structure. There are 15 main sub-hierarchies in MeSH, for example, tree A corresponds to Anatomy, tree B to Organisms, tree C to Diseases and so on. (See Section 2.6.1 for a more detailed description of MeSH.)

Some of the MeSH sub-hierarchies are more homogeneous than others. The tree A (Anatomy) for example, seems to be quite homogeneous; at level 1, the nodes are all *part of* (meronymic to) Anatomy: the Digestive (A03), Respiratory (A04) and the Urogenital (A05) Systems are all part of anatomy; at level 2, the Biliary Tract (A03.159) and the Esophagus (A03.365) are part of the Digestive System (level 1) and so on. Thus I assume that every node is a (body) part of the parent node (and all the nodes above it). Tree C for Diseases is also homogeneous; the child nodes are a *kind of* (hyponym of) the disease at the parent node: Neoplasms (C04) is a *kind of* Disease C and Hamartoma (C04.445) is a *kind of* Neoplasms.

Other trees are more heterogeneous, in the sense that the meanings among the nodes are more diverse. Information Science (L01), for example, contains, among others, Communications Media (L01.178), Computer Security (L01.209) and Pattern Recognition (L01.725). Another heterogeneous sub-hierarchy is Natural Science

(H01). Among the children of H01 I find Chemistry (parent of Biochemistry), Electronics (parent of Amplifiers and Robotics), Mathematics (Fractals, Game Theory and Fourier Analysis). In other words, I find a wide range of concepts that are not described by a simple relationship.

These observations suggest that once an algorithm descends to a homogeneous level, words falling into the subhierarchy at that level (and below it) may behave similarly with respect to relation assignment.

2.7.4 Counting Noun Compounds

In this and the next section, I describe how I investigated the hypothesis:

> For all two-word noun compounds (NCs) that can be characterized by a category pair (CP), a particular semantic relationship holds between the nouns comprising those NCs.

The kinds of relations I found are similar to those described in Section 2.7.2. Note that, in this analysis I focused on determining which sets of NCs fall into the same relation, without explicitly assigning names to the relations themselves. Furthermore, the same relation may be described by many different category pairs (see Section 2.7.5.5).

First, I extracted two-word noun compounds from approximately 1M titles and abstracts from the Medline collection of biomedical journal articles, resulting in about 1M NCs. The NCs were extracted by finding adjacent word pairs in which both words are tagged as nouns by a part-of-speech tagger (Brill, 1995) and appear in the MeSH hierarchy, and the words preceding and following the pair do not appear in MeSH.[14] Of these two-word noun compounds, 79,677 were unique.

[14]Clearly, this simple approach results in some erroneous extractions.

Next I used MeSH to characterize the NCs according to semantic category(ies). For example, the NC *fibroblast growth* was categorized into A11.329.228 (Fibroblasts) and G07.553.481 (Growth).

Note that the same words can be represented at different levels of description. For example, *fibroblast growth* can be described by the MeSH descriptors A11.329.228 G07.553.481 (original level), but also by A11 G07 (Cell and Physiological Processes, level 1) or A11.329 G07.553 (Connective Tissue Cells and Growth and Embryonic Development, level 2). If a noun fell under more than one MeSH ID, I made multiple versions of this categorization. I refer to the result of this renaming as a category pair (CP).

I placed these CPs into a two-dimensional table, with the MeSH category for the first noun on the X axis, and the MeSH category for the second noun on the Y axis. Each intersection indicates the number of NCs that are classified under the corresponding two MeSH categories.

A visualization tool (Ahlberg and Shneiderman, 1994) allowed me to explore the dataset to see which areas of the category space are most heavily populated, and to get a feeling for whether the distribution is uniform or not (see Figure 2.9). If my hypothesis holds (that NCs that fall within the same category pairs are assigned the same relation), then if most of the NCs fall within only a few category pairs then we only need to determine which relations hold between a subset of the possible pairs. Thus, the more clumped the distribution, the easier (potentially) my task is. Figure 2.9 shows that some areas in the CP space have a higher concentration of unique NCs (the Anatomy, and the E through N sub-hierarchies, for example), especially when I focus on those for which at least 50 unique NCs are found.

Figure 2.10 focuses on the distribution of NCs for which the first noun can be classified under the Anatomy category. Note that many of the possible second noun categories are sparsely populated, again potentially reducing the space of the problem.

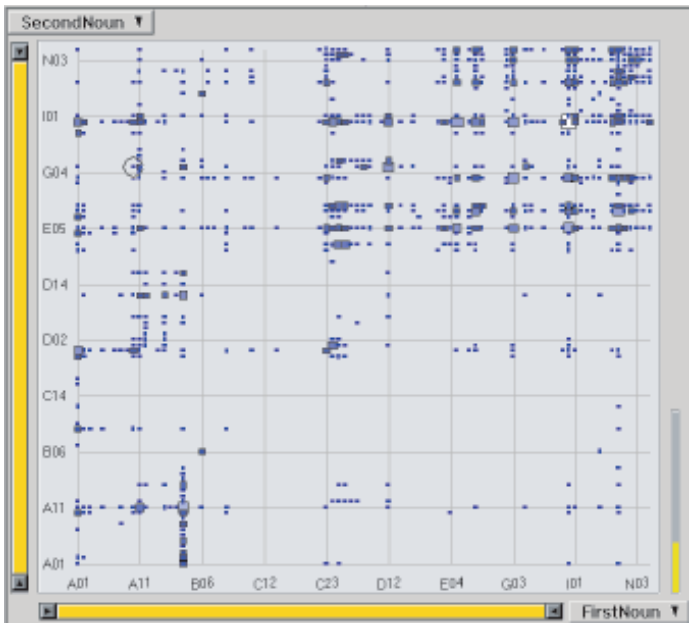


Figure 2.9: Distribution of Level 1 Category Pairs. Mark size indicates the number of unique NCs that fall under the CP. Only those for which > 50 NCs occur are shown.

2.7.5 Labeling NC Relations

Given the promising nature of the NC distributions, the question remains as to whether or not the hypothesis holds. To answer this, I examined a subset of the CPs to see if I could find positions within the sub-hierarchies for which the relation assignments for the member NCs are always the same.

2.7.5.1 Method

I first selected a subset of the CPs to examine in detail. For each of these I examined, by hand, 20% of the NCs they cover, paraphrasing the relation between the nouns, and seeing if that paraphrase was the same for all the NCs in the group. If it was the same, then the current levels of the CP were considered to be the correct levels of description. If, on the other hand, several different paraphrases were found, then

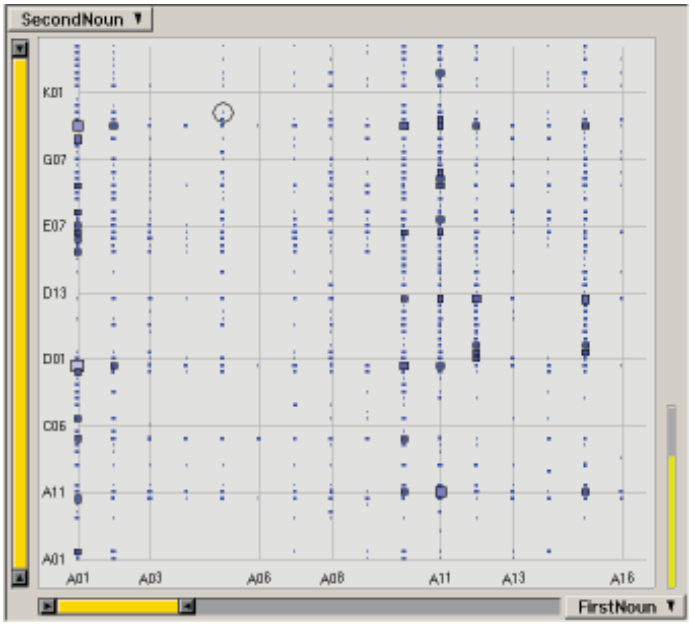


Figure 2.10: Distribution of Level 1 Category Pairs in which the first noun is from the A (Anatomy) category. Mark size indicates the number of unique NCs that fall under the CP.

the analysis descended one level of the hierarchy. This repeated until the resulting partition of the NCs resulted in uniform relation assignments.

For example, all the following NCs were mapped to the same CP, A01 (Body Regions) and A07 (Cardiovascular System): *scalp arteries, heel capillary, shoulder artery, ankle artery, leg veins, limb vein, forearm arteries, finger capillary, eyelid capillary, forearm microcirculation, hand vein, forearm veins, limb arteries, thigh vein, foot vein.* All these NCs are "similar" in the sense that the relationships between the two words are the same; therefore, I do not need to descend either hierarchy. I call the pair (A01, A07) a "rule," where a rule is a CP for which all the NCs under it have the same relationship. In the future, when I see an NC mapped to this rule, I will assign this semantic relationship to it.

On the other hand, the following NCs, having the CP A01 (Body Regions) and M01 (Persons), do not have the same relationship between the component words: *abdomen patients, arm amputees, chest physicians, eye patients, skin donor.* The relationships are different depending on whether the person is a patient, a physician or a donor. I therefore descend the M01 sub-hierarchy, obtaining the following clusters of NCs:

A01 M01.643 (Patients): *abdomen patients, ankle inpatient, eye outpatient*
A01 M01.526 (Occupational Groups): *chest physician, eye nurse, eye physician*
A01, M01.898 (Donors): *eye donor, skin donor*
A01, M01.150 (Disabled Persons): *arm amputees, knee amputees.*

In other words, to correctly assign a relationship to these NCs, we needed to descend one level for the second word. The resulting rules in this case are (A01 M01.643), (A01, M01.150) etc. Figure 2.11 shows one CP for which I needed to descend 3 levels.

A01 H01 (Natural Sciences):
A01 H01 *abdomen x-ray, ankle motion*
  A01 H01.770 (Science): *skin observation*
  A01 H01.548 (Mathematics): *breast risk*
  A01 H01.939 (Weights and Measures): *head calibration*
  A01 H01.181 (Chemistry): *skin iontophoresis*
  A01 H01.671 (Physics)
    A01 H01.671.538 (Motion): *shoulder rotations*
    A01 H01.671.100 (Biophysics): *shoulder biomechanics*
    A01 H01.671.691 (Pressure): *eye pressures*
    A01 H01.671.868 (Temp.): *forehead temperature*
    A01 H01.671.768 (Radiation): *thorax x-ray*
    A01 H01.671.252 (Electricity): *chest electrode*
    A01 H01.671.606 (Optics): *skin color*

Figure 2.11: Levels of descent needed for NCs classified under A01 H01.

Another example is the CP J01, A01 (Technology, Industry, and Agriculture and Body Regions): *glass eye, neoprene elbow, rubber fingers, detergent head, biotechnology face*. Here I go down on the first noun, obtaining the CPs:

J01.637 (Manufactured Materials) A01: *glass eye, neoprene elbow, rubber fingers*

J01.516 (Household Products) A01: *detergent head* (probably "wrong" NCs)

J01.937 (Transportation) A01: *boat heel* (wrong)

J01.219 (Commerce) A01: *business backs* (wrong).

Here going down allowed me to distinguish the correct vs. the "wrong" NCs.

In my collection, a total of 2627 CPs at level 1 have at least 10 unique NCs. Of these, 798 (30%) are classified with A (Anatomy) for either the first or the second noun. I randomly selected 250 of such CPs for analysis.

I also analyzed 21 of the 90 CPs for which the second noun was H01 (Natural Sciences); I decided to analyze this portion of the MeSH hierarchy because the NCs with H01 as second noun are frequent in my collection, and because I wanted to test the hypothesis that I do indeed need to descend farther for heterogeneous parts of

MeSH.

Finally, I analyzed three CPs in category C (Diseases); the most frequent CP in terms of the total number of non-unique NCs is C04 (Neoplasms) A11 (Cells), with 30606 NCs; the second CP was A10 C04 (27520 total NCs) and the fifth most frequent, A01 C04, with 20617 total NCs; I analyzed these CPs.

I started with the CPs at level 1 for both words, descending when the corresponding clusters of NCs were not homogeneous and stopping when they were. I did this for 20% of the NCs in each CP. The results were as follows.

For 187 of 250 (74%) CPs with a noun in the Anatomy category, the classification remained at level 1 for both words (for example, A01 A07). For 55 (22%) of the CPs I had to descend 1 level (e.g., A01 M01: A01 M01.898, A01 M01.643) and for 7 CPs (2%) I descended two levels. I descended one level most of the time for the sub-hierarchies E (Analytical, Diagnostic and Therapeutic Techniques), G (Biological Sciences) and N (Health Care) (around 50% of the time for these categories combined). I never descended for B (Organisms) and did so only for A13 (Animal Structures) in A. This was to be able to distinguish a few non-homogeneous subcategories (e.g., milk appearing among body parts, thus forcing a distinction between *buffalo milk* and *cat forelimb*).

For CPs with H01 as the second noun, of the 21 CPs analyzed, we observed the following (level number, count) pairs: (1, 1) (2, 8) (3, 12).

In all but three cases, the descending was done for the second noun only. This may be because the second noun usually plays the role of the head noun in two-word noun compounds in English, thus requiring more specificity. Alternatively, it may reflect the fact that for the examples I have examined so far, the more heterogeneous terms dominate the second noun. Further examination is needed to answer this decisively.

2.7.5.2 Accuracy

I tested the resulting classifications by developing a randomly chosen test set (20% of the NCs for each CP), entirely distinct from the labeled set, and used the classifications (rules) found above to automatically predict which relations should be assigned to the member NCs. An independent evaluator with biomedical training checked these results manually, and found high accuracies: for the CPs which contained a noun in the Anatomy domain, the assignments of new NCs were 94.2% accurate computed via intra-category averaging, and 91.3% accurate with extra-category averaging. For the CPs in the Natural Sciences (H01) I found 81.6% accuracy via intra-category averaging, and 78.6% accuracy with extra-category averaging. For the three CPs in the C04 category I obtained 100% accuracy.

The total accuracy across the portions of the A, H01 and C04 hierarchies that I analyzed were 89.6% via intra-category averaging, and 90.8% via extra-category averaging.

The lower accuracy for the Natural Sciences category illustrates the dependence of the results on the properties of the lexical hierarchy. The method can generalize well if the sub-hierarchies are in a well-defined semantic relation with their ancestors. If they are a list of "unrelated" topics, I cannot use the generalization of the higher levels; most of the mistakes for the Natural Sciences CPs occurred in fact when we failed to descend for broad terms such as Physics. Performing this evaluation allowed me to find such problems and update the rules; the resulting categorization should now be more accurate.

2.7.5.3 Generalization

An important issue is whether this method is an economic way of classifying the NCs. The advantage of the high level description is, of course, that we need to assign by

hand many fewer relationships than if we used all CPs at their most specific levels. My approach provides generalization over the "training" examples in two ways. First, I find that we can use the juxtaposition of categories in a lexical hierarchy to identify semantic relationships. Second, I find we can use the higher levels of these categories for the assignments of these relationships.

To assess the degree of this generalization I calculated how many CPs are accounted for by the classification rules created above for the Anatomy categories. In other words, if we know that A01 A07 unequivocally determines a relationship, how many possible (i.e., present in my collection) CPs are there that are "covered by" A01 A07 and that we do not need to consider explicitly? It turns out that my 415 classification rules cover 46001 possible CP pairs.[15]

This, and the fact that I achieve high accuracies with these classification rules, show that I successfully use MeSH to generalize over unique NCs.

2.7.5.4 Ambiguity

A common problem for NLP tasks is ambiguity. In this work I observe two kinds: lexical (word sense) and "relationship" ambiguity. As an example of the former, *mortality* can refer to the state of being mortal or to death rate. As an example of the latter, *bacteria mortality* can either mean "death of bacteria" or "death caused by bacteria."

In some cases, the relationship assignment method described here can help disambiguate the meaning of an ambiguous lexical item. *Milk* for example, can be both Animal Structures (A13) and Food and Beverages (J02). Consider the NCs *chocolate milk, coconut milk* that fall under the CPs (B06 -Plants-, J02) and (B06, A13). The

[15] Although I began with 250 CPs in the A category, when a descend operation is performed, the CP is split into two or more CPs at the level below. Thus the total number of CPs after all assignments are made was 415.

CP (B06, J02) contains 180 NCs (other examples are *berry wines, cocoa beverages*) while (B06, A13) has only 6 NCs (4 of which with *milk*). Assuming then that (B06, A13) is "wrong," I will assign only (B06, J02) to *chocolate milk, coconut milk*, therefore disambiguating the sense for milk in this context (Beverage). Analogously, for *buffalo milk, caprine milk* I also have two CPs (B02, J02) (B02, A13). In this case, however, it is easy to show that only (B02 -Vertebrates-, A13) is the correct one (i.e. yielding the correct relationship) and I then assign the MeSH sense A13 to *milk*.

Nevertheless, ambiguity may be a problem for this method. I see five different cases:

1. Single MeSH senses for the nouns in the NC (no lexical ambiguity) and only one possible relationship which can predicted by the CP; that is, no ambiguity. For instance, in *abdomen radiography*, *abdomen* is classified exclusively under Body Regions and *radiography* exclusively under Diagnosis, and the relationship between them is unambiguous. Other examples include *aciclovir treatment* (Heterocyclic Compounds, Therapeutics) and *adenocarcinoma treatment* (Neoplasms, Therapeutics).

2. Single MeSH senses (no lexical ambiguity) but multiple readings for the relationships that therefore cannot be predicted by the CP. It was quite difficult to find examples of this case; disambiguating this kind of NC requires looking at the context of use. The examples I did find include *hospital databases* which can be *databases* **regarding** (topic) *hospitals*, *databases* **found in** (location) or **owned by** hospitals. *Education efforts* can be *efforts* **done through** (*education*) or **done to achieve** *education*. *Kidney metabolism* can be *metabolism* **happening in** (location) or **done by** the *kidney*. *Immunoglobulin staining*, (D12 -Amino Acids, Peptides-, and Proteins, E05 -Investigative Techniques-) can mean either *staining* **with** *immunoglobulin* or *staining* **of** *immunoglobulin*.

3. Multiple MeSH mappings but only one possible relation. One example of this case is *alcoholism treatment* where *treatment* is Therapeutics (E02) and *alcoholism* is both Disorders of Environmental Origin (C21) and Mental Disorders (F03). For this NC we have therefore 2 CPs: (C21, E02) as in *wound treatments, injury rehabilitation* and (F03, E02) as in *delirium treatment, schizophrenia therapeutics*. The multiple mappings reflect the conflicting views on how to classify the condition of alcoholism, but the relationship does not change.

   Another example along this line is *milk temperature*. *Milk* is Animal Structures (A13) and Food and Beverages (J02), *temperature* is Environment and Public Health (G03) and Natural Science (H01). For this NC we have therefore 4 CPs but the relationship is the same for all of of them.

4. Multiple MeSH mappings and multiple relations that *can* be predicted by the different CPs. For example, *Bread diet* can mean either that a person usually eats *bread* or that a physician prescribed *bread* to treat a condition. This difference is reflected by the different mappings: *diet* is both Investigative Techniques (E05) and Metabolism and Nutrition (G06), *bread* is Food and Beverages (J02). We have therefore two CPs for this NC: (J02 E05) in which case we would have the relationship "prescription of" and (J02 G06) for which we have "usual nutrition." These different relationships are correctly predicted by the different senses for *diet*; in these cases, the category can help disambiguate the relation (as opposed to in case 5 below); word sense disambiguation algorithms that use context may be helpful.

5. Multiple MeSH mappings and multiple relations that *cannot* be predicted by the different CPs. As an example of this case, *bacteria mortality* can be both "death of bacteria" or "death caused by bacteria." The multiple mapping for *mortality* (Public Health, Information Science, Population Characteristics and

Investigative Techniques) does not account for this ambiguity. Similarly, for *inhibin immunization*, the first noun falls under Hormones and Amino Acids, while *immunization* falls under Environment and Public Health and Investigative Techniques. The meanings are *immunization* **against** *inhibin* or *immunization* **using** *inhibin*, and they cannot be disambiguated using only the MeSH descriptors.

I currently do not have a way to determine how many instances of each case occur. Cases 2 and 5 are the most problematic; however, as it was quite difficult to find examples for these cases, I suspect they are relatively rare.

A question arises as to if representing nouns using the topmost levels of the hierarchy causes a loss in information about lexical ambiguity. In effect, when I represent the terms at higher levels, I assume that words that have multiple descriptors under the same level are very similar, and that retaining the distinction would not be useful for most computational tasks. For example, *osteosarcoma* occurs twice in MeSH, as C04.557.450.565.575.650 and C04.557.450.795.620. When described at level 1, both descriptors reduce to C04, at level 2 to C04.557, removing the ambiguity. By contrast, *microscopy* also occurs twice, but under E05.595 and H01.671.606.624. Reducing these descriptors to level 1 retains the two distinct senses.

To determine how often different senses are grouped together, we calculated the number of MeSH senses for words at different levels of the hierarchy. Table 2.10 shows a histogram of the number of senses for the first noun of all the unique NCs in our collection, the average degree of ambiguity and the average description lengths.[16] The average number of MeSH senses is always less than two, and increases with length of description, as is to be expected.

I observe that 3.6% of the lexical ambiguity is at levels higher that 3, 16.0% at

[16]I obtained very similar results for the second noun.

level 3, 21.4% at level 2 and 59.0% at level 1. Level 2 and 3 combined account for more than 80% of the lexical ambiguity. This means that when a noun has multiple senses, those senses are more likely to come from different main subtrees of MeSH (A and B, for example), than from different deeper nodes in the same subtree (H01.671.538 vs. H01.671.252). This fits nicely with my method of describing the NCs with the higher levels of the hierarchy: if most of the ambiguity is at the highest levels (as these results show), information about lexical ambiguity is not lost when we describe the NCs using the higher levels of MeSH. Ideally, however, we would like to *reduce* the lexical ambiguity for similar senses and to *retain* it when the senses are semantically distinct (like, for example, for *diet* in case 4). In other words, ideally, the ambiguity left at the levels of my rules accounts for only (and for all) the semantically different senses. Further analysis is needed, but the high accuracy I obtained in the classification seems to indicate that this indeed is what is happening.

2.7.5.5 Multiple Occurrences of Semantic Relations

Because I determine the possible relations in a data-driven manner, the question arises of how often does the same semantic relation occur for different category pairs. To determine the answer, I could (i) look at all the CPs, give a name to the relations and "merge" the CPs that have the same relationships; or (ii) draw a sample of NC examples for a given relation, look at the CPs for those examples and verify that all the NCs for those CPs are indeed in the same relationship.

I may not be able to determine the total number of relations, or how often they repeat across different CPs, until I examine the full spectrum of CPs. However, I did a preliminary analysis to attempt to find relation repetition across category pairs. As one example, we hypothesized a relation "Afflicted by" and verified that it applies to all the CPs of the form (Disease C, Patients M01.643), e.g.: *anorexia (C23) patients, cancer (C04) survivor, influenza (C02) patients.* This relation also applies to some

# Senses	Original	L3	L2	L1
1 (Unambiguous)	51539	51766	54087	58763
2	18637	18611	18677	17373
3	5719	5816	4572	2177
4	2222	2048	1724	1075
5	831	827	418	289
6	223	262	167	0
7	384	254	32	0
8	2	2	0	0
9	61	91	0	0
10	59	0	0	0
Total(Ambiguous)	28138	27911	25590	20914
Avg # Senses	1.56	1.54	1.45	1.33
Avg Desc Len	3.71	2.79	1.97	1

Table 2.10: The number of MeSH senses for N1 when truncated to different levels of MeSH. Original refers to the actual (non-truncated) MeSH descriptor (C04.557.450.565.575.650, for example). Avg # Senses is the average number of senses computed for all first nouns in the collection. Avg Desc Len is the average description length; the value for level 2 is less than 2 and for level 3 is less that 3, because some nouns are always mapped to higher levels (for example, *cell* is always mapped to A11). L1 is level 1 (C04), L2 is level 2 (C04.557) and L3 is level 3 (C04.557.450).

of the F category (Psychiatry), as in *delirium (F03) patients, anxiety (F01) patient.*

It becomes a judgement call whether to also include NCs such as *eye (A01) patient, gallbladder (A03) patients*, and more generally, all the (Anatomy, Patients) pairs. The question is, is "afflicted-by (unspecified) Disease in Anatomy Part" equivalent to "afflicted by Disease?" The answer depends on one's theory of relational semantics.

Another relation could be "Time of": *abscess recurrence* (C23, C23.550) *hockey season* (I03, G03.230) *pollen season* (B06, G03.230), *acceleration time* (H01, H01.862), *dialysis time* (E05, H01.862), *convalescence time* (C23, H01.862).

The relation "Person/center who treats" apply, for example, to *arm physicians, eye nurse* and in general to the CPs (Anatomy, Occupational Groups), to *arthritis physician, disease caregiver, stroke investigators, asthma nurse* (Diseases, Occupational Groups), *inpatient nurses* (Patients, Occupational Groups), *adolescent hospital, women hospital* (Persons, Health Facilities), *tuberculosis hospital, aids laboratory, trauma hospital* (Diseases, Health Facilities) and *heart hospital, eye hospital* (Anatomy, Health Facilities).

"Caused by" apply to *cirrhosis death, infection death* (Disease, Death) *virus infection, virus fever* (Viruses, Diseases).

As another example, consider the relationship "Located in" like in *hospital dust, laboratory air* (Health Facilities, Environment), *hospital nurse* (Health Facilities, Occupational Groups), *arm fistulas, abdomen wound* (Anatomy, Diseases) *ankle bone, arm muscle* (Anatomy, Anatomy), *city children* (Geographic Locations, Persons) and *brazil nuts* (Geographic Locations, Plants).

2.7.6 Conclusions about the "hierarchy approach"

I provided evidence that the upper levels of a lexical hierarchy can be used to accurately classify the relations that hold between two-word technical noun compounds.

Here I focus on biomedical terms using the biomedical lexical ontology MeSH. It may be that such technical, domain-specific terminology is better behaved than NCs drawn from more general text; this would require assessment of the technique in other domains to fully assess its applicability.

It is also necessary to ensure that this technique works across the full spectrum of the lexical hierarchy. I have demonstrated the likely usefulness of such an exercise, but all of our analysis was done by hand. It may be useful enough to simply complete the job manually; however, it would be preferable to automate some or all of the analysis. There are several ways to go about this. One approach would be to use existing statistical similarity measures (see, e.g., Budanitsky and Hirst, 2001) to attempt to identify which subhierarchies are homogeneous. Another approach would be to see if, after analyzing more CPs, those categories found to be heterogeneous should be assumed to be heterogeneous across classifications, and similarly for those that seem to be homogeneous.

2.8 Conclusions

Technical text is especially rich in noun compounds and any language understanding program needs to be able to interpret them. My work on noun compounds is an important part of a larger effort to investigate the extraction of semantics from text. In this chapter, I discussed the problem of the assignment of semantic relations for noun compounds and I proposed two approaches for tackling this problem.

The main idea of the first approach is to define a set of relations that can hold between the terms and use standard machine learning techniques and a lexical hierarchy to train a classification system (see Section 2.6). The results are quite promising: I achieved an accuracy of about 60% for multi-class classification with 18 classes. I also showed that a class-based representation performs as well as a lexical-based model

despite the reduction of raw information content and despite a somewhat errorful mapping from terms to concepts.

The second approach explores the possibility of using that same lexical hierarchy but this time without statistics and machine learning. I show that mere membership within a particular subbranch of the hierarchy is sufficient in many cases for assignment of the appropriate semantic relation (Section 2.7). I find that the topmost levels of the hierarchy yield an accurate classification, thus providing an economic way of assigning relations to noun compounds.

An important issue left unaddressed in this work is how to extend the technique to multi-word noun compounds such as *acute migraine treatment* and *oral migraine treatment*.

Chapter 3

Role and relation identification

3.1 Introduction and problem description

In Chapter 2, I described a system for the classification of the semantic relations that held between two words in noun compounds. In this chapter, I extend this analysis to the sentence level. Specifically, I examine the problem of identifying entities of types "treatment" and "disease" in bioscience text and the problem of distinguishing among seven relation types that can occur between them. These tasks are considered part of the general problem of "information extraction."

In particular, I developed algorithms to tackle the following problems (Rosario and Hearst, 2004):

(1) **Named entity recognition**

The entities of interest for this chapter are *treatment* and *disease* (TREAT and DIS throughout this chapter). For example, given a phrase such as

```
The fluoroquinolones for urinary tract infections:  a review.
```

I want to extract all and only the strings of text that correspond to the semantic

roles TREAT (*fluoroquinolones*) and DIS (*urinary tract infections*). (This task is also called in the literature "role, entity or information extraction.")

(2) **Relation recognition**

To identify the type of relations that hold between the semantic roles in a sentence. Given the sentence above, I want the system to classify the sentence as containing a *cure* relationship.

Relation recognition is important because very often we are interested not only in the entities but in how the entities are related to each other (for example, cells being part of an organ) or in the effect that one entity has on another (a protein inhibiting another protein or a drug curing a disease). While the entities are often realized as noun phrases, the relationships often correspond to grammatical functional relations.

Often the different relationships are determined by the entities involved, for example an ORGANIZATION and a LOCATION could be in a *location of* type of relation and an ORGANIZATION and an EMPLOYEE could be related by *employer of* but not by *location of*, in which one entity must always be a LOCATION.[1] A more difficult and interesting case is when several different relations can hold between the same pair of entities; in this case, the recognition of the entities does not fully disambiguate the relations. For example, in the following sentences, *hepatitis*, which is a DIS, can be in different semantic relationships with the TREAT present in the sentences. In

```
Effect of interferon on hepatitis B.
```

there is a unspecified effect of the treatment *interferon* on the disease *hepatitis*. In the following sentence, the *vaccine* **prevents** *hepatitis*

```
A two-dose combined hepatitis A and B vaccine would facilitate
immunization programs.
```

[1] I denote the entities with capital letters; also, I interchange the terms entity, semantic class, class, role but I'll assume they all mean the same thing.

while in:

> `These results suggest that con A-induced hepatitis was ameliorated by pretreatment with TJ-135.`
>
> `Therefore administration of TJ-135 may be useful in patients with severe acute hepatitis accompanying cholestasis or in those with autoimmune hepatitis.`

the disease *hepatitis* is **treated** or **cured** by the treatment *TJ-135*. The disease can also occur alone, without a treatment, as in *hepatitis* below:

> `Histologic diagnosis of chronic hepatitis, grading and staging`

The different relations can be expressed in different ways. They can be expressed as nominalizations, as in

> `The treatment of diabetes 2 by the classic oral antidiabetic drugs (sulfamides and biguanides)`

as well as verbal predications, as in

> `The use of zinc lozenges to treat cold symptoms deserves further study.`

Also, the entities often entail relational information just by virtue of their semantics, as in the following sentences where we can infer that there is a **cure** kind of relationship only because we know the meanings of the words.

> `Elective surgery for colorectal cancer in the aged: a clinical - economical evaluation.`
>
> `Headache drugs.`

In this chapter, I describe my work on developing algorithms to identify the entities and the relationships between them. I compare five generative graphical models and a neural network, using lexical, syntactic, and semantic features. I find the latter particularly helpful for achieving high classification accuracy.

The remainder of the chapter is organized as follows: in the next section, I discuss related work (this discussion will be relevant to Chapter 4, in which I tackle the problem of labeling protein-protein interactions); Section 3.3 describes the data, the semantic relations, how the annotation was done and the evaluation. In Section 3.5, I discuss the features used. Finally, in Section 3.7 I present the models implemented for these tasks and the results they achieved.

3.2 Related work

This section describes the related work for the tasks of information and relation extraction (in this order). There are several dimensions along which we can analyze the related work. Here, I report on the domains that have been tackled, on the models that have been used, on the use of syntactic information and on the use of unlabeled data.

Most of the related work on relationship extraction assumes that the entity extraction task has been performed by another system and the entities of interests therefore are given as input. Note that the models of Section 3.7 do not make this assumption and indeed perform role and relation extraction simultaneously.

3.2.1 Domains tackled

We can roughly divide the research on IE into work that tackles general text and work specific to the bioscience domain.

For the domain-independent approaches, most of the research in IE has focused on the tasks in the MUC (Message Understanding) conferences. These conferences, which focus on the evaluation of information extraction systems applied to a common task, were funded by the government agency ARPA to measure progress in information extraction.[2] MUC-4 (1992) was about extracting information about terrorist events in Latin America from newswire articles (Riloff, 1993) and MUC-5 (1993) about joint ventures. MUC-6 (1995) and MUC-7 (2001) involved the recognition of entity names (people and organizations, for a management succession scenario), place names, temporal expressions, and certain types of numerical expressions (Borthwick et al., 1998; Bikel et al., 1999); MUC-7 was about air crashes and missile launches.

The ACE competition (Automatic Content Extraction)[3] is devoted to three types of sources: newswire, broadcast news (with text derived from ASR), and newspaper (with text derived from OCR); see Culotta and Sorensen (2004) for a paper that tackles this data.

Other work involves the extraction of locations and organizations (Agichtein and Gravano, 2000), and the extraction of speakers and locations from seminar announcements, of company names and job titles from Usenet job announcements and information about corporate acquisitions (Freitag and McCallum, 2000).

Gildea and Jurafsky (2002) and Thompson et al. (2003) address the problem of extracting semantic roles that are at an intermediate level between very general roles such as AGENT and PATIENT and those specific to individual verbs or specific situations (such as the terrorist in MUC or genes names in bioscience). These can be described by the *frame* level from the FrameNet project (Baker et al., 1998).

Turning now to the bioscience domain, although a huge amount of biological information is available in electronic form, most of it is unstructured text in MEDLINE

[2]See for example, http://www.cs.nyu.edu/cs/faculty/grishman/muc6.html
[3]http://www.itl.nist.gov/iad/894.01/tests/ace/index.htm

citations and full-text journals; methods to automatically process this information are greatly needed. In the IE field, most of the work so far has focused on extracting names of genes and proteins but also DNA, RNA, mRNA (Craven and Kumlien, 1999; Stapley and Benoit, 2000; Rindflesch et al., 2000a; Collier et al., 2000; McDonald and Pereira, 2005) and protein-protein interactions (Blaschke et al., 1999b; Craven, 1999; Pustejovsky et al., 2002; Rindfleisch et al., 1999). These papers are discussed in more detail later in this thesis.

3.2.2 Entity Extraction: Models

There are many IE systems that are rule-based: Appelt et al. (1993), for example, or Feldman et al. (2002) and Friedman et al. (2001) in the bioscience domain. However, this thesis is mainly concerned with *statistical* models and in what follows, I distinguish between several types of these.

3.2.2.1 Pattern Matching

"Pattern Matching" systems are not actually statistical, but in this section I describe some of them because they are an important part of the IE research, and because, at least, they *automatically* construct rules and often choose them on the basis of statistics (as opposed to systems for which the rules are hand-written).

The AutoSlog system (Riloff, 1993) constructs dictionaries for IE by creating extraction patterns automatically using heuristic rules. It relies on labeled data with domain specific tags.[4] Some heuristic rules (defined manually) are applied to the labeled text; an extraction pattern is created by instantiating the rule with the specific words that it matched in the sentence. For example, in the sentence

Ricardo Castellar, the mayor, was kidnapped yesterday by the FMLN.

[4]The domain described in the paper is on "terrorist events" and the training data was the MUC-4 corpus.

Ricardo Castellar was labeled as VICTIM. When the system is exposed to this sentence, all of the heuristic rules are tested and the rule "<subject> passive-verb" is found to match. The instantiated pattern that results is: "VICTIM was kidnapped." In new text, this pattern will be instantiated every time the verb "kidnapped" appears in a passive construction and the subject will be extracted as a VICTIM. The generated patterns are then manually inspected by a person who decides which ones to keep and which ones to reject. An F-measure of 50.51% is reported.

Soderland et al. (1995) describe CRYSTAL, a system similar to AutoSlog, in that it also automatically constructs dictionaries for IE by learning extraction patterns from labeled data. The task is to extract the fields DIAGNOSIS and SIGN OR SYMPTOM from hospital discharge reports. CRYSTAL allows more expressive pattern extraction patterns than AutoSlog: it considers the semantic classes of the words, and a matching occurs if the semantic classes are matched. For example, a pattern such as "PATIENT denies SIGN OR SYMPTOM" would extract the sentence "*The patient denies any episodes of nausea*" because *nausea* is a SIGN OR SYMPTOM, but would not extract "*She denies any history of asthma*" because *asthma* belongs to the semantic class DISEASE which is not a sub-class of SIGN OR SYMPTOM. The semantic classes used are those in the Semantic Network of UMLS.[5] Given an initial set of patterns obtained from the labeled data (that are actually called "concept nodes in Soderland et al., 1995), CRYSTAL gradually relaxes the constraints on these initial patterns to broaden their coverage, and it merges similar definitions to form a more compact dictionary (these steps also take into account the semantic classes). This is an improvement with respect with AutoSlog. Another similar system along these lines is WAVE, described in Aseltine (1999), that like CRYSTAL relies on unification to build more general patterns (but how the generalization is controlled is

[5]The Unified Medical Language System (UMLS) is a biomedical lexical resource produced and maintained by the National Library of Medicine (Humphreys et al., 1998).

different); WAVE is a on-line system (it learns from a stream of training instances) while AutoSlog and CRYSTAL are batch algorithms. These systems can be (and in fact, often are) the first step of bootstrapping systems (see Section 3.2.5.4).

3.2.2.2 Hidden Markov Models (HMMs)

HMMs are generative graphical models that are naturally suited for segmentation tasks. Many papers have been written on the use of HMMs for information extraction. HMMs are a natural solution for tasks involving discrete sequences, as is the case of IE (see Rabiner and Juang, 1986, for a nice short description of HMMs). My work on the dynamic models described in Section 3.7.1 was inspired by the HMM models.

Freitag and McCallum (2000) address the problem of (automatic) selection of HMM state-transition structure and show that this improves the accuracy on some IE tasks. They begin with a minimum number of states, generate a set of structures by various splitting state operations and choose the structures that give the best performances on a validation test; they do this procedure iteratively. They tested this approach on eight IE tasks (for example, extraction of JOBS from Usenet job announcements, SPEAKER and LOCATION from a collection of seminar announcements). They report an average F-measure of 57% across all tasks, and they show this is an improvement from the results obtained by other methods. This is an interesting paper that addresses an important problem of HMM formulation. It would have been nice, however, to see a comparison with a fully connected HMM in which the structure is learned via parameter estimation,[6] especially given that the number of states for the problem stated in this paper is small enough. They compare their system to a "simple HMM," which is the model with which the structure selection begins; this is a model with 4 states (backgrounds, prefix, suffix and target) with the structure defined by hand. Freitag and McCallum (2000) note how this simple model

[6]The zeros in the state transition correspond to missing links between the states.

out-performs the other methods (rule-learning approaches) for certain tasks. It would have been natural to investigate the performance of a simple 4-state fully connected model trained with ML, without structure selection. This is actually in part done by Seymore et al. (1999) in which another way of doing structure selection is introduced and compared with the ML model; they show that one (automatic) way of doing structure selection is essentially equivalent to the ML model (90.6% in accuracy for the structure selection method and 90.5% for ML) while they get an improvement in accuracy (91.3%) with another partially *manual* structure selection. In my view, these papers have not shown that automatic structure selection is really beneficial, at least for these models and this task.

Freitag and McCallum (1999) introduce a smoothing technique (called shrinkage, also known as "deleted interpolation") to address the problem of limited training data. They apply the smoothing to the emission probabilities, by creating four hierarchical shrinkage configurations and by using EM to find the optimal values of the mixing components.

Bikel et al. (1999) present an interesting variant of the HMM: the class C of time t depends on both the class and observation (word, w) at time $t-1$, which can be written as $P(C \mid C_{-1}, w_{-1})$, and the first word of a class depends on its class and on the previous class ($P(w_{first} \mid C, C_{-1})$), and all the subsequent words inside the class depend on the class and on the previous word ($P(w \mid C, w_{-1})$). The model is represented in Fig 3.1. They report an impressive F-measure of 94.9% for the task of extracting names of persons and organizations, locations, dates, and numerical quantities.

Ray and Craven (2001) hypothesize that incorporating some information about the sentence structure may be beneficial for the task of IE. They represent a sentence as a sequence of phrases, where a phrase consists of a grammatical type (NP, VP, PP,

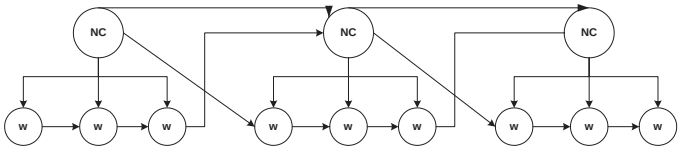


Figure 3.1: The HMM-like model in Bikel et al. (1999).

...) and the words that are part of that phrase.[7] They use an HMM whose states are <type, label> tuples (for example, <NP, LOCATION>) and whose observations are the sequences for words for that phrase (for example, <*the endoplasmic reticulum*>). To the best of my knowledge, this is the only work that analyzes the contribution of syntactic information in the HMM framework. They train a positive and a null model. The observation probabilities $P(o \mid q)$, where o is the observation and q the state, become $\prod_{j=1}^{|p_i|} P(o_j \mid q)$ where p_i is the number of words emitted for that phrase and o_j is the j^{th} word in the phrase (this is essentially a Naive Bayes model for the words in the phrase). They compare this "phrase" model with two "token" models, one that includes the part-of-speech of the words and one with only words and nonsyntactic information, and report better results for "phrase," suggesting that there is value in representing the grammatical function for this task. They also investigate the use of discriminative training and report improvements in accuracy for the same levels of recall. The overall results, however, are quite poor; their results graphs show, approximately, an F-measure of 26% at 2% recall and an F-measure of 32% at 35% recall for the relationship *subcellular location* and an F-measure of 50% for *disorder-association*. (These results are for the best cases of the "phrase" model and discriminative training.)

Other recent papers on the use of HMM for IE are Zhou and Su (2002) and Collier et al. (2000). A model closely related to HMM is proposed by Thompson et al. (2003)

[7] They use the Sundance system (Riloff, 1998) to obtain a shallow parse of the text.

to both extract the (FrameNet) semantic roles and determine the frame. The models I propose for this thesis (see Section 3.7) are very similar to these HMM models and to that in Thompson et al. (2003).

3.2.2.3 Discriminative Models

Unlike generative approaches that model the joint probability distribution of the features and the classes (and compute the posterior probability of classes given features from the joint probability), discriminative methods directly learn a mapping from the features to classes (often using some form of kernel function or other "similarity measure"). A special case of the latter are discriminative methods that directly learn the posterior probability of the class given the features; these posterior probabilities can be thresholded to obtain a classification decision. Ng and Jordan (2002) indicate that with fully observed data for classification tasks a discriminative approach may be more appropriate, while generative models are the natural choice with partially observed or missing data. Since labeled data is expensive to collect, generative models may be useful when no labels are available.

Klein and Manning (2002) compare two model structures, HMM and an "upward" conditional Markov model (the same as McCallum et al., 2000, described below) on the task of part-of-speech tagging and note how the "independence assumption embodied by the conditional structure" resulted in a lower accuracy for this model, compared with the HMM. Klein and Manning (2002) also note that there is a dysfunctional behavior, a "reverse" explaining away-phenomenon, the *observation bias* that happens when an observation explains its state so well that the previous state is essentially ignored. They show how in fact it is the observation bias that actually contributed to the tagging error, rather than the *label bias* (which is another dysfunctional behavior, another explaining away-phenomenon: when the previous state explains the current one so well, the observation at the current state is ignored).

They claim that the model structure is an important factor and that local conditional structures are worse than generative models.

This approach is taken by Gildea and Jurafsky (2002) for FrameNet (see discussion in Section 3.2.2.5) and Collins (2002) for detecting named entities boundaries. McDonald and Pereira (2005) use conditional random fields (CRF) to extract genes and proteins. CRFs are very closely related to the maximum entropy markov models described in the next section; their relationship is discussed in some detail in McDonald and Pereira (2005).

3.2.2.4 Maximum Entropy Models

Maximum entropy markov models are also conditional models; in this section I describe some of the work done using this formalism because these models have become increasingly popular in statistical language processing, mainly because of their ability to incorporate a richer set of (possibly dependent) features (see Berger et al., 1996, for a nice overview/introduction of maximum entropy models).

McCallum et al. (2000) present a model similar to the HMM based on the maximum entropy framework, MEMM, Maximum Entropy Markov Model. The two main original contributions of this paper are:
1) The model represents a conditional probability of the state sequence *given* the observations; (by inverting the arrows from the observations to the states); this is done to address the problem of the HMM that uses a *generative* model while, in most cases (e.g. for IE), we are really interested in solving a *conditional* problem.
2) The conditional probabilities $P(s_t \mid o_t, s_{t-1})$ are parameterized by an exponential model that allows for a richer representation in terms of many overlapping features.

They investigate segmenting Usenet FAQ-s into HEAD, QUESTION, ANSWER, TAIL (which I think is an easier problem than the classical role extraction). The segmentation is made at the line level, and in fact all the features are line-based fea-

tures. I do not think this experiment shows that this model would be appropriate for a "real" IE task. Moreover, regarding point 1), it has been pointed out (by for example, Lafferty et al., 2001) that conditionally structured models suffer from the *label bias problem* which is a dysfunctional behavior, the "explaining away" phenomenon, that essentially favors state sequences that have no relations with the actual observations. In other words, when the state at time $t-1$ explains "very well" the state at time t, then the observation at time t is ignored.

Lafferty et al. (2001) propose a *conditional random field model* to address this problem. Conditional random field models are indirect graphical models that have a *single* exponential model for the joint probability of the entire sequence of states given the observations, $P(s_0, s_1, ...s_T \mid o_0, ...o_T)$.[8] The claim is that the weights of the different features at different states can be traded off against one other, therefore avoiding the label bias problem. They apply this model to synthetic data and to a part-of-speech tagging task. McCallum and Li (2003) applied this model to named entity extraction on the CoNLL-03 shared task.

Borthwick et al. (1998) use the maximum entropy model for the task of Named Entity recognition. The results reported are very good (F-measure of 97.12%) for the MUC-7 corpus (25 articles mainly on aviation disasters) but it is not clear what the role of the maximum entropy models was versus that of the inclusion of a rich pool of features.[9]

[8] In contrast, MEMM models T conditional probabilities $P(s_t \mid o_t, s_{t-1})$.

[9] They use binary features (such as "the token begins with a capitalized letter," "the token contains a number"), lexical features, section features, dictionary features (they use 8 dictionaries) and external features (i.e., features provided by -3- external systems, like the predictions for the named entities made by these external systems).

3.2.2.5 Parsing Models

"Parsing Models" are models that incorporate both syntactic and semantic information in the models themselves (as opposed to, for example, the syntactic information being a feature), usually by augmenting the parse tree with semantic labels on the tree nodes.

Miller et al. (2000) use the syntactically annotated Penn Treebank corpus and annotate the semantics (in terms of both entities and relationships) on top of it. They call their model the "sentence-level model." They first train this model on the purely syntactic Treebank trees. Then, for each sentence of the semantically annotated corpus, they:

1. Apply the sentence-level model to syntactically parse the tree, *constraining the model to produce only parses that are consistent with the semantic annotation* (for example, by prohibiting the model to produce parses for which a semantic constituent would be broken up by the syntactic constituents).

2. Augment the resulting tree to reflect both the syntactic and semantic structure (for example, by inserting nodes that describe the named entities or the arguments to the relations).

3. Retrain the sentence-level model on the augmented tree. At this point they have an integrated model of syntax and semantics.

They define the "structure" of the model by deciding, for example, that the categories for the head constituents depend only on the categories of the parent node, and that the part-of-speech-tags for a modifier depend on the category of the modifier, on the part-of-speech tag of the head-word and on the head-word itself. They train the model with ML estimates and smoothing and apply dynamic programming to find the most likely parse tree given a new sentence. They also apply this model to the

MUC-7 corpus and report an F-measure of 83% for the entity extraction and 64% for the relationship extraction.

Chelba and Mahajan (2001) use a very similar model that integrates syntactic and semantic information; a syntactic parser is trained to match the semantic constituents and then used to recover the most likely parse tree (and therefore the semantic labels, since the tree is augmented with semantic tags) given test sentences. Collins and Miller (1997) use a probabilistic context free grammar for the task of IE in the management succession domain of MUC-6.[10] They assume that the state sequence is generated by the application of r context-free rules $LHS_i \Rightarrow RHS_i$[11] and that $P(s_1 s_2 ... s_n) = \prod_{i=1..r} P(RHS_i \mid LHS_i)$. They recover the (quite complex) underlying tree structure from the training data labels and they use ML estimates for all the rules (with a backing-off smoothing); they report an accuracy of 77.5%.

Gildea and Jurafsky (2002) separate the tasks of parsing and entity extraction. They first run a parser on the text and then use the features obtained from the parser (such as phrase type, grammatical function, voice and the head word) to find the probability of a semantic role given these features; they extract FrameNet roles (Baker et al., 1998).

3.2.3 Relation Recognition

Relation recognition is the task of recognizing the relationships between entities. If between two semantic entities only one relationship is possible, then the task of relation recognition essentially coincides with the task of entity extraction. While this may be true with some approximation for certain entities, the most interesting case is when two or more entities can be in several different relations and we need to

[10] The entities extracted are the position, the person leaving the position and the person coming into the position.

[11] LHS is the left hand side and RHS is the right hand side.

distinguish between those (like my examples for TREAT and DIS in Sections 3.3). Most of the related work in this field does not consider this case: sometimes co-occurrences of entities are found and the relation is implied, as in Ray and Craven (2001), (reducing therefore this task to the related problem of entity extraction); similarly, the ACE competition[12] has a relation recognition subtask, but assumes a particular type of relation holds between particular entity types (e.g., if the two entities in question are an EMP and an ORG, then an employment relation holds between them; which type of employment relation depends on the type of entity, e.g., staff person vs. partner).

The related work on relation classification can (roughly) be divided into three approaches, (i) rules and templates to match linguistic patterns, (ii) co-occurrences of entities and (iii) machine learning methods.

In the BioNLP literature, there have recently been a number of attempts to automatically extract protein-protein interactions from bio-medical text; Chapter 4 describes my work in this field.

Some approaches simply report that a relation exists between two proteins but do not determine which relation holds (Bunescu et al., 2005; Marcotte et al., 2001; Ramani et al., 2005), while most others start with a list of interaction verbs and label only those sentences that contain these trigger verbs (Blaschke and Valencia, 2002; Blaschke et al., 1999b; Rindflesch et al., 1999; Thomas et al., 2000; Sekimizu et al., 1998; Ahmed et al., 2005; Phuong et al., 2003; Pustejovsky et al., 2002). However, as Marcotte et al. (2001) note, *"... searches for abstracts containing relevant keywords, such as "interact*[13]*", poorly discriminate true hits from abstracts using the words in alternate senses and miss abstracts using different language to describe the interactions."*

[12] http://www.itl.nist.gov/iaui/894.01/tests/ace/

[13] Where the character "*" matches everything beginning with the string *interact*.

Most of the existing methods also suffer from low recall because they use hand-built specialized templates or patterns (see next section). Moreover, as Blaschke and Valencia (2002) note, most approaches use pre-defined lists of protein names rather than deal with the difficult problem of protein name recognition. In most cases, papers evaluate on their own test set, and so it is quite difficult to compare systems.

3.2.3.1 Rule-based and pattern systems for relation classification

In the bioscience domain the work on relation classification is primary done through hand-built rules. Feldman et al. (2002) use hand-built rules that make use of syntactic and lexical features and semantic constraints to find relations between genes, proteins, drugs and diseases. The GENIES system (Friedman et al., 2001) uses a hand-built semantic grammar along with hand-derived syntactic and semantic constraints, and recognizes a wide range of relationships between biological molecules. An evaluation comparing their system to an expert's assessment of 57 test sentences containing binary relations (all taken from one article, so on a focused topic) yielded a precision of 96%. Examples of templates are: `P1[(,CC DT)|,(IN)|:|;]P2` (Ono et al., 2001) or `'interaction of' (PROTEIN_1) 'and' (PROTEIN_2)` (Corney et al., 2004). Some systems use link grammars in conjunction with trigger verbs instead of templates (Ahmed et al., 2005; Phuong et al., 2003).

Pustejovsky et al. (2002) extract *inhibit*-relations. Automata were developed for the extraction of the entities and the relation from a shallow parsed representation of the text. A precision of 94% and a recall of 58.9% are reported. The experiments were based on 95 sentences from MEDLINE that contained verbal and nominal forms of the stem *inhibit*. Note that therefore, the real task here is to extract entities that are connected by some form of the stem *inhibit*, which is arguably different from the extraction of entities in the *inhibit*-relation, if there are other linguistic ways to express this relation (but Pustejovsky et al., 2002, do not discuss this issue). Focusing on one

very specific syntactic structure can result in a system with very few false positives (i.e., high precision), which is what we may want in some cases, however, such a system will also have poor recall.

In Saric et al. (2004), the goal is to find all the proteins that are responsible for regulating the expression of which genes, i.e., the relation is fixed and the systems is supposed to find all its instances; they use finite state automata and report an accuracy of 83% but no information about recall.

Ng and Wong (1999) also develop a set of rules that models very simple sentence patterns ("A inhibits B, C, and D," "A, an activator of B, ...") for the extraction of protein-protein interactions, again, given a fixed list of key function words. Friedman et al. (2001) is more complex than Ng and Wong (1999) and has a broader set of biological relations but the underlying method is the same: manually developed grammar rules to recognize well-formed patterns and to generate target output.

Blaschke et al. (1999b) claim to extract protein-protein interactions, but impose a number of very strong assumptions: *both* protein names are specified by users[14] and an instance of a set of 14 pre-specified words (such as *activate, interact, suppress*) must be present. If these conditions are met, the extraction is done by simple rules.

For more general text, Agichtein and Gravano (2000) describe strategies for generating patterns to extract pairs of entities in a given relationship. For example, the goal is to extract pairs such as (*Microsoft, Redmond*), meaning that *Redmond* is the location of the organization *Microsoft*. Given an initial set of example pairs, the system analyses the context (pattern) in which these pairs occur, and from this set of initial patterns, the system finds new pairs, from which new context patterns are generated and so on (see Section 3.2.5.4 for a discussion of the bootstrapping process). In this paper it is assumed that the two entities LOCATION and ORGANIZATION

[14] This system, therefore, would not solve the important problem of finding all proteins related to one given protein.

are always in the same relation, which can be true for this specific case, but is not generally applicable. Similarly to the work in Craven and Kumlien (1999), as long as they identify *one* example of a pair, they consider the system to be correct for that pair. This is different from the goal of IE, in which *all* instances have to be retrieved; for this reason, they introduce a metric that is different from the standard metric in IE (see Agichtein and Gravano, 2000, for details). The system relies on pairs appearing multiple times in the document collection. Different results are reported depending on the number of occurrences of such pairs: 85% precision and 80% recall for 1 occurrence of the pairs and 90% precision and 85% recall and for 10 occurrence of the pairs.

3.2.3.2 Co-occurrence for relation classification

Stapley and Benoit (2000) hypothesize a functional relationship between genes that occur together in the same document with statistically significant frequency. A graph is generated, with edges between pairs of genes. The lengths of the edges are a function of the co-occurrence of the couples in the literature. Stapley and Benoit (2000) claim that the type of the relationships are implicitly represented graphically by the clustering of genes with related functions.

In Stephens et al. (2001), a relationship between a pair of genes (taken from a fixed list) is predicted if there is a strong association between them, measured as co-occurrence; in other words, if two genes co-occur frequently in a collection, then a relationship between them is predicted. This method would not work for rare events that may be the most interesting cases (at least for the task of knowledge discovery). If a relationship is predicted, it is classified by looking up a list of predefined relationship keywords (*activates, binds, transports*).

3.2.3.3 Machine learning systems for relation classification

Zelenko et al. (2002) introduce kernel methods for the task of relation classification. The input is a shallow parse of the sentence with noun phrases and names tagged with the relevant entity tags. Kernels are defined over the shallow parses and Support Vector Machine and Voted Perceptron algorithms are used for the classification over the kernels. The task is to extract the relationships *person-affiliation* and *organization-location*. They compare their results with those of some feature methods (such as Naive Bayes) and report superior performance of the kernel methods (F-measure of 86.8% for *person-affiliation* and 83.3% for *organization-location* versus respectively 82.93% and 80.4% for the Naive Bayes).

Culotta and Sorensen (2004) create dependency trees for each entity pair (representations that denotes grammatical relations between words in a sentence); a relation instance is a dependency tree. Kernel functions are defined over these trees and support vector machines used for the classification. The experiments were done on the ACE data, with five relation types; they report an F-measure of 45%.

For the bio-medical domain, Craven (1999) tackles the problem of relationship extraction from MEDLINE (for the relation *subcellular-location*) also as a problem of text classification and uses a Naive Bayes for the classification. They assume that semantic labels are given to the words[15] and a relation is assigned to a sentence if the sentence contains words belonging to the semantic classes of interest and if a classifier classifies the sentence as a positive example. They propose and compare two classifiers: a Naive Bayes classifier with a bag-of-words representation and a relational learning algorithm that learns relational rules in terms of a (shallow) parsed representation of the text. The results reported are: a precision of 78% and a recall

[15]It is not described how these semantic labels, PROTEIN and SUBCELLULAR-STRUCTURE, were assigned.

of 32%[16] for an F-measure of 45.3% for the Naive Bayes and a a precision of 92% and a recall of 21% (34.1% F-measure) for the relational learning.[17]

They do not mention if these semantic classes can be related by another functional relationship. They classify sentences into two groups: those that have these entities in this relation and all the other. It would have been interesting to see a three-(at least)-way classification: (i) sentences with PROTEIN and SUBCELLULAR-STRUCTURE in the *subcellular-localization* relation, (ii) sentences with PROTEIN and SUBCELLULAR-STRUCTURE in another relation (iii) all other sentences. It would have been interesting to see how (i) differs from (ii). They do have a baseline system that predicts that this relation holds if a PROTEIN and a SUBCELLULAR-STRUCTURE occur in the same sentence. This system has an F-measure of 51% approximately,[18] meaning that there are indeed instances of other relations between these two entities but unfortunately examples of such cases are not reported.

HMMs also have been proposed for this task: Palakal et al. (2002) distinguishes between *directional relationships* (for which it is important to know the direction of the action, as in "protein A inhibits protein B") and *hierarchical relationships* for which the direction is not needed, as for "the brain is part of the nervous system"[19] and classify them with two different methods.[20] Directional relationships are classified

[16]These numbers are taken (by me) from a graph but never appear in the text and therefore are approximate.

[17]Note also that they have a somewhat "forgiving" evaluation policy: they say that "although each instance of the target relation may be represented multiple times in the corpus, we consider the IE-method to be correct as long as it extracts this instance from *one* of its occurrences." This could be fine for certain applications, however, I believe that for many applications it may be important to return *all* the articles that talk about a certain relation.

[18]The F-measure of 51% is at 71% recall, for which value the results of the Naive Bayes are not reported, but at the same level of recall of about 32%, the Naive Bayes achieves 78% precision, while the sentence co-occurrence baseline achieves 40% precision.

[19]It is not clear to me why it is so, in fact, we could also find: "the brain belongs to the nervous system" in which the direction is not symmetric; we could assume that we have a hierarchical ontology from which we can extract the information of the "hierarchical directionality" but this is not mentioned in the paper.

[20]There is no discussion on how the system is supposed to distinguish between the two kinds of

using an HMM model (that finds the state sequence, therefore distinguishing between the "agent" and the "patient" entities). The model was trained using four directional relationships: *inhibit, activate, binds* and *same* but it is not clear how (and if) it also classifies the different relationships or if it only finds the state sequence. Palakal et al. (2002) claim that for *hierarchical relationships* the verb is not important and that therefore this type of relationship can be defined using co-occurrence; the method used is the same as Stephens et al. (2001).

Ray and Craven (2001) apply an HMM to extract the entities PROTEINS and LOCATIONS in the relationship *subcellular location* and the entities GENE and DISORDER in the relationship *disorder-association*. Ray and Craven (2001) acknowledge that the task of extracting relations is different from the task of extracting entities (because entities can be in different relationships with each other) but then they consider (MEDLINE) sentences that contain words that match target tuples collected from two databases[21] to be positive sentences. This essentially means that the positive sentences are all the sentences that contained the entities *no matter in which relationship the entities were in*. The actual task is therefore the one of *entity* extraction. They do acknowledge the problem with this approach and they report that approximately 10% to 15% of the sentences are labeled incorrectly but unfortunately there is no discussion of how the sentences with entities in the relationships of interest differ from the sentences in which the entities are in another relationship and what other relations occur between these entities (but see Section 3.2.2 for a description of their interesting model).

relationships.

[21]Yeast protein Database (YPD) for *subcellular location* and the OMIM database for *disorder-association*.

3.2.4 Syntactic information

Another dimension along which we can analyze IE systems is the amount of syntactic information they make use of.

Systems that tackle structured (or semi-structured) text do not usually use any syntactic information. HTML and SGML tags can help the identification of the constituents; some systems also use format information (position on the web page, font, colors, new-line, etc.). These features are easier to include in a statistical system, are usually more accurate than syntactic information and the space they define is much smaller. Blei et al. (2002) use formatting, layout, directory structures, and linkage information to extract job descriptions from Web pages. The intuition is that: "if one wished to extract the titles of all the books on Amazon.com, one can rely on the fact that the book title appears in the same location on the books home page and in the same font." McCallum et al. (2000) use only line-based features for the talk of extracting certain fields from Usenet FAQs.

One may think that syntactic information for systems that tackle free text should be necessary, however, many IE systems do not use any (see, for instance Freitag and McCallum, 2000; Bikel et al., 1999; Borthwick et al., 1998). In these cases, the features can be the words themselves, word-based features, like for example, "word contains a digit," "word is a two-digit number" like in Bikel et al. (1999), and semantic features (like dictionary features).

Klein et al. (2003) propose a character level HMM (the observations are the characters) and a character level maximum entropy markov model for the task of named entity recognition; the reason to do so is to address the problem of data sparsity. Interestingly, they report a significant error reduction switching from word-based models to the character based ones.

For systems that do use syntactic information, on one end of the spectrum we

find systems that use only part-of-speech tags (Collins and Miller, 1997), on the other, systems that use complete parse trees (Gildea and Jurafsky, 2000; Miller et al., 2000; Chelba and Mahajan, 2001). In between, there are the systems that use only "shallow" or "flat" syntactic information.

AutoSlog-TS (Riloff, 1996), for example, uses a syntactic parser to produce a shallow parse tree that segments sentences into noun phrases, prepositional phrases and verb phrases and finds some basic grammatical functions (subject, object) and the voice of the verbs.[22] Ray and Craven (2001) incorporate phrase-constituent information in a HMM, representing a sentence as a sequence of phrases (see the discussion of this paper in Section 3.2.2.2). It is difficult to understand the impact of the syntactic representation, since these systems have very different models, different features, different and training procedures.

It would be interesting to understand what "level" of syntactic information should be used and under what circumstances. The following papers address this problem.

Ray and Craven (2001) compare a "phrase" model with two "token" models, one that includes the part-of-speech of the words and one with only words and no syntactic information (keeping fixed the statistical model, an HMM, see Section 3.2.2.2); they report better results for the "phrase" model suggesting that there is value in representing the grammatical function for the task of IE.

Gildea and Palmer (2002) compare a "flat," "phrase" model with a full parser model; it examines how the information provided by modern statistical parsers, such as Collins (1997) and Charniak (1995), contribute to solving IE. They measure the effect of the parser accuracy and determine whether a complete parse tree is necessary for accurate role prediction in IE for free text from the Propbank corpus. They use the system described in Gildea and Jurafsky (2000) that passes sentences through

[22]To distinguish, for example, the patterns "VICTIM passive-verb by OFFENDER" versus "OFFENDER active-verb VICTIM."

an automatic parser, extracts syntactic features from the parser and estimates the probabilities for the semantic roles from the syntactic and lexical features (in other words, it calculates the probability of a semantic role given the features and chooses the role that maximizes this probability). The features used are the following: Phrase Type (NP, VP, S), Parse Tree Path (the path from the predicate through the parse tree to the constituent in question; this feature is designed to capture the syntactic relation of a constituent to the predicate), Position (whether the constituent to be labeled occurs before or after the predicate; this feature is highly correlated with grammatical function), Voice and Head Word. They test the hypothesis that features based on a full parser are useful for IE by comparing their tree-based system with a system which is given only a flat, "chunked" representation of the input sentence. They report better results with the complete parse based system: 57.7% precision and 50% recall of the full parse system (53.6% F-measure) versus 49.5% precision and 35.1% recall (41% of F-measure) for the chunked representation. They conclude that this shows that the constituent structure provided by the statistical parser provides relevant information for IE. They also show that head word information (a side product of the parser) improves the results. To the best of my knowledge, this is the first paper that tackles this problem explicitly, and it does seem to provide some evidence for its claim. However, the comparison does not seem to be completely fair. Much work was done to develop the full parse based system (and 2 papers written about it), while the IE system using the flat representation is very simple and (apparently) developed only for a comparison for this paper. In other words, it is not entirely clear to me whether the flat representation is to be claimed for the worse performance or the IE system built in top of it.

Xue and Palmer (2004) show that the syntactic information has yet to be fully exploited and that different features are needed for different subtasks (argument identification and argument classification); they propose a set of syntactic features and

show how these features lead to a significant improvement; the task was the semantic annotation of Proposition Bank (Kingsbury and Palmer, 2002).

3.2.5 Using unlabeled data

One major problem of natural language processing is the sparsity of data; linguistic patterns occur in a very skewed distribution, with a small number of events occurring very frequently and a long "tail" of events occurring very rarely (the Zipf distribution). Therefore, to accurately learn a linguistic model we need many patterns to cover the tail and this means that we need to label a large amount of text, which is usually an expensive requirement.

For information extraction, the labeling process is particularly difficult and time consuming. Moreover, we need different labeled data for each domain. (In Chapter 4, I address this problem and propose a method to gather data for the task of protein interaction classification).

In the following sections, I describe the different methods that have been proposed in this direction.

3.2.5.1 Unsupervised methods

Unsupervised methods do not use labeled data and try to learn a task from the "properties" of the data. They can be viewed as clustering methods.

In Hasegawa et al. (2004) relations between entities are discovered through a clustering process; pairs of entities occurring in similar context can be clustered and each pair in a cluster is an instance of the same relation. They also label the clusters, with most frequent common words in the cluster being its label. They use an entire year of the NY Times and report an F-measure of 80% for the 12 relations between the entities PERSON and GPE and an F-measure of 75% for the 6 relations between

COMPANY and COMPANY.

Freitag (2004) advocates the combination of supervised learning on a small training set with features derived from a much larger unlabeled set via a clustering algorithm. F-measures on the extraction of MUC roles show that the use of such features improves the performance on most roles, typically benefiting the recall.

3.2.5.2 Weakly and distantly labeled data

Seymore et al. (1999) apply the term *distantly labeled* data to data that was labeled for another purpose but which can be applied to the problem at hand. Their goal is to extract fields such as TITLE, ABSTRACT, AUTHOR, EMAIL, INTRODUCTION from headers of computer science papers. They take advantage of the fact that several of the labels that occur in citations (such as TITLE and AUTHOR), also occur in the headers of papers and they use BIB-TEX files to obtained such labeled data (which constitute the *distantly* labeled data). They show how this data is useful: its addition provides a 10.7% improvement in extraction accuracy for headers.

In Craven (1999), *weakly labeled data* are facts that can be easily extracted and that *may* offer some evidence with which we can "label" the (otherwise) unlabeled text. For example, the goal in Craven (1999) is to extract the relationship *subcellular-location*;[23] they observe that the Yeast Protein Database (YPD) includes a field *subcellular-location* for many proteins and also the references to the MEDLINE articles containing that relation for those proteins. They assume then that a MEDLINE abstract that is referenced by instances of this field in YPD contains sentences that do indeed contain the relation.[24] In other words, instead of the hand labeling they use the reference from YPD. They report a 69% precision at 30% recall (41% F-measure)

[23]Or, more precisely, to classify sentences into sentences that contain the relationship and sentences that do not.

[24]They actually assume that the sentence in that abstract that contain both a PROTEIN and a LOCATION is the "positive" sentence.

for a Naive Bayes classifier trained on hand-labeled data and a 77% precision at 30% recall (43% F-measure) for the same Naive Bayes classifier trained on weakly labeled data.

3.2.5.3 EM (Semi-supervised learning)

EM (Expectation-Maximization) is one standard approach for learning with missing values. EM is essentially Maximum Likelihood for unlabeled data, which is usually a good choice if the model chosen is the "right" model for the data; if, on the other hand, the model is not the right one, maximizing the likelihood of the data under that model may not be the right thing to do. This is what seems to happen in Seymore et al. (1999) where they use an HMM and EM with unlabeled data. They set the initial parameters to the ML estimates obtained from labeled data and then they run the Baum-Welch (Baum, 1972) algorithm on the unlabeled data.[25] Their results show how adding unlabeled data to labeled and distantly labeled data does not improve the results (in terms of classification accuracy) with respect to the results with only labeled and distantly labeled data. They say that this can be due to the fact that the Baum-Welch algorithm maximizes the likelihood of the unlabeled data, not the accuracy of the classification and that this is actually shown by the improvement in test perplexity (which is a measure of how well the models fits the data).

3.2.5.4 Bootstrapping

Bootstrapping is an iterative process where, given (usually) a small amount of labeled data (seed-data), the labels for the unlabeled data are estimated at each round of the process, and the (accepted) labels then incorporated as training data. Jones et al. (1999) describe one such process. Their IE system relies on two dictionaries: a

[25] Baum-Welch training essentially produces new parameter estimations that maximize the likelihood of the unlabeled data.

dictionary of extraction patterns and a semantic lexicon; in Jones et al. (1999) a bootstrapping technique generates both dictionaries simultaneously. Their approach is based on the following observations:

1. objects that belong to a semantic class can be used to identify extraction patterns for that class (for example, suppose we know that "terrier" is a DOG, then given the sentence "the terrier barked" we can extract the pattern "<X> barked" as representative of the class DOG).

2. (Conversely) extraction patterns that are known to be of a semantic class can be used to identify new members of that class (for example, knowing that the pattern "<X> barked" is associated with the class DOG, when we find "the dalmation barked" we can assume that "dalmation" is DOG).

Their bootstrapping algorithm starts with a small number of seed words that belong to a semantic class of interest; these seed words are used to learn the extraction patterns that then can be used to extract new members of the same semantic class. This process is iterated several times and at each times a score is given to each extraction pattern and to each new lexicon entry and only the most reliable ones are retained. They report a precision of 76% for a Web location dictionary and a precision of 63% for a terrorist location dictionary.

Riloff (1996) is based on the same idea, but the patterns are not only a sequence of words (like in Jones et al., 1999, 'shot in <X>," "to occupy <X>") but contain some syntactic information: "<subject> kidnapped" and "exploded on <np>" are two examples. Agichtein and Gravano (2000) and Yangarber et al. (2000) are again very similar; Agichtein and Gravano (2000) have a different technique for generating patterns (and a slight different definition for "pattern") and a different evaluation measure and Yangarber et al. (2000) include semantic classes in the definition of patterns and in the pattern matching procedure.

Collins and Singer (1999) propose an unsupervised algorithm based on decision lists and a boosting-based algorithm along with unlabeled data and only seven seed rules to tackle named entity classification.

Swier and Stevenson (2004) use a verb lexicon (VerbNet), a bootstrapping algorithm and a back-off model to label the arguments of verbs with their semantic roles (there are 13 roles such as AGENT, AMOUNT, BENEFICIARY, CAUSE, EXPERIENCER). They use the lexicon to assign the initialize the model probabilities.

3.2.5.5 Co-training

As defined in Blum and Mitchell (1998), co-training is a type of bootstrapping for problems in which "the description of each example can be partitioned into two distinct views" and for which both (a small amount of) labeled data and (much more) unlabeled data are available. For their application of classifying Web pages, one view is the bag-of-words in the Web page and the other in the words that occurs in the hyperlinks pointing to that page. They train two classifiers (Naive Bayes) on the labeled data, each classifier using as features only one "view."[26] They assume that the classification of the 2 classifiers is consistent, that is, that each view itself is sufficient for a correct classification. After training the two classifiers, they use them to label unlabeled data. They train again the classifiers with these "self-labeled" examples and show how the error rate decreases from the error rate obtained using only the original labeled data, concluding that this is evidence that co-training successfully use the unlabeled data to out-perform standard supervised training.

In my view, co-training is essentially the *one-iteration, probabilistic* version of bootstrapping as defined in Section 3.2.5.4.

[26]For example, one classifier classifies the Web page given its bag-of-words and the other given its hyperlink-words.

3.3 Data and annotation

In this section I describe the data used for my experiments for entity and relationship classification.

The text was obtained from MEDLINE 2001. I took the first 100 titles and the first 40 abstracts from the 59 files medline01n*.xml in Medline 2001. The intention was to retrieve a broad variety of concepts. No keywords of any sort were used to retrieve the documents.

The annotator, a SIMS masters student with a biology background, looked at the titles and abstracts separately and did the labeling through the text sentence by sentence. I decided to concentrate on the semantic roles TREAT and DIS and I asked the annotator to see how many different types of relationships could be found between these two roles. She came up with 8 types of relationships (see Section 3.3.1) and labeled the text accordingly. She writes: "I labeled sentences based solely on the content of that individual sentence and not other sentences in the same abstract. Sometimes reading the abstract helped me figure out what was going on in general especially when the disease or treatment names were obscure or weird or abbreviated. But overall I tried to ensure that a labeled relation within a sentence was not dependent on other sentences around it and could stand on its own."

Riloff (1996) notes how complex the annotation task is, in that it is not always clear, for example, what constitutes a relevant noun phrase. Should we include all the modifiers, only the most relevant ones or just the head noun? Determiners? Should we include prepositional phrases?

I did not specify an exact labeling convention for the noun phrase boundaries, and this resulted in some inconsistency in the data. For example, for ``ovarian cancer'' only cancer was labeled to be a DIS but in another sentence with ``breast cancer'' both words were labeled as DIS; in: ``<DIS> non-recurrent cancer </DIS> of

the cervix''[27] only non-recurrent cancer was labeled as a DIS but in ``<DIS> complicated cancer of the large bowel</DIS>'' the whole phrase was considered to be a DIS. The reason for this could be due to the different importance and emphasis of the concepts in the sentences; it could also be the case that these are just labeling inconsistencies.

In some systems the annotation depends on the syntactic information, as in Gildea and Palmer (2002).[28] In our case, the annotation was done independently of any syntactic information and with no constraints whatsoever and this also gives rise to some inconsistency in the labeling; for example, in ``The <DIS> lesion </DIS> was resected by ...'' only *part* of the noun phrase ``The lesion'' was labeled as a DIS and the determiner left out, while in ``...<TREAT> the paravertebral block </TREAT> ...'' the whole NP was labeled.

I retained the sentences that were found *not* to contain the entities and relationships of interest and in my experiments I distinguish between relevant and non-relevant sentences. The non-relevant sentences come from the same population of abstracts and titles than the relevant ones, and therefore relevant and non-relevant sentences can be very similar to one another, in terms of discussing the many of the same concepts.

A total of 3570 sentences were labeled. Table 3.3 shows the number of sentences found for each type of relation. These labeled sentences are available at:
http://biotext.berkeley.edu/data.html

[27] <label> means that the word that follows it is the first of the entity and </label> that the word that proceeds it is the last of the entity.

[28] Gildea and Palmer (2002) use Propbank and note that "Propbank annotation takes place with reference to the Penn Treebank trees - not only are the annotators shown the trees when analyzing a sentence, they are constrained to assign the semantic labels to portions of the sentence corresponding to nodes in the tree."

Relationship	Definition	Num.
Cure	TREAT cures DIS	810
Only DIS	TREAT not mentioned	616
Only TREAT	DIS not mentioned	166
Prevent	TREAT prevents the DIS	63
Vague	Very unclear relationship	36
Side Effect	DIS is a result of a TREAT	29
NO Cure	TREAT does not cure DIS	4
Complex	More than one relation	75
Total relevant		1799
Non-relevant	TREAT and DIS not present	1771
Total		3570

Table 3.1: Candidate semantic relationships between treatments and diseases, a short definition and the total number of sentences found for each relation.

3.3.1 Semantic relationships

In this section, I describe the various types of semantic relations that were found to occur between the semantic classes of TREAT and DIS and I report a few examples for each relationship.

3.3.1.1 Cure

For the cure relation, according to the annotator: "To label a sentence as *cure* the treatment has to cure the disease or it is *meant* to cure it but might still be in testing (e.g. clinical trials). On more thought I wonder if these two relationships should actually be separated into two relationships. This might be useful due to the obvious difference between a treatment that has been shown to be effective clinically versus a treatment that is still being tested or was inconclusive. I decided for the moment to have only one relation for these two concepts."
Some examples for this relation are:

OBJECTIVES: <DIS> Obesity </DIS> is an important clinical problem, and the use of <TREAT> dexfenfluramine hydrochloride </TREAT> for weight reduction has been widely publicized since its approval by the Food and Drug Administration.

<TREAT> Antibiotics </TREAT> prescribed for <DIS> sore throat </DIS> during the previous year had an additional effect (hazard ratio 1.69, 1.20 to 2.37).

<TREAT> Intravenous immune globulin </TREAT> for <DIS> recurrent spontaneous abortion </DIS>.

3.3.1.2 Only Disease

The Only Disease relation is assigned when a treatment is not mentioned in the sentence (other entities may have been present). Some examples:

The objective of this study was to determine if the rate of <DISONLY> preeclampsia </DISONLY> is increased in triplet as compared to twin gestations.

<DISONLY> Down syndrome </DISONLY> (12 cases) and <DISONLY> Edward syndrome </DISONLY> (11 cases) were the most common <DISONLY> trisomies </DISONLY>, while 4 cases of <DISONLY> Patau syndrome </DISONLY> were also diagnosed.

<DISONLY> Chronic pancreatitis </DISONLY> and <DISONLY> carcinoma of the pancreas </DISONLY>

3.3.1.3 Only Treatment

The Only Treatment relation is assigned when a disease was not mentioned in the sentence (other entities may be present). Some examples:

> Patients were randomly assigned either <TREATONLY> roxithromycin </TREATONLY> 150 mg orally twice a day (n = 102) or placebo orally twice a day (n = 100).

> <TREATONLY> Heterologous vaccines: </TREATONLY> proponent sparks some interest.

> Meta-analysis of trials comparing <TREATONLY> antidepressants </TREATONLY> with active placebos.

3.3.1.4 Prevent

The Prevent relation is assigned when there is a clear implication that a TREAT will prevent a DIS. This might be inherent in the definition of the treatment, e.g., a vaccine works by preventing a disease from occurring, or explicitly stated, often with the words "prevent" or "prevention of." Also seen is the phrase "reduce incidents," "reduce rates of," or "reduction in rates..." because these also imply that disease events are being prevented. Examples:

> I investigated the hypothesis that <TREAT PREV> an antichlamydial macrolide antibiotic, roxithromycin </TREAT PREV>, can prevent or reduce recurrent major ischaemic events in patients with <DIS PREV> unstable angina </DIS PREV>.

Immunogenicity of <DIS PREV> hepatitis B </DIS PREV>
<TREAT PREV> vaccine </TREAT PREV> in term and preterm infants.

<TREAT PREV> Modified bra </TREAT PREV> in the prevention of
<DIS PREV> mastitis </DIS PREV> in nursing mothers

3.3.1.5 Side Effect

The Side Effect relation is assigned when a DIS is a result of a TREAT. The cause/effect relationship should be explicitly stated or at least very clearly implied or hypothesized. Usually in "side effect" sentences there is a time-line element because the DIS occurs after some TREAT. Examples:

Initially, all eyes that had <TREAT SIDE EFF> optic capture
</TREAT SIDE EFF> without <TREAT SIDE EFF> vitrectomy </TREAT
SIDE EFF> also remained clear, but after 6 months, four of five
developed <DIS SIDE EFF> opacification </DIS SIDE EFF>

Appetite suppressants-most commonly <TREAT SIDE EFF>
fenfluramines </TREAT SIDE EFF> -increase the risk of developing
<DIS SIDE EFF> PPH </DIS SIDE EFF> (odds ratio, 6.3),
particularly when used for more than 3 months (odds ratio, > 20)

The most common toxicity is <DIS SIDE EFF> bone pain </DIS SIDE
EFF>, and other reactions such as <DIS SIDE EFF> inflammation
</DIS SIDE EFF> at the site of <TREAT SIDE EFF> injection
</TREAT SIDE EFF> have also occurred.

3.3.1.6 Vague

The Vague relation is assigned when a relationship of some sort between a TREAT and a DIS is implied but not better specified. It can be either a TREAT that affects a DIS or something associated with the condition of a DIS or, not as often, a DIS that has some sort of effect on a TREAT. Often these sentences contain phrases such as "effect on," "effect of," "association between," "changes in," "following," "impact of." This differs from the regular "cure" because it is not readily apparent that the TREAT is actually meant to be a direct treatment for the DIS in the sentence. Often the effect of the TREAT is specifically on some element associated with the DIS itself like the first example below: the effect of the TREAT, *lorazepam*, is specifically on respiratory muscles in people with the DIS, *chronic obstructive pulmonary disease*, but the actual kind of effect is not known, and can be either good or bad.

Examples:

```
Acute effect of <TREAT VAG> lorazepam </TREAT VAG> on
respiratory muscles in patients with <DIS VAG> chronic
obstructive pulmonary disease </DIS VAG>
```

```
Comparison of the effects of <TREAT VAG> salmeterol </TREAT VAG>
and <TREAT VAG> ipratropium bromide </TREAT VAG> on exercise
performance and breathlessness in patients with <DIS VAG> stable
chronic obstructive pulmonary disease </DIS VAG>
```

```
Impact of postmenopausal <TREAT VAG> hormone therapy </TREAT
VAG> on cardiovascular events and <DIS VAG> cancer </DIS VAG>.
Testing for <DIS VAG> Helicobacter pylori infection </DIS VAG>
```

after <TREAT VAG> antibiotic treatment </TREAT VAG>

<DIS VAG> Hyponatremia </DIS VAG> with <TREAT VAG> venlafaxine </TREAT VAG>

<TREAT VAG> Hormone replacement therapy </TREAT VAG> and <DIS VAG> breast cancer </DIS VAG>

3.3.1.7 Do NOT Cure

The relation Do Not Cure is assigned when a TREAT that is meant to cure a DIS does not work. Unfortunately (and, in my view, surprisingly), I found only 4 instances for this relationship:

More of those initially prescribed <TREAT NO> antibiotics </TREAT NO> initially returned to the surgery with <DIS NO> sore throat </DIS NO>.

To avoid medicalising a self limiting illness doctors should avoid <TREAT NO> antibiotics </TREAT NO> or offer a delayed prescription for most patients with <DIS NO> sore throat </DIS NO>.

<TREAT NO> Subcutaneous injection of irradiated LLC-IL2 </TREAT NO> did not affect the growth of preexisting <DIS NO> s.c. tumors </DIS NO> and also did not improve survival of mice bearing the <DIS NO> lung or peritoneal tumors </DIS NO>

Evidence for double resistance to <TREAT NO> permethrin and malathion </TREAT NO> in <DIS NO> head lice </DIS NO>

3.3.1.8 Complex

I did not include in our experiments these more complex sentences that incorporate more than one relationship, often with multiple entities or the same entities taking part in several interconnected relationships. For example, in the first sentence, there is a "cure" relationship between *oral fludarabine* and the DIS *chronic lymphocytic leukemia* but also a "side effect" of the TREAT, *progressive multifocal leukoencephalopathy*. In the second one, there is one treatment that cures and one that does not. I found 75 of such sentences.

<DIS SIDE EFF> Progressive multifocal leukoencephalopathy </DIS SIDE EFF> following <TREAT> oral fludarabine </TREAT> treatment of <DIS> chronic lymphocytic leukemia </DIS>.

<TREAT> Intraperitoneal injection of irradiated LLC-IL2 </TREAT> cured <DIS> pre-existing LLC peritoneal tumors </DIS> and extended the survival of the mice but did not affect survival of mice bearing <DIS NO> lung tumors </DIS NO> nor did it affect the growth of <DIS NO> s.c. tumors </DIS NO>.

3.4 Preprocessing

3.4.1 Analyzing at the Sentence Level

Analysis was done at the sentence level given the empirical results in the bioscience text analysis literature that suggest this is a proper unit of analysis. For example,

Ding et al. (2000) compare abstracts, adjacent sentence pairs, sentences and phrases as processing units for the task of mining interactions among biomedical terms and show statistically significant differences between them. In particular, they report an F-measure of 0.729 for the sentences, 0.727 for the abstracts, 0.677 for the phrases and 0.50 for the sentence pairs.

Cooper and Kershenbaum (2005) manually analyze 65 abstracts about protein-protein interactions and note how "In all but one case, the interactions were described in the same sentence, and thus resolving co-reference issues would add only marginally to the quality of the interaction detections. Thus the fact that two proteins occurred in the same abstract, but not in the same sentence was not a good metric for the number of relations we should be able to find."

Finally, for the task described in Chapter 4, I compared the results of the classification using only the sentences containing the entities of interest (proteins) with the results obtained using a larger window: the sentence with the entities along with the previous and following ones or the three consecutive sentences that contained the proteins (the proteins could appear in any of the sentences). The results obtained by using these larger chunks were consistently worse. This evidence supports the choice of the sentence as an unit of text from which to extract facts.

3.4.2 Preprocessing Steps

Given the labeled text, I passed it through a series of processing steps.

- **Sentence splitter**: to divide the abstracts and titles into sentences.[29]

[29] I wrote a program that splits the text at the periods, unless the periods are part of certain words (this list was mainly found empirically and contains words such as "e.g.," "i.e.," "u.s.," "jan." etc.); I also included a list of honorifics ("mr.," "mrs.") taken from another sentence splitter (http://L2R.cs.uiuc.edu/ cogcomp/cc-software.html).

- **Tokenizer**: I used the tokenizer used (and provided) by the The Penn Treebank Project.[30]

- **Brill's POS tagger** (Brill, 1995).

- **Collins parser**: for the moment, I use the parser only to get, from its output, a shallower representation (see next step) (Collins, 1996).

- **Shallow parser**: given the output of the Collins parser, I wrote a program that does a simple chunking. For example, given the following sentence

  ```
  Underutilization of aspirin in older patients with prior
  myocardial infraction
  ```

  the parse tree found by Collins parser is shown in Figure 3.2 and the output of my chunker for this sentence is the following

  ```
  ( NP [NNS] Underutilization )
  ( PP [IN] of )
  ( NP [NN] aspirin )
  ( PP [IN] in )
  ( NP [JJR] older [NNS] patients )
  ( PP [IN] with )
  ( NP [JJ] prior [JJ] myocardial [NN] infarction )
  ```

- **Semantic tagging with MeSH** As I did for the noun compounds (see Sections 2.6.1 and 2.7.3) I mapped the words into MeSH terms. The mapping of the previous sentence is the following (keeping the chunked representation):

[30] http://www.cis.upenn.edu/ treebank/tokenization.html

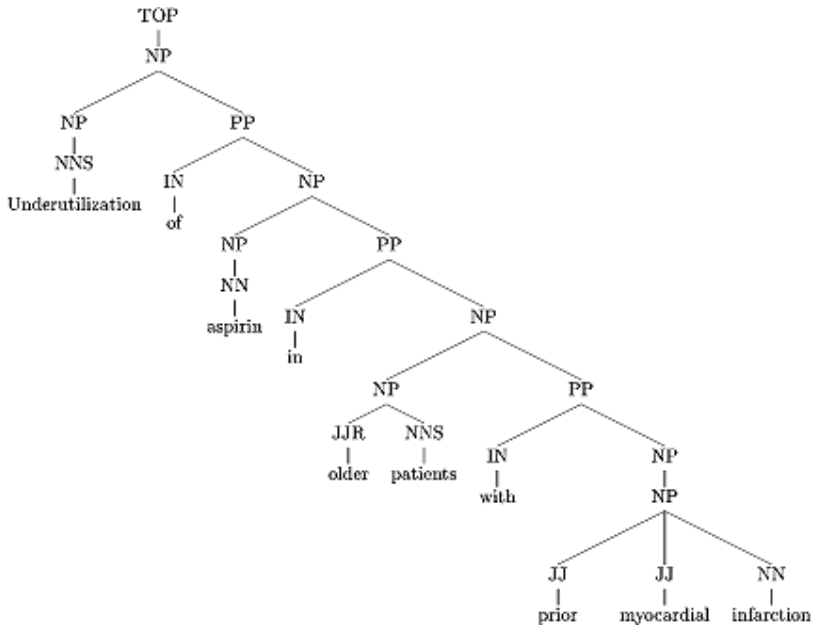


Figure 3.2: Parse tree found by Collins parser for the sentence: *Underutilization of aspirin in older patients with prior myocardial infraction*

```
( NP [NNS] Underutilization )
( PP [IN] of )
( NP [NN] aspirin D02.241.223.100.380.800.075,
D02.241.511.390.700.075, D02.755.410.700.075 )
( PP [IN] in )
( NP [JJR] older [NNS] patients M01.643 )
( PP [IN] with )
( NP [JJ] prior [JJ] myocardial [NN] infarction
C23.550.717.489)
```

(The multiple mapping can be due to lexical ambiguity or just to different ways of classifying the same concept. Ambiguity, however, is relatively rare

for this ontology. For this work, I simply retain the first MeSH mapping for each word, so that, for example, the mapping of `aspirin` becomes just `D02.241.223.100.380.800.075`, but further processing for word sense disambiguation is probably needed.)

Different levels of description of the hierarchy may be needed for different words, or for different parts of the hierarchy (see discussion in Section 2.7) or for different tasks. For the moment I represent all individual MeSH terms up to the second level, so that `aspirin` is mapped to `D02.241`.

3.5 Features

For each word in the sentence I extract the following features:

- **Semantic Role** (ROLE) given by the labeling. The possible values are DIS (disease), TREAT (treatment) and NONE. In Section 3.3.1, I showed how I labeled the diseases and treatments in sentences with different semantic relations with different labels. For example, `hepatitis B` was a <DIS PREV> in

  `Immunogenicity of <DIS PREV> hepatitis B </DIS PREV> <TREAT PREV> vaccine </TREAT PREV> in term and preterm infants.`

  but a <DIS EFFECT> in

  `Effect of <TREAT EFFECT> interferon </TREAT EFFECT> on <DIS EFFECT> hepatitis B < /DIS EFFECT>`

  Keeping these different labels would require a different model for each label or for each pair of labels (see Section 3.7). For now I assume that *all types* of diseases are DIS and *all types* of treatments are TREAT. In the named entity recognition task, I extract *all* TREAT and DIS from the sentences without distinguishing

between them. In the relationship recognition task, I distinguish between the different relations, essentially assigning, for example, the label <DIS PREV> to a <DIS> that occur in a *prevention* relationship.

- **Word** (w). The words themselves. I substitute the words that occur less than 3 times with the "unknown" token. I do not do any stemming.
- **Part of speech** (pos) from Brill's tagger.
- **Phrase Constituent** (constChunk). The phrase type from the shallow parse. The purpose of this feature is to include some "higher level" syntactic information into the word-based model. For example constChunk of `underutilization` is "np" and constChunk of `with` is "pp."
- **Belongs to the same chunk as previous word** (differentChunks). For example for `older` differentChunks=NO, but for `patients` differentChunks=YES, because the word that precedes `patients` (`older`) belongs to the same chunk.
- **MeSH** (mesh). The MeSH mapping of the words
- **Domain Knowledge** (lab). We know that terms that are mapped to the C sub-tree in MeSH are usually diseases and that only some of the terms in the E tree and only some of the G terms in MeSH are treatments; I have identified all the sub-hierarchies of MeSH that correspond to treatments and the sub-hierarchy that corresponds to diseases. This allows me to include some domain knowledge. For example, for `infarction` (C23.550), lab=dis; lab can have 3 values: disease, treatment and null.
- **Morphological Features**
  - **Is number**

- **Only part is number**
- **Is negation**
- **First letter is capital**
- **All capital letters**
- **All non word character**
- **Contains non word character**

I ran the experiments using different combinations of these features.

3.6 Evaluation

For the evaluation of the role extraction task, I use the evaluation scoring from the MUC Manual.[31] Evaluation is done at the word level. The semantic label assigned by the system is compared to the "true" roles.

In the MUC scoring framework, the evaluation metrics (which they call tallies) are:

- COR: Correct. The truth and the (system) prediction agree
- INC: Incorrect. Truth and prediction disagree
- MIS: Missing. There was a truth value but no prediction
- SPU: Spurious. There was a prediction but not a truth value (i.e., the truth value was "null")
- POS: Possible. The total number of truth values.
  POS = COR + INC + MIS

[31] http://www.itl.nist.gov/iaui/894.02/related_projects/muc/muc_sw/muc_sw_manual.html

- ACT: Actual. The total number of predictions.
  ACT = COR + INC + SPU

Given this set of tallies, there are several values calculated in the alignment and final scoring:

- REC: Recall. A measure of how many of the truth values were produced in the response:
  REC = COR / POS

- PRE: Precision. A measure of how many of the predictions are actually in the truth:
  PRE = COR / ACT

- F-measure: $F_\beta = ((\beta^2 + 1) * PRE * REC)/(\beta^2 * PRE + REC)$
  If $\beta = 1$ PRE and REC are given equal weight. In my experiments, $\beta = 1$ and the F-measure formula reduces to
  $F_1 = (2 * PRE * REC)/(PRE + REC)$

The goal is to achieve a high F-measure.

Below is a typical output of the alignment. The first column shows the original token from teh sentence, the second column shows the system's prediction, the third is the "true" label, and the last one is the tally, or score, assigned by the evaluation. A blank means that there was no prediction and/or no true value.

```
underutilization || || ||
of || || ||
aspirin || TREAT || TREAT || COR
in || || ||
```

```
older || || ||
patients || || ||
with || || ||
prior || || DIS || MIS
myocardial || DIS || DIS || COR
infarction || DIS || DIS || COR
at || || ||
the || || ||
time || || ||
of || || ||
admission || || ||
to || || ||
a || || ||
nursing || TREAT || || SPU
home || || ||
.  || || ||
```

From the tallies of all words in all test sentences I calculate the F-measure precision and recall measures (results shown in Section 3.7). Note that in my evaluation the tallies for *all* words have the same weight. In case of punctuation, for example, I could have chosen not to include them in the evaluation, but to be more rigorous I do include them. For example, for the sentence

```
<TREAT> Mitomycin, ifosfamide, and cisplatin </TREAT> in <DIS>
unresectable non-small-cell lung cancer </DIS>
```

the alignment found was

```
mitomycin || TREAT || TREAT || COR
, || || TREAT || MIS
ifosfamide || TREAT || TREAT || COR
, || || TREAT || MIS
and || || TREAT || MIS
cisplatin || TREAT || TREAT || COR
in || || ||
unresectable || DIS || DIS || COR
non-small-cell || DIS || DIS || COR
lung || DIS || DIS || COR
cancer || DIS || DIS || COR
```

that is, the system missed the punctuation that was part of the labeled roles. Although punctuation is unimportant for my task, I include all of the tallies in the evaluation.

For the task of relation classification, I simply calculate the classification accuracy, which is the percentage of the sentences for which the system prediction is correct.

Craven and Kumlien (1999) and Craven (1999) (and others) have a more "forgiving" evaluation policy: they say that "although each instance of the target relation may be represented multiple times in the corpus, we consider the IE-method to be correct as long as it extracts this instance from *one* of its occurrences." This could be fine for certain applications, however, I believe that for many applications it is important to return *all* the articles that talk about a certain relation. In my evaluation I consider *all* instances of the target roles.

As mentioned in Section 3.3, the annotation was done independently of any syntactic information and with no constraints whatsoever and this gives rise to some

inconsistency in the labeling. Correcting this would probably produce better results.

I run the experiments 1) for all sentences that were found to have the semantic roles of interest (relevant sentences) and 2) for *all* sentences (relevant + non-relevant), that is, sentences with semantic roles and sentences with no semantic role of interest. It is not always clear from the papers in the literature, but my understanding is that most report the results for only the relevant sentences, omitting an explanation of how a real system would distinguish between relevant and non-relevant sentences. As explained in Section 3.7, I not only report the results for the case in which we include the non-relevant sentences but I also propose a method to distinguish between relevant and non-relevant ones.

I had a second annotator annotate the relevant sentences. For these, the F-measures between the 2 annotations was 81% which gives us an upper limit for the system performance.[32]

[32]More precisely, this value is a limit on how much we believe the annotation, assuming the system in theory can do better given better annotations. However, we expect that if the task is so difficult that the annotators do not agree, it would be difficult for the system to perform (much) better than the annotators.

Some examples of the diverging annotations are:

- – Administration of <TREAT> dexamethasone </TREAT> induces <DIS> proteinuria of glomerular origin </DIS> in mice.
  – Administration of <TREAT> dexamethasone </TREAT> induces <DIS> proteinuria </DIS> of glomerular origin in mice.
- – Both groups received similar <TREAT> antibiotic </TREAT> and <TREAT> insulin </TREAT> treatment.
  – Both groups received similar <TREAT> antibiotic and insulin treatment </TREAT>.
- – How long should <TREAT> suction drains </TREAT> stay in after <TREAT> breast surgery with axillary dissection </TREAT> ?
  – How long should suction drains stay in after <TREAT> breast surgery </TREAT> with <TREAT> axillary dissection </TREAT> ?
- – Mutants of cholera toxin as an effective and safe adjuvant for <TREAT> nasal influenza vaccine </TREAT>.
  – Mutants of cholera toxin as an effective and safe adjuvant for <DIS> nasal influenza </DIS> <TREAT> vaccine </TREAT>.

3.7 Models

The goal of this work is twofold: first, to identify the semantic roles DIS and TREAT, given a natural language sentence (this is a segmentation task), and second to identify the semantic relation, if any, that holds between them. This section describes the models and their performance on both entity extraction and relation classification. The relationships are those described in Section 3.3.1.

I evaluate five generative models (two static and three dynamic) and one discriminative model. Discriminative models learn the probability of the class given the features. When we have fully observed data and we just need to learn the mapping from features to classes (classification), a discriminative approach may be more appropriate, as shown in Ng and Jordan (2002).

Generative models learn the prior probability of the class and the probability of the features given the class and are the natural choice in cases with hidden variables (partially observed or missing data). Since labeled data is expensive to collect, these models may be useful when no labels are available. However, in this paper I test the generative models on fully observed data.

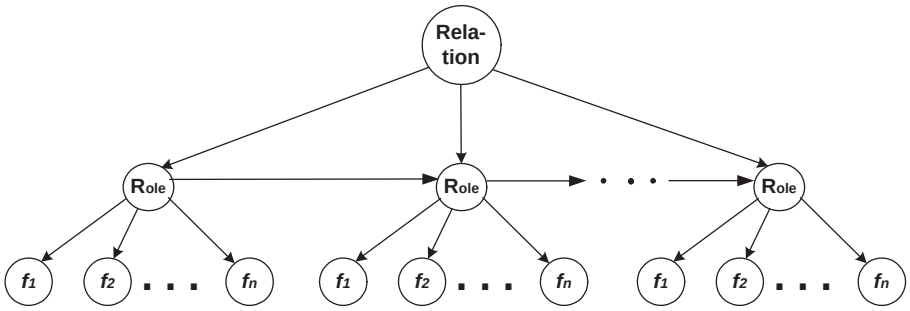
3.7.1 Generative Models

In Figure 3.3 I show the three dynamic models and in Figure 3.4 the two static models that I designed and implemented for these tasks. The nodes labeled "Role" represent the entities (in this case the choices are DIS, TREAT and NULL) and the node labeled "Relation" represents the relationship present in the sentence. I assume here that there is a single relation for each sentence between the entities. The children of the role nodes are the words and their features, thus there are as many role states as there are words in the sentence; for each state, the features f_i are those mentioned in Section 3.5. For the static models, this is depicted by the box (or "plate") which

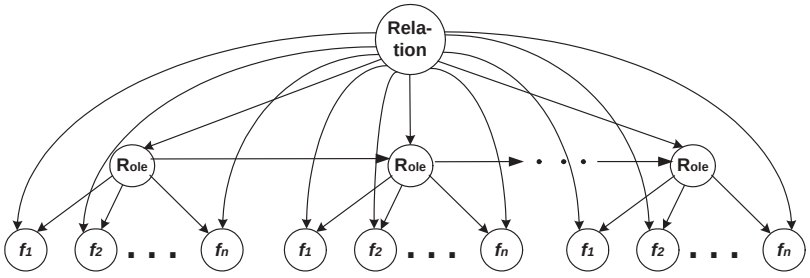
is the standard graphical model notation for replication (Spiegelhalter et al., 1996). For clarity, in Figure 3.5 are the same static models of Figure 3.4 without the plate notation. One can see that the static models S1 and S2 are missing the transitions between states. In other words, they do not assume an ordering in the role sequence.

The dynamic models were inspired by prior work on HMM-like graphical models for role extraction (Bikel et al., 1999; Freitag and McCallum, 2000; Ray and Craven, 2001). These models consist of a Markov sequence of states (usually corresponding to semantic roles) where each state generates one or multiple observations. Model D1 in Figure 3.3 is typical of these models, but I have augmented it with the Relation node.

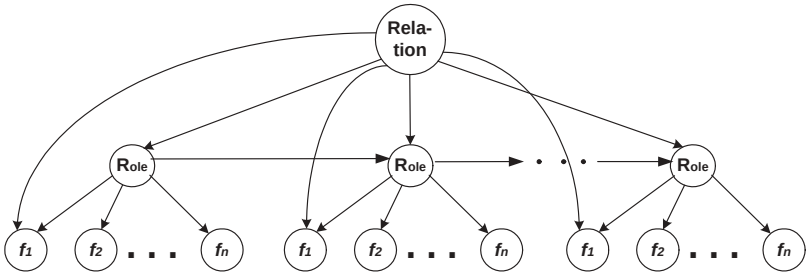
The task is to recover the sequence of Role states, given the observed features. These models assume that there is an ordering in the semantic roles that can be captured with the Markov assumption and that the role generates the observations (the words, for example). All my models make the additional assumption that there is a relation that generates the role sequence; thus, these models have the appealing property that they can *simultaneously* perform role extraction and relationship recognition, given the sequence of observations. In S1 and D1 the observations are independent from the relation (given the roles). In S2 and D2, the observations are dependent on both the relation and the role (or in other words, the relation generates not only the sequence of roles but also the observations). D2 encodes the fact that even when the roles are given, the observations depend on the relation. For example, sentences containing the word *prevent* are more likely to represent a "prevent" kind of relationship. Finally, in D3 only one observation per state is dependent on both the relation and the role, the motivation being that some observations (such as the words) depend on the relation while others might not (like for example, the parts of speech). In the experiments reported here, the observations which have edges from both the role and the relation nodes are the words. (I ran an experiment in which



Dynamic model (D1)

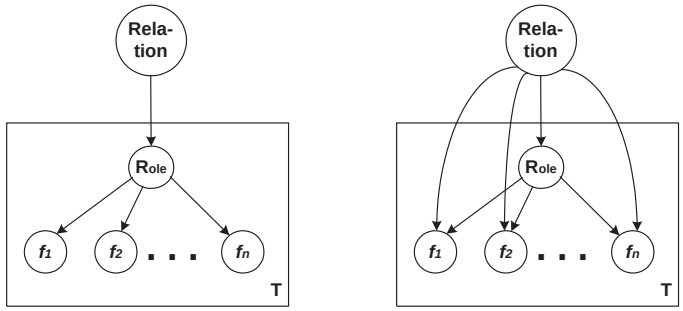


Dynamic model (D2)



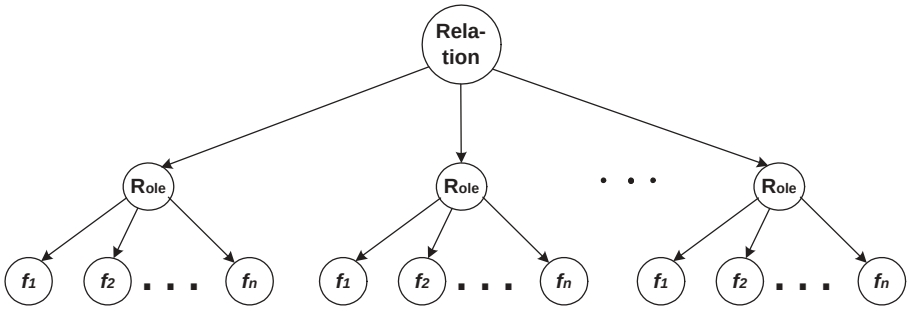
Dynamic model (D3)

Figure 3.3: Dynamic models for role and relation classification.

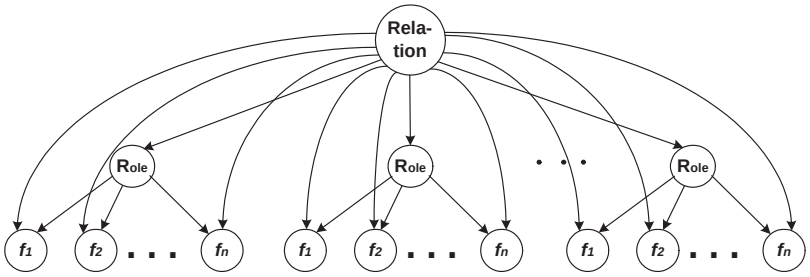


Static (S1) Static (S2)

Figure 3.4: Static models for role and relation classification.



Static (S1)



Static (S2)

Figure 3.5: Static models of Figure 3.4 without the plate notation. Note how the transitions between the role nodes are missing.

this observation node was the MeSH term, obtaining similar results.)

Model D1 defines the following joint probability distribution over relations, roles, words and word features, assuming the leftmost Role node is $Role_0$, and T is the number of words in the sentence:

$$\begin{aligned} & P(Rel, Role_0, .., Role_T, f_{10}, .., f_{n0}, ..., f_{1T}, .., f_{nT}) \\ = \; & P(Rel)P(Role_0 \mid Rel) \prod_{j=1}^{n} P(f_{j0} \mid Role_0) \\ & \prod_{t=1}^{T} P(Role_t \mid Role_{t-1}, Rel) \prod_{j=1}^{n} P(f_{jt} \mid Role_t) \end{aligned} \tag{3.1}$$

Model D1 is similar to the model in Thompson et al. (2003) for the extraction of roles, using a different domain. Structurally, the differences are (i) Thompson et al. (2003) has only one observation node per role and (ii) it has an additional node "on top," with an edge to the relation node to represent a predicator "trigger word" which is always observed; the predicator words are taken from a fixed list and one must be present in order for a sentence to be analyzed.

The joint probability distributions for D2 and D3 are similar to Equation (1) where I substitute the term $\prod_{j=1}^{n} P(f_{jt}|Role_t)$ with $\prod_{j=1}^{n} P(f_{jt}|Role_t, Rel)$ for D2 and $P(f_{1t}|Role_t, Rel) \prod_{j=2}^{n} P(f_{jt}|Role_t)$ for D3. The parameters $P(f_{jt}|Role_t)$ and $P(f_{j0}|Role_0)$ of Equation (1) are constrained to be equal.

The joint probability distribution for the static model S1 of Figure 3.4 is the following:

$$
\begin{aligned}
& P(Rel, Role_0, .., Role_T, f_{10}, .., f_{n0}, ..., f_{1T}, .., f_{nT}) \\
&= P(Rel) \prod_{t=1}^{T} P(Role_t \mid Rel) \prod_{j=1}^{n} P(f_{jt} \mid Role_t)
\end{aligned} \tag{3.2}
$$

where I substitute the term $\prod_{j=1}^{n} P(f_{jt}|Role_t)$ with $\prod_{j=1}^{n} P(f_{jt}|Role_t, Rel)$ for S2.

The parameters were estimated using maximum likelihood on the training set; I also implemented a simple absolute discounting smoothing method (Zhai and Lafferty, 2001) that improves the results for both tasks.

Table 3.2 shows the results (F-measures) for the problem of finding the most likely sequence of roles given the features observed. In this case, the relation is hidden and I marginalize over it.[33] I experimented with different values for the smoothing discount factor ranging from a minimum of 0.0000005 to a maximum of 10; the results shown fix the smoothing factor at its minimum value. I found that for the dynamic models, for a wide range of smoothing factors, I achieved almost identical results. By contrast, the static models were more sensitive to the value of the smoothing factor.

Using maximum likelihood with no smoothing, model D1 performs better than D2 and D3. This was expected, since the parameters for models D2 and D3 are more sparse than D1. However, when smoothing is applied, the three dynamic models achieve similar results. Although the additional edges in models D2 and D3 did not help much for the task of role extraction, they did help for relation classification, discussed next. Model D2 achieves the best F-measures: 0.73 for "only relevant" and 0.71 for "rel. + non-rel."; the inter-annotator agreement was a F-measure of 0.81 for "only relevant" which gives us an upper limit of the F-measure.

[33]To perform inference for the dynamic model, I used the junction tree algorithm. I used Kevin Murphy's BNT package:
http://www.cs.ubc.ca/ murphyk/Software/BNT/bnt.html

Sentences	**Static**		**Dynamic**		
	S1	**S2**	**D1**	**D2**	**D3**
	No Smoothing				
Only rel.	0.67	0.68	0.71	0.52	0.55
Rel. + non-rel.	0.61	0.62	0.66	0.35	0.37
	Absolute discounting				
Only rel.	0.67	0.68	0.72	**0.73**	0.73
Rel. + non-rel.	0.60	0.62	0.67	**0.71**	0.69

Table 3.2: F-measures for the models of Figures 3.4 and 3.3 for *role* extraction.

It is difficult to compare results with the related work since the data, the semantic roles and the evaluation are different; in Ray and Craven (2001) however, the role extraction task is quite similar to mine and the text is also from MEDLINE. They report approximately an F-measure of 32% for the extraction of the entities PROTEINS and LOCATIONS, and an F-measure of 50% for GENE and DISORDER.

The second target task is to find the most likely relation, i.e., to classify a sentence into one of the possible relations. Two types of experiments were conducted. In the first, the true roles are hidden and I classify the relations given only the observable features, marginalizing over the hidden roles. In the second, the roles are given and only the relations need to be inferred. Table 3.3 reports the results for both conditions, both with absolute discounting smoothing and without.

Again model D1 outperforms the other dynamic models when no smoothing is applied; with smoothing and when the true roles are hidden, D2 achieves the best classification accuracies. When the roles are given D1 is the best model; D1 does well in the cases when both roles are not present. By contrast, D2 does better than D1 when the presence of specific words strongly determines the outcome (e.g., the presence of "prevention" or "prevent" helps identify the Prevent relation).

The percentage improvements of D2 and D3 versus D1 are, respectively, 10% and 6.5% for relation classification and 1.4% for role extraction (in the "only relevant,"

Sentences	Input	B	Static		Dynamic			NN
			S1	S2	D1	D2	D3	
			No Smoothing					
Only rel.	only feat.	46.7	51.9	50.4	65.4	58.2	61.4	**79.8**
	roles given		51.3	52.9	66.6	43.8	49.3	**92.5**
Rel. + non-rel.	only feat.	50.6	51.2	50.2	68.9	58.7	61.4	**79.6**
	roles given		55.7	54.4	82.3	55.2	58.8	**96.6**
			Absolute discounting					
Only rel.	only feat.	46.7	51.9	50.4	66.0	72.6	70.3	
	roles given		51.9	53.6	83.0	76.6	76.6	
Rel. + non-rel.	only feat.	50.6	51.1	50.2	68.9	74.9	74.6	
	roles given		56.1	54.8	91.6	82.0	82.3	

Table 3.3: Accuracies of *relationship* classification for the models in Figures 3.4 and 3.3 and for the neural network (NN). For absolute discounting, the smoothing factor was fixed at the minimum value. B is the baseline of always choosing the most frequent relation. The best results are indicated in boldface.

"only features" case). This suggests that there is a dependency between the observations and the relation that is captured by the additional edges in D2 and D3, but that this dependency is more helpful in relation classification than in role extraction.

For relation classification the static models perform worse than for role extraction; the decreases in performance from D1 to S1 and from D2 to S2 are, respectively (in the "only relevant," "only features" case), 7.4% and 7.3% for role extraction and 27.1% and 44% for relation classification. This suggests the importance of modeling the sequence of roles for relation classification.

To provide an idea of where the errors occur, Table 3.4 shows the confusion matrix for model D2 for the most realistic and difficult case of "rel + non-rel.," "only features." This indicates that the algorithm performs poorly primarily for the cases for which there is little training data, with the exception of the ONLY DISEASE case, which is often mistaken for CURE.

3.7.2 Neural Network

To compare the results of the generative models of the previous section with a discriminative method, I use a neural network (NN), using the Matlab package to train a feed-forward network with conjugate gradient descent.

The features are the same as those used for the models in Section 3.7.1, but are represented with indicator variables. That is, for each feature I calculated the number of possible values v and then represented an observation of the feature as a sequence of v binary values in which one value is set to 1 and the remaining $v - 1$ values are set to 0.

The input layer of the NN is the concatenation of this representation for all features. The network has one hidden layer, with a hyperbolic tangent function. The output layer uses a logistic sigmoid function. The number of units of the output layer is fixed to be the number of relations (seven or eight) for the relation classification task and the number of roles (three) for the role extraction task. The network was trained for several choices of numbers of hidden units; I chose the best-performing networks based on training set error. I then tested these networks on held-out testing data.

The results for the neural network are reported in Table 3.3 in the column labeled NN. These results are quite strong, achieving 79.6% accuracy in the relation classification task when the entities are hidden and 96.9% when the entities are given, outperforming the graphical models. Two possible reasons for this are: as already mentioned, the discriminative approach may be the most appropriate for fully labeled data; or the graphical models I proposed may not be the right ones, i.e., the independence assumptions they make may misrepresent underlying dependencies.

It must be pointed out that the neural network is much slower than the graphical models, and requires a great deal of memory; I was not able to run the neural network

	Prediction								Relation
Truth	**V**	**OD**	**NC**	**C**	**P**	**OT**	**SE**	**Irr.**	**accuracy**
Vague (V)	0	3	0	4	0	0	0	1	0
Only DIS (OD)	2	69	0	27	1	1	0	24	55.6
No Cure (NC)	0	0	0	1	0	0	0	0	0
Cure (C)	2	5	0	150	1	1	0	3	92.6
Prevent (P)	0	1	0	2	5	0	0	5	38.5
Only TREAT (OT)	0	0	0	16	0	6	1	11	17.6
Side effect (SE)	0	0	0	3	1	0	0	1	20
Non-relevant	1	32	1	16	2	7	0	296	83.4

Table 3.4: Confusion matrix for the dynamic model D2 for "rel + non-rel.," "only features." In the last column the classification accuracies for each relation. The total accuracy for this case is 74.9%.

package on my machines for the role extraction task, when the feature vectors are very large. The graphical models can perform both tasks simultaneously; the percentage decrease in relation classification of model D2 with respect to the NN is of 8.9% for "only relevant" and 5.8% for "relevant + non-relevant."

I should also point out that the neural network has a hidden layer which is responsible for significantly expanding the class of functions that it is able to fit to data. The graphical models that we have explored are more limited. Indeed, these models are parametric while the neural networks are nonparametric. It's not clear if the superior performance of the neural network is due to its discriminative nature or to its nonparametric nature. To perform a fair comparison in terms of generative versus discriminative approaches, parametric discriminative models (e.g., those lacking a hidden layer, such as a perceptron) should be tried on this data.

3.7.3 Features impact

In order to analyze the relative importance of the different features, we performed both tasks using the dynamic model D1 of Figure 3.3, leaving out single features and

Relation	**Num. Sent.** Train, Test
Vague	28, 8
Only DIS	492, 124
No Cure	3, 1
Cure	648, 162
Prevent	50, 13
Only TREAT	132, 34
Side effect	24, 5
Non-relevant	1416, 355

Table 3.5: The numbers of training and testing sentences for each relation.

sets of features (grouping all of the features related to the MeSH hierarchy, meaning both the classification of words into MeSH IDs and the domain knowledge as defined in Section 3.4). The results reported here were found with maximum likelihood (no smoothing) and are for the "relevant only" case; results for "relevant + non-relevant" were similar.

For the role extraction task, the most important feature was the word: not using it, the GM achieved only 0.65 F-measure (a decrease of 9.7% from 0.72 F-measure using all the features). Leaving out the features related to MeSH the F-measure obtained was 0.69% (a 4.1% decrease) and the next most important feature was the part-of-speech (0.70 F-measure not using this feature). For all the other features, the F-measure ranged between 0.71 and 0.73.

For the task of relation classification, the MeSH-based features seem to be the most important. Leaving out the word again lead to the biggest decrease in the classification accuracy for a single feature but not so dramatically as in the role extraction task (62.2% accuracy, for a decrease of 4% from the original value), but leaving out all the MeSH features caused the accuracy to decrease the most (a decrease of 13.2% for 56.2% accuracy). For both tasks, the impact of the domain knowledge

alone was negligible.

As described in Section 3.4, words can be mapped to different levels of the MeSH hierarchy. Currently, I use the "second" level, so that, for example, *surgery* is mapped to G02.403 (when the whole MeSH ID is G02.403.810.762). This is somewhat arbitrary (and mainly chosen with the sparsity issue in mind), but in light of the importance of the MeSH features it may be worthwhile investigating the issue of finding the optimal level of description. (This can be seen as another form of smoothing.)

3.8 Conclusions

I have addressed the problem of distinguishing between several different relations that can hold between two semantic entities, a difficult and important task in natural language understanding. Because there is no existing gold-standard for this problem, I have developed the relation definitions of Table 3.3; this however may not be an exhaustive list. I have presented five graphical models and a neural network for the tasks of semantic relation classification and role extraction from bioscience text. The methods proposed yield quite promising results. The graphical models perform these tasks simultaneously.

The next chapter shows how these models perform for another set of relation types, protein-protein interactions.

Chapter 4

Labeling protein-protein interactions

4.1 Introduction

Identifying the interactions between proteins is one of the most important challenges in modern genomics, with applications throughout cell biology, including expression analysis, signaling, docking, and rational drug design. Most biomedical research is available electronically, but only in free-text format. Automatic mechanisms are needed to convert the text into more structured forms.

In this chapter, I address the problem of multi-way relation classification, applied to identification of the interactions between proteins in bioscience text. I use the models described in Chapter 3 that were found to achieve high accuracy in the related task of extracting TREATMENT - DISEASE relations.

Labeling the training and test data is a time-consuming and subjective process. Here I report on results using an existing curated database, the HIV-1 Human Protein Interaction Database,[1] to train and test the classification systems.

The accuracies obtained by the classification models proposed are quite high,

[1] www.ncbi.nlm.nih.gov/RefSeq/HIVInteractions/index.html

confirming the viability of the approach.

I also find support for the hypothesis that the sentences surrounding citations are useful for extraction of key information from technical articles (Nakov et al., 2004).

4.2 Related work

In the BioNLP literature there have recently been a number of attempts to automatically extract protein-protein interactions from PubMed abstracts. Some approaches simply report that a relation exists between two proteins but do not determine which relation holds (Bunescu et al., 2005; Marcotte et al., 2001; Ramani et al., 2005), while most others start with a list of interaction words and label only those sentences that contain these trigger verbs (Blaschke and Valencia, 2002; Blaschke et al., 1999a; Rindflesch et al., 1999; Thomas et al., 2000; Sekimizu et al., 1998; Ahmed et al., 2005; Phuong et al., 2003; Pustejovsky et al., 2002).

Most of the existing methods also suffer from low recall because they use hand-built specialized templates or patterns (Ono et al., 2001; Corney et al., 2004). Moreover, often the proteins involved are assumed to be given.

For this work, I use state-of-the-art machine learning methods to determine the interaction *types* and also to extract the proteins involved. I do not use interaction words, templates, or dictionaries.

See Section 3.2 for an in-depth discussion of the related work on role and relation extraction.

4.3 Data

I use the information from a domain-specific database to gather labeled data for the task of classifying the interactions between proteins in text. The HIV-1 Human Pro-

tein Interaction Database provides a summary of documented interactions between HIV-1 proteins and host cell proteins, other HIV-1 proteins, or proteins from disease organisms associated with HIV or AIDS. This database is manually curated. I use this database also because it contains information about the *type* of interactions, as opposed to other protein interaction databases (BIND, MINT, DIP, for example[2]) that list the protein pairs interacting, without specifying the type of interactions.

The database contains 65 types of interactions and 809 proteins for which there is interaction information, with a total of 2224 pairs of interacting proteins.

In this database, the definitions of the interactions depend on the proteins involved (and the articles describing the interaction), thus there are several definitions for each interaction type.[3] As an example, for the interaction *bind* and the proteins *ANT* and *Vpr*, we find (among others) the definition *"Interaction of HIV-1 Vpr with human adenine nucleotide translocator (ANT) is presumed based on a specific binding interaction between Vpr and rat ANT"*; for *bind* and the proteins *actin, gamma 1* and *gag* we find the following definition: *"Mature HIV-1 Nucleocapsid, as well as the nucleocapsid domain of the HIV-1 Gag polyprotein, binds filamentous actin resulting in incorporation of actin into virus particles and enhancement of cell motility."*

For each documented protein-protein interaction the database includes information about:

- A pair of proteins (PP),
- The interaction type(s) between them (I), and

[2]DIP lists only the protein pairs; BIND additionally includes some information about the method used to provide evidence for the interaction; MIND contains interaction type information but the vast majority of the entries (99.9% of the 47,000 pairs) are assigned the same type of interaction (*aggregation*). These databases are all manually curated.
DIP (Database of Interacting Proteins): http://dip.doe-mbi.ucla.edu
BIND (Biomolecular Interaction Network Database): http://bind.ca
MINT (Molecular Interactions Database): http://160.80.34.4/mint

[3]There are 1001 descriptions for the 65 interactions, for an average of 15.4 descriptions per protein (max = 138 for *binds*, min = 1).

Interaction	#Triples	Interaction	#Triples
Interacts with	1115	*Complexes with*	45
Activates	778	*Modulates*	43
Stimulates	659	*Enhances*	41
Binds	647	*Stabilizes*	34
Upregulates	316	*Myristoylated by*	34
Imported by	276	*Recruits*	32
Inhibits	194	*Ubiquitinated by*	29
Downregulates	124	*Synergizes with*	28
Regulates	86	*Co-localizes with*	27
Phosphorylates	81	*Suppresses*	24
Degrades	73	*Competes with*	23
Induces	52	*Requires*	22
Inactivates	51		

Table 4.1: Number of triples for the most common interactions of the HIV-1 database, after removing the distinction in directionality and the triples with more than one interaction.

- PubMed identification numbers of the journal article(s) describing the interaction(s) (A).

A protein pair PP can have multiple interactions (for example, UNG2 *binds* to Gag-Pol and also *incorporates* it) for an average of 1.9 interactions per PP and a maximum of 23 interactions, for the pair CDK9 and TAT p14.

I refer to the combination of a protein pair PP and an article A as a "triple." The database associates to each triple an interaction type and my goal is to develop systems that do this automatically. For the example above, the triple [UNG2 Gag-Pol 12667798] is assigned the interaction *binds* (12667798 being the PubMed number of the paper providing evidence for this interaction). (To be precise, there are for this PP, as there are often, multiple articles, three in this case, describing the interaction *binds*, thus I have the following three triples to which I associate *binds*: [UNG2 Gag-Pol 12667798], [UNG2 Gag-Pol 9882380] and [UNG2 Gag-Pol 12458223]. The

triples [UNG2 Gag-Pol 12670953] [UNG2 Gag-Pol 12458223] describe the interaction *incorporates*.)

Journal articles can contain evidence for multiple interactions: there are 984 journal articles in the database and on average each article is reported to contain evidence for 5.9 PP (with a maximum number of 90 PP).

In some cases the database reports multiple different interactions for a given triple. There are 5369 unique triples in the database and of these 414 (7.7%) have multiple interactions. I exclude these triples from my analysis; however, I do include articles and PPs with multiple interactions. In the example above I exclude the triple [UNG2 Gag-Pol 12458223] (to which the database assign both *binds* and *incorporates*) but I include all the others, since the evidence for the different interactions is given by different articles.

Some of the interactions differ only in the directionality (e.g., *regulates* and *regulated by*, *inhibits* and *inhibited by*, etc.); I collapsed these pairs of related interactions into one.[4] I did this because the directionality of the interactions was not always reliable in the database (some pairs were assigned one interaction and the same interaction in the opposite direction as well). Table 4.1 shows the list of the 25 interactions of the HIV-1 database for which there are more than 10 triples. (The most frequent interaction is the generic *interacts with* that indicates that the two proteins are known to interact, without specifying the nature of the interaction.)

For these interactions and for a random subset of the protein pairs PP (around 45% of the total pairs in the database), I downloaded the corresponding PubMed papers. From these, I extracted all and only those sentences that contain both proteins from the indicated protein pair. I assigned each of these sentences the corresponding

[4]This implies that for some interactions, I am not able to infer the different roles of the two proteins; moreover I considered only the pair "prot1 prot2" or "prot2 prot1," not both. However, as described in Section 4.5.3, my algorithm can detect which are the proteins involved in the interactions.

interaction I from the database (I call this group "papers").

Nakov et al. (2004) argue that the sentences surrounding citations to related work, or *citances*, are a useful resource for BioNLP. Building on that work, I use citances as an additional form of evidence to determine protein-protein interaction types. For a given database entry containing PubMed article A, protein pair PP, and interaction type I, I downloaded a subset of the papers that cite A. From these citing papers, I extracted all and only those sentences that mention A explicitly; I further filtered these to include all and only the sentences that contain PP. I labeled each these sentences with interaction type I (I call this group "citances").

As an example, for the triple [AIP1 Gag 14519844], I extract from the target paper A (PubMedID 14519844) the following sentences:

- *The interpretation of these results was slightly complicated by the fact that AIP-1/ALIX depletion by using siRNA likely had deleterious effects on cell viability , because a Western blot analysis showed slightly reduced Gag expression at later time points (fig. 5C).*

- *Gag p6 - p6 - , Gag pb - p9 - , and Gag pd - PTAP - complemented HIV - 1 was generated as in fig. 4 , but , in this case , luciferase (control) - , Tsg101 - , or AIP-1/ALIX - specific siRNAs were cotransfected .*

From the papers that cite A, I extract:

- *They also demonstrate that the GAG protein from membrane - containing viruses , such as HIV , binds to Alix / AIP1 , thereby recruiting the ESCRT machinery to allow budding of the virus from the cell surface (TARGET_CITATION; CITATION) .*

- *Recently , the entire cellular protein network that participates in HIV - 1 budding was mapped , with TSG101 and AIP1 identified as direct interaction partners*

of the Gag p6 domain (TARGET_CITATION, CITATION, CITATION) .

where "TARGET_CITATION" is the token with which I substitute the reference to paper A and "CITATION" is the token for all other references.

There are often many different names for the same protein (as for AIP-1 and ALIX in the examples above). I use LocusLink[5] protein identification numbers and synonym names for each protein, and extract the sentences that contain an exact match for (some synonym of) each protein. By being conservative with protein name matching, and by not doing co-reference analysis, I miss many candidate sentences; however this method is very precise.

On average, for "papers," I extracted 0.5 sentences per triple (maximum of 79) and 50.6 sentences per interaction (maximum of 119); for "citances" I extracted 0.4 sentences per triple (with a maximum of 105) and 49.2 sentences per interaction (162 maximum). I required a minimum number (40) of sentences for each interaction type for both "papers" and "citances"; the 10 interactions of Table 4.2 met this requirement. I used these sentences to train and test the models described below.[6]

Since all the sentences extracted from one triple are assigned the same interaction, I required the sentences in the training and test sets to originate from disjoint triples. Roughly 75% of the data were used for training and the rest for testing (this is not exactly 75% because the number of triples per paper is not uniform).

As mentioned above the goal is to automatically associate to each triple an interaction type. The task tackled here is actually slightly more difficult: given some sentences extracted from article A, assign to A an interaction type I and extract the proteins PP involved. In other words, for the purpose of classification, we act as if

[5]LocusLink was recently integrated into Entrez Gene, a unified query environment for genes (http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene).

[6]I also looked at larger chunks of text, in particular, I extracted the sentence containing the PP along with the previous and the following sentences, and the three consecutive sentences that contained the PP (the proteins could appear in any of the sentences). However, the results obtained by using these larger chunks were consistently worse, see discussion in Section 3.4.

Interaction	Papers	Citances
Degrades (Degr)	60	63
Synergizes with (SynerW)	86	101
Stimulates (Stim)	103	64
Binds (Bind)	98	324
Inactivates (Inact)	68	92
Interacts with (InterW)	62	100
Requires (Req)	96	297
Upregulates (Upreg)	119	98
Inhibits (Inhib)	78	84
Suppresses (Supp)	51	99
Total	821	1322

Table 4.2: Number of interaction sentences extracted.

we do not have information about the proteins that interact. However, given the way the sentence extraction was done, all the sentences extracted from A contain the PP.

By using the HIV-1 database as a way of collecting labeled data for training the models, I make the following rather strong assumption: given article A listed in the HIV-1 database as evidence for the interaction I between two proteins PP, assume that every sentence extracted from A that contains these two proteins expresses interaction I. Of course, this assumption will not always be correct, and has the side-effect of introducing noise into the labeled data (in Section 4.6 I examine how much noise), but it circumvents the need to hand-label all of the sentences in the corresponding papers in order to train the algorithm.

A hand-assessment of the individual sentences shows that not every sentence that mentions the target proteins PP actually describes the interaction I (see Section 4.6). Thus the evaluation on the test set can be done at the document level (to determine if the algorithm can predict the interaction that a curator would assign to a document as a whole given the protein pair) and at the individual sentence level to determine if the algorithm can assign the actual correct interaction to each sentence.

Note that I assume here that the papers that provide the evidence for the interactions are given, an assumption not usually true in practice.

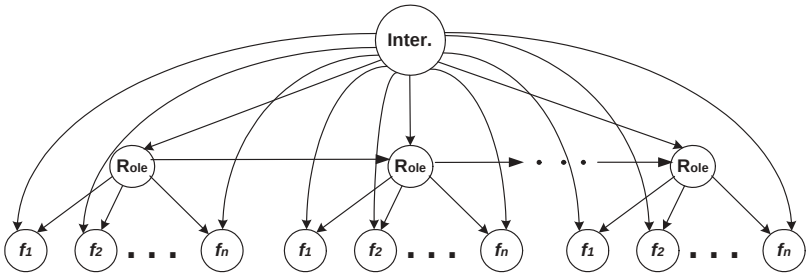
4.4 Models

For assigning interactions, I used generative graphical models similar to those described in Section 3.7.2, with some slight differences. Figure 4.1 shows the generative models. The nodes labeled "Role" represent the entities (in this case the choices are PROTEIN and NULL); the children of the role nodes are the words and their features, thus there are as many role states as there are words in the sentence; the dynamic model consists of a Markov sequence of states where each state generates one or multiple observations. This model makes the additional assumption that there is an interaction present in the sentence (represented by the node "Inter.") that generates the role sequence and the observations. (I assume here that there is a single interaction for each sentence between the proteins.)

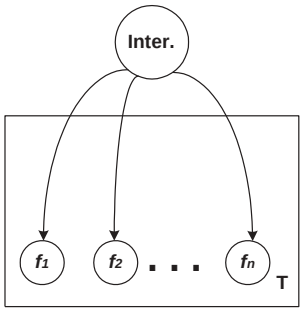
The static model is a simple Naive Bayes, in which the node representing the interaction generates the observable features. Note that this model is slightly different from the static models described in Section 3.7 in that the model does not include role information.

The "Role" nodes can be observed or hidden; here they are hidden and marginalized over for the interaction classification (i.e., I had no information regarding which proteins were involved). In Section 4.5.3 I describe experimental results for the role extraction task.

I also used a neural network setup with the same settings as that described in Section 3.7.2.



Dynamic (DM)



Static (NB)

Figure 4.1: Dynamic (DM) and static (NB) graphical models for protein interaction classification (and role extraction).

4.5 Results

The task is the following: predict one of the interactions of Table 4.2 for a given triple, given the sentences extracted for that triple. This is a 10-way classification problem, a difficult problem and significantly more complex than much of the related work in which the task is to predict whether there is an interaction or not (see Section 4.2).

The results reported here were obtained using two sets of features:

1. Only words (i.e., in the dynamic model of Figure 4.1 there is only one feature node per role); results in rows "words" of Table 4.3.

2. Words along with some semantic features: MeSH terms, the GO codes codes and whether the word is a "molecular function," a "biological process" or a "cellular component."[7] In this case the dynamic model DM has three feature nodes per role. The results for this case are in rows "+sem." of Table 4.3.

I defined joint probability distributions over these models, estimated using maximum likelihood on the training set with a simple absolute discounting smoothing method. I performed 10-fold cross validation on the training set and I chose the smoothing parameters for which I obtained the best classification accuracies (averaged over the ten runs) on the training data; the results reported here were obtained using these parameters on the held-out test sets.[8]

The evaluation was done on a document-by-document basis. During testing, I choose the interaction using the following aggregate measures that use the constraint

[7]GO, the Gene Ontology consists of three structured, controlled vocabularies (ontologies) that describe gene products. The three organizing principles of GO are molecular function, biological process and cellular component. http://www.geneontology.org/GO.evidence.shtml.
I used a program that matches strings to gene names and finds the GO codes for the genes.

[8]I didn't have enough data to require that the sentences in the training and test sets *of the cross validation procedure* originate from disjoint triples (they do originate from disjoint triple in the final held out data). This may result in a less than optimal choice of the parameters for the aggregate measures described below.

		Mj	Mj*	Cf	Cf*
		All (papers + citances)			
Baseline (Mf)		21.8			
DM	words	60.5	59.7	59.7	59.7
	+sem.	58.1	59.7	58.1	61.3
NB	words	58.1	58.9	61.3	59.7
	+sem.	59.7	58.9	59.7	58.9
NN	words	**63.7**	62.9		
	+sem.	**63.7**	62.9		
Key		20.1			
KeyB		25.8			
		Papers			
Baseline (Mf)		11.1			
DM	words	**57.8**	46.7	55.6	55.6
	+sem.	40.0	40.0	44.4	53.3
NB	words	**57.8**	**57.8**	53.3	**57.8**
	+sem.	40.0	40.0	46.7	44.4
NN	words	44.4	44.4		
	+sem.	48.9	48.9		
Key		24.4			
KeyB		40.0			
		Citances			
Baseline (Mf)		26.1			
DM	words	53.4	54.5	54.5	55.7
	+sem.	60.2	59.1	**61.4**	57.9
NB	words	55.7	55.7	54.5	54.5
	+sem.	56.8	55.7	55.7	56.8
NN	words	55.8	53.4		
	+sem.	58.0	58.0		
Key		20.4			
KeyB		26.1			

Table 4.3: Accuracies for classification of the 10 protein-protein interactions of Table 4.2 when evaluating at the document level. Best results shown in boldface. DM: the dynamic model, NB: the Naive Bayes (Figure 4.1), NN the neural network. For each model, the first line gives the results when only the words are used as input, the second line, the results adding the semantic features (MeSH and GO codes). Key: the trigger word approach, KeyB: same as Key with backing off to the most frequent interaction when no interaction predicted (see Section 4.5.2). The Baseline (Mf) is accuracy for choosing the most frequent interaction. Chance is 10%.

that all sentences coming from the same triple are assigned the same interaction.

- **Mj**: For each triple, for each sentence of the triple, find the interaction that maximizes the posterior probability of the interaction given the features; then assign to *all* sentences of this triple the most frequent interaction between those predicted for the individual sentences.

- **Mj***: Same as Mj, except that if the interaction predicted is the generic *interacts with*, choose instead the next most frequent interaction (retain *interacts with* only if it is the only interaction predicted).

- **Cf**: Retain all the conditional probabilities (i.e., don't first choose an interaction per sentence), then for each triple choose the interaction that maximizes the sum over all the sentences of the triple.

- **Cf***: Same as Cf, substituting *interacts with* with the next most confident interaction.

Table 4.3 reports the results in terms of classification accuracies averaged across all interactions, for the cases "all" (sentences from "papers" and "citances" together), only "papers" and only "citances"; separating "papers" and "citances" allowed us to analyze the difference in performance when using the sentences from the original articles and when using those from the articles that cite them.

The accuracies are quite high; the dynamic model achieves around 60% for "all," 58% for "papers" and 61% for "citances." The neural net achieves the best results for "all" with around 64% accuracy.

From the results in Table 4.3 I can make the following observations:

- Accuracy using only "citances" is in most cases higher than the accuracy using

only "papers."[9] Accuracy of "all" is the highest, perhaps due to the larger training set for this case.

- The four aggregate measures achieve similar results.
- The performances of the dynamic model DM, the Naive Bayes NB and the NN are very similar, but the best results were obtained with the dynamic model DM (except for the case "all," where the neural net did better).
- Using the semantic features I obtained lower accuracies, especially for "papers" – but not for "citances" (see Section 3.7.3 for a discussion of the impact of the MeSH features for the classification of the relations that hold between TREAT and DIS.) It may be the case that the semantic features used here are not appropriate for this task, but further analysis is needed.
- All models perform significantly better than the baselines (that are: chance, most frequent class and a trigger word approach described in Section 4.5.2).

In the confusion matrix in Table 4.4 we can see the accuracies for the individual interactions for the dynamic model DM, "all" and "Mj." For three interactions (*degrades, inactivates, suppresses*) this model achieves perfect accuracy.

4.5.1 Hiding the protein names

In order to ensure that the algorithm was not over-fitting on the protein names, I ran an experiment in which I replaced the protein names in all sentences with the token "PROT_NAME." For example, the sentence: *"Selective CXCR4 antagonism by Tat"* became: *"Selective PROT_NAME2 antagonism by PROT_NAME1."*

[9]Note that, however, the baseline of always choosing the most frequent interaction is higher for "citances" than for "papers."

	Prediction										Acc.
Truth	***D***	***Sy***	***St***	***B***	***Ina***	***IW***	***R***	***Up***	***Inh***	***Su***	**(%)**
Degrad	5	0	0	0	0	0	0	0	0	0	100.0
Synerg	0	1	0	0	0	1	0	3	3	0	12.5
Stimul	0	0	4	0	0	0	6	0	1	0	36.4
Binds	0	0	0	18	0	4	1	1	3	0	66.7
Inacti	0	0	0	0	9	0	0	0	0	0	100.0
Intera	0	0	4	3	0	5	1	0	1	2	31.2
Requir	0	0	0	0	0	3	3	0	1	1	37.5
Upregu	0	0	0	2	1	0	0	12	2	0	70.6
Inhibi	0	0	0	3	0	0	1	1	12	0	70.6
Suppre	0	0	0	0	0	0	0	0	0	6	100.0

Table 4.4: Confusion matrix for the dynamic model DM for "all," "Mj." The overall accuracy is 60.5%. The numbers indicate the numbers of articles A (for each paper we have several sentences).

By inspection, over-fitting seems unlikely; the average number of (distinct) interaction types per protein is 1.8 (max = 25) and per PP is 1.3 (max = 9); 38% of the proteins and 20% of the PPs have multiple interaction types. (These numbers are slightly different from the numbers given in Section 4.3 which are based on the entire HIV-1 database; here I consider only the PPs I used in my experiments.)

In Table 4.5 the first rows for each model show the results of running the models on this data (the second rows are the corresponding results taken from Table 4.3 when the protein names were given, "original"). These results were obtained using only the words as features. The last column of Table 4.5 show the difference in accuracy (in percentage) with respect to the original case, for each model averaged over all evaluation methods.

For "papers" and "citances" there is always a decrease in the classification accuracy when I remove the protein names, showing that the protein names do help the classification. The differences in accuracy in the two cases using "citances" are much smaller than the differences using "papers" at least for the graphical models. This

suggests that citation sentences may be more robust for some language processing tasks and that the models that use "citances" learn better the linguistic context of the interactions.

4.5.2 Using a "trigger word" approach

As mentioned above, much of the related work in this field makes use of "trigger words" or "interaction words" (see Section 4.2). In order to (roughly) compare my work and to build a more realistic baseline, I created a list of 70 keywords that are representative of the 10 interactions. For example, for the interaction *degrade* some of the keywords are "degradation," "degrade," for *inhibit* I have "inhibited," "inhibitor," "inhibitory" and others. I then checked whether a sentence contained such keywords. If it did, I assigned to the sentence the corresponding interaction. If it contained more than one keyword corresponding to multiple interactions consisting of the generic *interact with* plus a more specific one, I assigned the more specific interaction; if the two predicted interactions did not include *interact with* but two more specific interactions, I did not assign an interaction, since I wouldn't know how to choose between them. Similarly, I assigned no interaction if there were more than two predicted interactions or no keywords present in the sentence. Case "KeyB" is the "Key" method with back-off: when no interaction was predicted, I assigned to the sentence the most frequent interaction in the training data. As before, I calculated the accuracy when I force all the sentences from one triple to be assign the same interaction, the most frequent interaction among those predicted for the individual sentences.

KeyB is more accurate than Key and although the KeyB accuracies are higher than the other baselines, they are significantly lower than those obtained with the graphical models and the neural net. The low accuracies of the trigger-word based

	Mj	**Mj***	**Cf**	**Cf***	**Dec**
	All (papers + citances)				
DM	60.5 =	59.7 =	60.5+	59.7 =	0.3%
pn	60.5	59.7	59.7	59.7	
NB	59.7 +	59.7 +	59.7	62.1 +	1.4%
pn	58.1	58.9	61.3	59.7	
NN	51.6	50.8			-19.1%
pn	63.7	62.9			
	Papers				
DM	44.4	42.2	40.0	42.2	-21.2%
pn	57.8	46.7	55.6	55.6	
NB	46.7	44.4	51.1	51.1	-14.5%
pn	57.8	57.8	53.3	57.8	
NN	44.4 =	44.4 =			0%
pn	44.4	44.4			
	Citances				
DM	52.3	53.4	53.4	53.4	-2.5%
pn	53.4	54.5	54.5	55.7	
NB	53.4	54.5	53.4	52.3	-3.1%
pn	55.7	55.7	54.5	54.5	
NN	50.0	52.3			-6.2%
pn	55.8	53.4			

Table 4.5: Accuracies of the classification of the 10 protein-protein interactions of Table 4.2 with the *protein names removed*. For each model, the second lines show the corresponding results, taken from Table 4.3, when the protein names are given. In the last column, the difference in accuracy (in percentage) with respect to the original case, averaged over all evaluation methods. The "+" signs indicate the cases for which the accuracy improves removing the protein names, the "=" signs the cases whan the accuracy does not change. The results reported here are obtained using only the words as features. The baseline measures, Key and KeyB are the same as Table 4.3.

methods show that the relation classification task is nontrivial, in the sense that not all the sentences contain the most obvious word for the interactions, and suggests that the trigger word approach is insufficient.

4.5.3 Protein name tagger

The dynamic model of Figure 4.1 has the appealing property of simultaneously performing interaction recognition and protein name tagging (also known as role extraction): the task consists of identifying all the proteins present in the sentence, given a sequence of words. I assessed a slightly different task: the identification of all the proteins present in the sentence *that are involved in the interaction*. For instance, in the following sentence: *"These results suggest that Tat- induced phosphorylation of serine 5 by CDK9 might be important after transcription has reached the +36 position, at which time CDK7 has been released from the complex."* there are three proteins (*Tat*, *CDK9* and *CDK7*) but the proteins involved in the interaction that I want to extract are only *Tat* and *CDK9*. Role extraction is a difficult task in general, made here more difficult for the reason above: *CDK7* can be (and in fact is) involved in an interaction in another sentence (*"Tat might regulate the phosphorylation of the RNA polymerase II carboxyl - terminal domain in pre - initiation complexes by activating CDK7"*).

I perform inference with the junction tree algorithm. The F-measure[10] achieved by this model for this task is 0.79 for "all," 0.67 for "papers" and 0.79 for "citances" (see Table 4.6); again, the model parameters were chosen with cross validation on the training test, and "citances" had superior performance. Note that I did not use a dictionary: the system learned to recognize the protein names using only the training data. Moreover, my role evaluation is quite strict: every token is assessed and I do not assign partial credit for constituents for which only some of the words are correctly

[10]The F-measure is a weighted combination of precision and recall. Here, precision and recall are given equal weight, that is, F-measure $= (2 * PRE * REC)/(PRE + REC)$.

	Recall	Precision	F-measure
All	0.74	0.85	0.79
Papers	0.56	0.83	0.67
Citances	0.75	0.84	0.79

Table 4.6: F-measures for the dynamic model DM of Figure 4.1 for the task of identifying the proteins involved in the interactions. (Only words as features.)

labeled. I did not use the information that all the sentences extracted from one triple contain the same proteins.

Given these promising results (both F-measure and classification accuracies), I believe that the dynamic model of Figure 4.1 is a good model for performing both name tagging and interaction classification simultaneously, or either of these task alone.

4.6 Sentence-level evaluation

In addition to assigning interactions to protein pairs, I am interested in sentence-level semantics, that is, in determining the interactions that are actually expressed in the sentence. To test whether the information assigned to the entire document by the HIV-1 database record can be used to infer information at the sentence level, an annotator with biological expertise hand-annotated the sentences from the experiments. The annotator was instructed to assign to each sentence one of the interactions of Table 4.2, "not interacting," or "other" (if the interaction between the two proteins was not one of Table 4.2).

Of the 2114 sentences that were hand-labeled, 68.3% of them disagreed with the HIV-1 database label, 28.4% agreed with the database label, and 3.3% were found to contain multiple interactions between the proteins. Among the 68.3% of the sentences for which the labels did not agree, 17.4% had the vague *interact with* relation, 7.4%

	Annotator											
DB	***D***	***Sy***	***St***	***B***	***Ina***	***R***	***Up***	***Inh***	***Su***	***IW***	***Ot***	***NO***
Degrad	44	0	2	5	6	5	2	0	23	9	11	6
Synerg	0	78	3	14	0	13	8	0	0	26	31	11
Stimul	0	5	23	12	0	8	7	5	1	26	60	18
Binds	0	6	9	118	0	25	8	10	1	129	77	22
Inacti	0	0	4	25	0	2	4	33	6	14	27	11
Requir	0	5	29	20	0	63	8	54	0	85	80	33
Upregu	0	4	24	0	0	0	124	2	0	21	32	4
Inhibi	0	8	4	8	2	2	2	43	9	24	37	19
Suppre	3	0	0	1	5	0	0	42	34	33	24	4
Intera	0	1	5	28	1	12	6	1	1	49	27	28
Acc. (%)	93.6	72.9	22.3	51.1	0	48.5	73.4	22.7	45.3	11.8		

Table 4.7: Confusion matrix between the hand-assigned interactions and the interactions that we obtain from the HIV-1 database. *Ot* is "other" (if the interaction between the two proteins in the sentence was not one of Table 4.2), *NO* is "not interacting." In the last row Acc. is the accuracy of using the database to label the individual sentences (assuming the annotator's labels to be the true labels).

did not contain any interaction and 43.5% had an interaction different from that specified by the triple.

In Table 4.7 we report the mismatch between the two sets of labels. The total agreement of 38.9%[11] provides a useful baseline of using a database for the labeling at the sentence level. It may be the case that certain interactions tend to be biologically related and thus tend to co-occur (*upregulate* and *stimulate* or *inactivate* and *inhibit*, for example).

I investigated a few of the cases in which the labels were "suspiciously" different, for example a case in which the database interaction was *stimulate* but the annotator found the same proteins to be related by *inhibit* as well. It turned out that the authors of the article found little evidence for this interaction (and suggested instead *inhibit*), suggesting an error in the database. In another case the database interaction was

[11]The accuracy without including the vague *interact with* is 49.4%.

require but the authors of the article, while supporting this, found that under certain conditions (when a protein is too abundant) the interaction changes to one of *inhibit*. In another complex case, an *inhibition* is caused by a concurrent *upregulation*. It is interesting that I was able to find controversial facts about protein interactions just by looking at the confusion matrix of Table 4.7.

For 72% of the triples, at least one of the sentences extracted from the target paper were found by the annotator to contain the interaction given by the database. I read four of the papers for which none of the sentences extracted were found to contain the interaction given by the database and did find sentences describing that interaction, but my system had failed to extract them.

I also trained the systems using the hand-labeled sentences. The goal in this case is to determine the interaction expressed *for each sentence*. This is a difficult task, for some sentences it took the annotator several minutes to understand them and decide which interaction applied: the language is difficult, and the task presupposes a lot of background knowledge in the protein and protein-interaction domains.

Table 4.8 shows the results on running the classification models on the six interactions for which I had more that 40 examples in the training sets. Again, the sentences from "papers" are especially difficult to classify; the best result for "paper" is 36.7% accuracy versus 63.2% accuracy for "citances." In this case the difference in performance of "papers" and "citances" is bigger than for the previous task of document classification. In Table 4.9 the confusion matrix for the case "citances," using the Naive Bayes model.

4.7 Conclusions

I tackled an important and difficult task: the classification of different interaction types between proteins in text. A solution to this problem would have an impact on

All	
DM	48.9
NB	47.1
NN	**52.9**
Bas. (Mf)	36.3
Key	30.5
KeyB	46.2
Papers	
DM	28.9
NB	33.3
NN	**36.7**
Bas. (Mf)	34.4
Key	18.9
KeyB	36.6
Citances	
DM	47.9
NB	53.4
NN	**63.2**
Bas. (Mf)	37.6
Key	38.3
KeyB	52.6

Table 4.8: Classification accuracies when the systems were trained and tested on the *hand labeled* sentences. The task was to predict the interaction *for each sentence*; the evaluation was done on a sentence-by-sentence basis. Best results shown in boldface. Here I considered only the six interactions for which I could find more that 40 examples for the training sets (see Table 4.9). Only words as features. Bas. (Mf) is the accuracy for choosing the most frequent interaction. Chance is 16.7.

a variety of important challenges in modern biology. The graphical models I implemented can simultaneously perform protein name tagging and relation identification, achieving high accuracy on both problems. I also found evidence supporting the hypothesis that citation sentences are a good source of training data, most likely because they provide a concise and precise way of summarizing facts in the bioscience literature.

	Prediction						**Acc.**
Truth	***Intera***	***Upreg***	***NO***	***Inhib***	***Stimu***	***Binds***	**(%)**
Intera	25	5	5	6	1	8	50.0
Upregu	2	11	0	1	0	1	73.3
NO	1	0	6	1	1	0	66.7
Inhibi	0	1	1	17	0	1	85.0
Stimul	2	3	1	3	0	1	0.0
Binds	7	1	5	4	0	12	41.4

Table 4.9: Confusion matrix for the Naive Bayes NB for "citances," with the system trained with the *hand labeled* sentences. The overall accuracy is 53.4%. *NO* stands for "no interaction."

Chapter 5

Conclusions

5.1 Contributions of this thesis

This thesis described the contribution of my work in the field of computational semantics. In particular, I described three projects that extract entities and relations from bioscience text.

The first project tackled the problem of assigning semantic relations to noun compounds and in Chapter 2, I described two approaches to this problem. For the first approach (Section 2.6), I identified several semantic relations for a collection of NCs extracted from medical journals and proposed a classification algorithm for the automatic classification of the relations. In this task of multi-class classification (with 18 classes) I achieved an accuracy of about 62%. These results can be compared with Vanderwende (1994) who reports an accuracy of 52% with 13 classes and Lapata (2000) whose algorithm achieves about 80% accuracy for a much simpler binary classification. The second approach described in Section 2.7 was linguistically motivated; I showed that mere membership within a particular sub-branch of a domain specific lexical hierarchy is sufficient in many cases for assignment of the appropriate

semantic relation, obtaining a classification accuracy of approximately 90%. I also showed how most of the related work relies on hand-written rules and/or addresses the easier task of classification with much fewer classes.

In Chapter 3, I described my work for the tasks of role and relation extraction, proposing several machine learning techniques that were shown to achieve good results for these difficult tasks: for the task of role extraction I achieved an F-measure of 70% and for the task of relation recognition an accuracy of 80%. I addressed the problem (rarely tackled in the related work) of distinguish between *different* relations that can occur between the *same* semantic entities. Most of the related work on relationship extraction assumes the entity extraction task performed by another system and the entities of interests therefore are given as input. My models do not make this assumption and perform role and relation extraction simultaneously. When the entities are given, the models proposed achieved around 97% accuracy for the task of distinguishing between eight semantic relations.

Finally, Chapter 4 tackled the identification of protein interactions; the accuracy achieved was 64% for a ten-class distinction. This work also introduced the use of an existing curated database for gathering labeled data and the use of citations. This work represents a significant improvement over the related work: some approaches simply report that a relation exists between two proteins but do not determine which relation holds, and most others use "trigger words"and/or hand-built templates.

The contribution of this thesis is to have applied several rigorous methods to these difficult, real-life problems. Some of the machine learning methods (the graphical models of Section 3.7) were designed and developed by me especially for these problems. The results obtained were quite encouraging.

5.2 Directions for future research

Many open questions and research directions remain, of course.

The relations and entities extracted by the algorithms described in this thesis are intended to be combined to produce larger propositions that can then be used in a variety of interpretation paradigms, such as abductive reasoning or inductive logic programming.

Other important issues relevant to the task of extracting information from text, not tackled in this thesis are the following: the access to huge amount of textual data, the connection between several databases and/or text collections for linking different pieces of information and the related problem of a system architecture to support multiple layers of annotation on text, the development of effective interface, and the challenge of finding a good knowledge representation and the right inference procedures.

The hope is to have one day an unifying theory that explains the process of understanding (and generating) meaning in linguistic utterances. We do not know whether we need to understand how humans process language to be able do so computationally, but we are still very far from this.

Meanwhile, in this thesis, I tackle some practical but difficult and important problems, I propose methods to address them achieving results good enough to be useful for a actual implementation of the systems. My hope is that this can be considered progress, too.

Bibliography

Agichtein, E. and Gravano, L. (2000). Snowball: Extracting relations from large plain-text collections. In *Proceedings of the Fifth ACM International Conference on Digital Libraries*.

Ahlberg, C. and Shneiderman, B. (1994). Visual information seeking: Tight coupling of dynamic query filters with starfield displays. In *Proceedings of ACM CHI'94*, pages 313–317.

Ahmed, S., Chidambaram, D., Davulcu, H., and Baral, C. (2005). Intex: A syntactic role driven protein-protein interaction extractor for bio-medical text. In *Proceedings ISMB/ACL Biolink 2005*.

Appelt, D., Hobbs, J., Bear, J., Israel, D., Kameyama, M., Kehler, A., Martin, D., Meyers, K., and Tyson, M. (1993). Sri international fastus system: Muc-6 test results and analysis.

Aseltine, J. (1999). Wave: An incremental algorithm for information extraction. In *Proceedings of the AAAI 1999 Workshop on Machine Learning for Information Extraction*.

Baker, C. F., Fillmore, C. J., and Lowe, J. B. (1998). The Berkeley FrameNet project. In Boitet, C. and Whitelock, P., editors, *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*, pages 86–90, San Francisco, California. Morgan Kaufmann Publishers.

Barker, K. and Szpakowicz, S. (1998). Semi-automatic recognition of noun modifier relationships. In *Proceedings of COLING-ACL '98*, Montreal, Canada.

Barrett, L., Davis, A. R., and Dorr, B. J. (2001). Interpreting noun-noun compounds using wordnet. In *Proceedings of 2001 CICLing Conference*, Mexico City.

Baum, L. E. (1972). An inequality and associated maximisation techniques in statistical estimation of probabilistic functions of markov processes. In *Inequalities*, pages 3:1–8.

Berger, A. L., Della Pietra, S. A., and Della Pietra, V. J. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Bikel, D. M., Schwartz, R. L., and Weischedel, R. M. (1999). An algorithm that learns what's in a name. *Machine Learning*, 34(1-3):211–231.

Blaschke, C., Andrade, M., Ouzounis, C., and Valencia, A. (1999a). Automatic extraction of biological information from scientific text: Protein-protein interactions. *Proceedings of ISMB*.

Blaschke, C., Andrade, M. A., Ouzounis, C., and Valencia, A. (1999b). Automatic extraction of biological information from scientific text: Protein-protein interactions. pages 60–67.

Blaschke, C. and Valencia, A. (2002). The frame-based module of the suiseki information extraction system. *IEEE Intelligent Systems*, 17(2).

Blei, D. M., Bagnell, J. A., and McCallum, A. K. (2002). Learning with scope, with application to information extraction and classification. In *Uncertainty in Artificial Intelligence*.

Blum, A. and Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*.

Borthwick, A., Sterling, J., Agichtein, E., and Grishman, R. (1998). Exploiting diverse knowledge sources via maximum entropy in named entity recognition. In *Proceedings of the Sixth Workshop on Very Large Corpora*. Association for Computational Linguistics.

Brill, E. (1995). Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–565.

Brill, E. and Resnik, P. (1994). A rule-based approach to prepositional phrase attachment disamibuation. In *Proceedings of COLING-94*.

Budanitsky, A. and Hirst, G. (2001). Semantic distance in wordnet: an experimental, application-oriented evaluation of five measures. In *Proceedings of the NAACL 2001 Workshop on WordNet and Other Lexical Resources*, Pittsburgh, PA.

Buitelaar, P. (1997). A lexicon for underspecified semantic tagging. In *Proceedings of ANLP 97, SIGLEX Workshop*, Washington DC.

Bunescu, R., Ge, R., Kate, R., Marcotte, E., Mooney, R. J., Ramani, A. K., and Wong, Y. W. (2005). Comparative experiments on learning information extractors for protiens and their interactions. *Artificial Intelligence in Medicine*, 33(2).

Charniak, E. (1995). Parsing with context-free grammars and word statistics. Technical Report CS-95-28.

Chelba, C. and Mahajan, M. (2001). Information extraction using the structured language model. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*. ACL.

Collier, N., No, C., and Tsujii, J. (2000). Extracting the names of genes and gene products with a hidden markov model. In *Proc. COLING 2000*, pages 201–207.

Collins, M. (1997). Three generative, lexicalized models for statistical parsing. In Cohen, P. R. and Wahlster, W., editors, *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 16–23, Somerset, New Jersey. Association for Computational Linguistics.

Collins, M. (2002). Ranking algorithms for named-entity extraction: Boosting and the voted perceptron. In *Proceedings of the 40th Annual Conference of the Association for Computational Linguistics*. ACL.

Collins, M. and Miller, S. (1997). Semantic tagging using a probabilistic context free grammar. In *Proceedings of 6th Workshop on Very Large Corpora*.

Collins, M. and Singer, Y. (1999). Unsupervised models for named entity classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.

Collins, M. J. (1996). A new statistical parser based on bigram lexical dependencies. In Joshi, A. and Palmer, M., editors, *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics*, pages 184–191, San Francisco. Morgan Kaufmann Publishers.

Cooper, J. W. and Kershenbaum, A. (2005). Discovery of protein-protein interactions using a combination of linguistic, statistical and graphical information. *BMC Bioinformatics*, 6.

Corney, D., Buxton, B., Langdon, W., and Jones, D. (2004). Biorat: extracting biological information from full-length papers. *Bioinformatics*, 20(17).

Craven, M. (1999). Learning to extract relations from Medline. In *AAAI-99 Workshop*.

Craven, M. and Kumlien, J. (1999). Constructing biological knowledge-bases by extracting information from text sources. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*, pages 77–86, Germany.

Culotta, A. and Sorensen, J. (2004). Dependency tree kernels for relation extraction. In *Proceedings of the 42th Annual Conference of the Association for Computational Linguistics*, pages 423–429. ACL.

Ding, J., Berleant, D., Nettleton, D., and Wurtele, E. (2000). Mining medline: Abstracts, sentences, or phrases? In *Proceedings of the Pacific Symposium on Biocomputing (PSB)*.

Dowing, P. A. (1977). On the creation and use of english compound nouns. In *Language*, pages 53:810–842.

Feldman, R., Regev, Y., Finkelstein-Landau, M., Hurvitz, E., and Kogan, B. (2002). Mining biomedical literature using information extraction. *Current Drug Discovery*.

Fellbaum, C., editor (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Fillmore, C. J. (1968). The case for case. *Universals in Linguistic Theory*, pages 1–88.

Fillmore, C. J. (1977). The case for case revisited. *Syntax and Semantics*, 8:59–81.

Finin, T. W. (1980). *The Semantic Interpretation of Compound Nominals*. Ph.d. dissertation, University of Illinois, Urbana, Illinois.

Freitag, D. (2004). Trained named entity recognition using distributional clusters. In *Proceedings of the EMNLP*.

Freitag, D. and McCallum, A. (1999). Information extraction with HMMs and shrinkage. In *Proceedings of the AAAI-99 Workshop on Machine Learning for Informatino Extraction*.

Freitag, D. and McCallum, A. (2000). Information extraction with HMM structures learned by stochastic optimization. In *AAAI/IAAI*, pages 584–589.

Friedman, C., Kra, P., Yu, H., Krauthammer, M., and Rzhetsky, A. (2001). Genies: a natural-language processing system for the extraction of molecular pathways from journal articles. In *Bioinformatics, vol. 17*. Oxford Univ. Press.

Gildea, D. and Jurafsky, D. (2000). Automatic labeling of semantic roles. In *Proceedings of ACL 2000*, Hong Kong.

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

Gildea, D. and Palmer, M. (2002). The necessity of syntactic parsing for predicate argument recognition. In *Proceedings of the 40th Annual Conference of the Association for Computational Linguistics*. ACL.

Grishman, R. (1986). *Computational Linguistics*. Cambridge University Press, Cambridge.

Hasegawa, T., Sekine, S., and Grishman, R. (2004). Discovering relations among named entities from large corpora. In *Proceedings of the 42th Annual Conference of the Association for Computational Linguistics*, pages 415–422. ACL.

Humphreys, L., Lindberg, D., Schoolman, H., and Barnett, G. O. (1998). The unified medical language system: An informatics research collaboration. *Journal of the American Medical Informatics Assocation*, 5(1):1–13.

Jones, R., McCallum, A., Nigam, K., and Riloff, E. (1999). Bootstrapping for text learning tasks. In *IJCAI-99 Workshop on Text Mining: Foundations, Techniques and Applications*.

Jordan, M. I. (2004). Graphical models. *Statistical Science (Special Issue on Bayesian Statistics)*, 19:140–155.

Kingsbury, P. and Palmer, M. (2002). From treebank to propbank. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*.

Klein, D. and Manning, C. D. (2002). Conditional structure versus conditional estimation in NLP models. In *EMNLP*.

Klein, D., Smarr, J., Nguyen, H., and Manning, C. (2003). Named entity recognition with character-level models. In *Proceedings of CoNLL'03*.

Lafferty, J., McCallum, A., and Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA.

Lapata, M. (2000). The automatic interpretation of nominalizations. In *Proceedings of AAAI*.

Lauer, M. (1995a). Corpus statistics meet the compound noun. In *Proceedings of the 33rd Meeting of the Association for Computational Linguistics*.

Lauer, M. (1995b). *Designing Statistical Language Learners: Experiments on Noun Compounds*. PhD thesis, Sydney.

Lauer, M. and Dras, M. (1994). A probabilistic model of compound nouns. In *Proceedings of the 7th Australian Joint Conference on AI*.

Leonard, R. (1984). *The Interpretation of English Noun Sequences on the Computer*. North-Holland, Amsterdam.

Levi, J. (1978). *The Syntax and Semantics of Complex Nominals*. Academic Press, New York.

Li, H. and Abe, N. (1998). Generalizing case frames using a thesaurus and the MDI principle. *Computational Linguistics*, 24(2):217–244.

Liberman, M. Y. and Church, K. W. (1992). Text analysis and word pronunciation in text-to-speech synthesis. In Furui, S. and Sondhi, M. M., editors, *Advances in Speech Signal Processing*, pages 791–831. Marcel Dekker, Inc.

Marcotte, E., Xenarios, I., and Eisenberg, D. (2001). Mining literature for protein-protein interactions. *Bioinformatics*, 17(4).

McCallum, A., Freitag, D., and Pereira, F. (2000). Maximum entropy markov models for information extraction and segmentation. In *Proc. 17th International Conf. on Machine Learning*, pages 591–598. Morgan Kaufmann, San Francisco, CA.

McCallum, A. and Li, W. (2003). Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL)*.

McDonald, R. and Pereira, F. (2005). Identifying gene and protein mentions in text using conditional random fields. *BMC Bioinformatics*, 6(1).

Miller, S., Fox, H., Ramshaw, L., and Weischedel, R. (2000). A novel use of statistical parsing to extract information from text. In *Proceedings of the NAACL 2000*, pages 226–233.

Nakov, P., Schwartz, A., and Hearst, M. (2004). Citances: Citation sentences for semantic analysis of bioscience text. In *Proceedings of the SIGIR'04 workshop on Search and Discovery in Bioinformatics*.

Ng, A. and Jordan, M. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and Naive Bayes. *NIPS 14*.

Ng, S. and Wong, M. (1999). Toward routine automatic pathway discovery from on-line scientific text abstracts. In *Genome Informatics, 10:104–112*.

Ono, T., Hishigaki, H., Tanigami, A., and Takagi, T. (2001). Automated extraction of information on protein-protein interactions from the biological literature. *Bioinformatics*, 17(1).

Palakal, M., Stephens, M., Mukhopadhyay, S., Raje, R., and Rhodes, S. (2002). Multi-level text mining method to extract biological relationships. In *Proceedings of the 2002 IEEE Computer Society Bioinformatics Conference*, pages 60–67.

Phuong, T., Lee, D., and Lee, K.-H. (2003). Learning rules to extract protein interactions from biomedical text. In *PAKDD*.

Pustejovsky, J., editor (1995). *The Generative Lexicon*. MIT Press.

Pustejovsky, J., Bergler, S., and Anick, P. (1993). Lexical semantic techniques for corpus analysis. *Computational Linguistics*, 19(2).

Pustejovsky, J., Castano, J., and Zhang, J. (2002). Robust relational parsing over biomedical literature: Extracting inhibit relations. In *PSB 2002*, pages 362–373.

Rabiner, L. R. and Juang, B. H. (1986). An introduction to hidden markov models. In *IEEE ASSP Magazine*.

Ramani, C., Marcotte, E., Bunescu, R., and Mooney, R. (2005). Using biomedical literature mining to consolidate the set of known human protein-protein interactions. In *Proceedings ISMB/ACL Biolink 2005*.

Ray, S. and Craven, M. (2001). Representing sentence structure in Hidden Markov Models for Information Extraction. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI-2001)*.

Resnik, P. (1993). *Selection and Information: A Class-Based Approach to Lexical Relationships*. PhD thesis, University of Pennsylvania. (Institute for Research in Cognitive Science report IRCS-93-42).

Resnik, P. (1995). Disambiguating noun groupings with respect to WordNet senses. In *Third Workshop on Very Large Corpora*. Association for Computational Linguistics.

Resnik, P. and Hearst, M. A. (1993). Structural ambiguity and conceptual relations. In *Proceedings of the ACL Workshop on Very Large Corpora*, Columbus, OH.

Riloff, E. (1993). Automatically constructing a dictionary for information extraction tasks. In *National Conference on Artificial Intelligence*, pages 811–816.

Riloff, E. (1996). Automatically generating extraction patterns from untagged text. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 1044–1049. AAAI Press/The MIT Press.

Riloff, E. (1998). The sundance sentence analyzer. In *http://www.cs.utah.edu/projects/nlp/*.

Rindfleisch, T. C., Hunter, L., and Aronson, A. R. (1999). Mining molecular binding terminology from biomedical text. In *Proceedings of the 1999 AMIA Annual Fall Symposium*, pages 127–136.

Rindflesch, T., Hunter, L., and Aronson, L. (1999). Mining molecular binding terminology from biomedical text. *Proceedings of the AMIA Symposium*.

Rindflesch, T., Tanabe, L., Weinstein, J., and Hunter, L. (2000a). Edgar: extraction of drugs, genes and relations from the biomedical literature. *BMC Bioinformatics*, 5:514–525.

Rindflesch, T., Tanabe, L., Weinstein, J. N., and Hunter, L. (2000b). Extraction of drugs, genes and relations from the biomedical literature. *Pacific Symposium on Biocomputing*, 5(5).

Rosario, B. and Hearst, M. (2004). Classifying semantic relations in bioscience texts. In *Proceedings of the 42th Annual Conference of the Association for Computational Linguistics*. ACL.

Rosario, B., Hearst, M., and Fillmore, C. (2002). The descent of hierarchy, and selection in relational semantics. In *Proceedings of ACL-02*.

Rosario, B. and Hearst, M. A. (2001). Classifying the semantic relations in noun compounds via a domain-specific lexical hierarchy. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*. ACL.

Saric, J., Jensen, L. J., Bork, P., Ouzounova, R., and Rojas, I. (2004). Extracting regulatory gene expression networks from pubmed. In *Proceedings of the 42th Annual Conference of the Association for Computational Linguistics*, pages 191–198. ACL.

Sekimizu, T., Park, H., and Tsujii, J. (1998). Identifying the interaction between genes and gene products based on frequently seen verbs in medline abstracts. *Genome Informatics Ser Workshop Genome Inform.*

Seymore, K., McCallum, A., and Rosenfeld, R. (1999). Learning hidden Markov model structure for information extraction. In *AAAI 99 Workshop on Machine Learning for Information Extraction*.

Soderland, S., Fisher, D., Aseltine, J., and Lehnert, W. (1995). CRYSTAL: Inducing a conceptual dictionary. In Mellish, C., editor, *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 1314–1319, San Francisco. Morgan Kaufmann.

Spiegelhalter, D. J., A.Thomas, and Best, N. G. (1996). Computation on bayesian graphical models. *Bayesian Statistics*, 5:407–425.

Stapley, B. J. and Benoit, G. (2000). Biobibliometrics: Information retrieval and visualization from co-occurrences of gene names in medline abstracts. In *Proceedings of the Pacific Symposium of Biocomputing, 5*, pages 529–540.

Stephens, M., Palakal, M., Mukhopadhyay, S., and Raje, R. (2001). Detecting gene relations from medline abstracts.

Swier, R. S. and Stevenson, S. (2004). Unsupervised semantic role labeling. In *Proceedings of the EMNLP*.

Thomas, J., Milward, D., Ouzounis, C., and Pulman, S. (2000). Automatic extraction of protein interactions from scientific abstracts. *Proceedings of the Pac Symp Biocomput*.

Thompson, C., Levy, R., and Manning, C. (2003). A generative model for semantic role labeling. *Proceedings of EMCL '03*.

Vanderwende, L. (1994). Algorithm for automatic interpretation of noun sequences. In *Proceedings of COLING-94*, pages 782–788.

Vapnik, V. (1998). *Statistical Learning Theory*. Oxford University Press.

Warren, B. (1978). *Semantic Patterns of Noun-Noun Compounds*. Actr Universitatis Gothoburgensis, Gothenburg.

Xue, N. and Palmer, M. (2004). Calibrating features for semantic role labeling. In *Proceedings of the EMNLP*.

Yangarber, R., Grishman, R., Tapanainen, P., and Huttunen, S. (2000). Unsupervised discovery of scenario-level patterns for information extraction. In *Proceedings of the Sixth Conference on Applied Natural Language Processing, (ANLP-NAACL 2000)*, pages 282–289.

Zelenko, D., Aone, C., and Richardella, A. (2002). Kernel methods for relation extraction. In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP 2002)*.

Zhai, C. and Lafferty, J. (2001). A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR '01*.

Zhou, G. and Su, J. (2002). Named entity recognition using an HMM-based chunk tagger.