THE UNIVERSITY OF
SYDNEY

COPYRIGHT AND USE OF THIS THESIS

This thesis must be used in accordance with the provisions of the Copyright Act 1968.

Reproduction of material protected by copyright may be an infringement of copyright and copyright owners may be entitled to take legal action against persons who infringe their copyright.

Section 51 (2) of the Copyright Act permits an authorized officer of a university library or archives to provide a copy (by communication or otherwise) of an unpublished thesis kept in the library or archives, to a person who satisfies the authorized officer that he or she requires the reproduction for the purposes of research or study.

The Copyright Act grants the creator of a work a number of moral rights, specifically the right of attribution, the right against false attribution and the right of integrity.

You may infringe the author's moral rights if you:

- fail to acknowledge the author of this thesis if you quote sections from the work
- attribute this thesis to another author
- subject this thesis to derogatory treatment which may prejudice the author's reputation

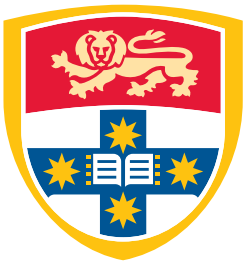
For further information contact the University's Director of Copyright Services

sydney.edu.au/copyright

Structured Named Entities

Nicky Ringland

Supervisor: James Curran



THE UNIVERSITY OF
SYDNEY

A thesis submitted

in fulfilment of the requirements

for the degree of Doctor of Philosophy

in the School of Information Technologies at

The University of Sydney

School of Information Technologies

2016

Abstract

The names of people, locations, and organisations play a central role in language, and named entity recognition (NER) has been widely studied, and successfully incorporated, into natural language processing (NLP) applications. The most common variant of NER involves identifying and classifying proper noun mentions of these and miscellaneous entities as linear spans in text.

Unfortunately, this version of NER is no closer to a detailed treatment of named entities than chunking is to a full syntactic analysis. NER, so construed, reflects neither the syntactic nor semantic structure of NE mentions, and provides insufficient categorical distinctions to represent that structure.

Representing this nested structure, where a mention may contain mention(s) of other entities, is critical for applications such as coreference resolution. The lack of this structure creates spurious ambiguity in the linear approximation.

Research in NER has been shaped by the size and detail of the available annotated corpora. The existing structured named entity corpora are either small, in specialist domains, or in languages other than English.

This thesis presents our *Nested Named Entity* (NNE) corpus of named entities and numerical and temporal expressions, taken from the WSJ portion of the Penn Treebank (PTB, Marcus et al., 1993). We use the BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein, 2005a) as our basis, manually annotating it with a principled, fine-grained, nested annotation scheme and detailed annotation guidelines. The corpus comprises over 279,000 entities over 49,211 sentences (1,173,000 words), including 118,495 top-level entities.

Our annotations were designed using twelve high-level principles that guided the development of the annotation scheme and difficult decisions for annotators. We also monitored the semantic grammar that was being induced during annotation, seeking to identify and reinforce common patterns to maintain consistent, parsimonious annotations.

The result is a scheme of 118 hierarchical fine-grained entity types and nesting rules, covering all capitalised mentions of entities, and numerical and temporal expressions. Unlike many corpora, we have developed detailed guidelines, including extensive discussion of the edge cases, in an ongoing dialogue with our annotators which is critical for consistency and reproducibility.

We annotated independently from the PTB bracketing, allowing annotators to choose spans which were inconsistent with the PTB conventions and errors, and only refer back to it to resolve genuine ambiguity consistently.

We merged our NNE with the PTB, requiring some systematic and one-off changes to both annotations. This allows the NNE corpus to complement other PTB resources, such as PropBank, and inform PTB-derived corpora for other formalisms, such as CCG and HPSG. We compare this corpus against BBN.

We consider several approaches to integrating the PTB and NNE annotations, which affect the sparsity of grammar rules and visibility of syntactic and NE structure. We explore their impact on parsing the NNE and merged variants using the Berkeley parser (Petrov et al., 2006), which performs surprisingly well without specialised NER features.

We experiment with flattening the NNE annotations into linear NER variants with stacked categories, and explore the ability of a maximum entropy and a CRF NER system to reproduce them. The CRF performs substantially better, but is infeasible to train on the enormous stacked category sets. The flattened output of the Berkeley parser are almost competitive with the CRF.

Our results demonstrate that the NNE corpus is feasible for statistical models to reproduce. We invite researchers to explore new, richer models of (joint) parsing and NER on this complex and challenging task.

Our nested named entity corpus will improve a wide range of NLP tasks, such as coreference resolution and question answering, allowing automated systems to understand and exploit the true structure of named entities.

Acknowledgements

The completion of this thesis has been a long journey, and I am immensely glad to have had wonderful travelling companions. Foremost thanks must go to James Curran, without whose encouragement, support and guidance I would never have embarked on this journey. Thank you for introducing me to computer science, for nurturing (and occasionally rekindling) my love of computational linguistics, and for encouraging my passion for education.

Thank you to all members of Schwa Lab past and present who have helped me in so many ways. I have been incredibly fortunate to work with such a wonderful group of friends and colleagues. Thanks especially to Ben Hachey, whose extra guidance at the pointy end of things was particularly appreciated, Matt Honnibal, especially for giving me that first glimmer of hope that a linguistics student can add that extra 'computational' adjective, David Vadas, Tara McIntosh, Jonathan Kummerfeld, Stephen Merity, Tim O'Keefe, Daniel Tse, Will Radford, Dominick Ng, James Constable, Glen Pink, and to Tim Dawborn, without whom I would have neither servers to run my experiments on, nor an NER system to evaluate on. Katie Bell, thank you for helping me debug my very first Python program, and for being an inspiration henceforth. Joel Nothman, thank you for the many stimulating discussions, cups of tea, continued encouragement, and for embodying the epitome of a research student. My work is much the better thanks to the good example you have always set. Kellie Webster, thank you for being my thesis writing buddy, for helping me

find words when I was stuck, for keeping me accountable to my deadlines, for the many hours of annotation and proof-reading and for countless other ways your presence and efforts have improved the past few years.

My work rests on that of my annotators: Kellie Webster, Vivian Li, Joanne Yang and Kristy Hughes. Thank you for the many hours of discussion, and for battling through all 50,000 WSJ articles with me. I hope that, in time, you can start to enjoy reading newspaper articles again without too many flashbacks.

I gratefully acknowledge the funding received from the Tempe Mann Travelling Scholarship, which enabled me to take a mini-sabbatical to Edinburgh, where my research was greatly improved by the efforts of Bonnie Webber, Mark Steedman and the entire computational linguistics group. Thanks also to the research communities of Cambridge, especially Stephen Clark, and the University of Texas at Austin, especially Jason Baldridge. Thank you also to the many local academics who have advised, guided and influenced me along the way, especially Alan Fekete and Jon Patrick, and to the support, help navigating various bureaucratic hurdles, and friendship from all the SIT staff, especially Josie Spongberg and Evelyn Riegler.

To the women of SIT, especially Georgina Wilcox, Emma Fitzgerald, Shaghayegh Sharif Nabavi, Mahboobeh Moghaddam, and Tara Babaie, thank you for always being ready with tea and chocolate. To the National Computer Science School and Girls' Programming Network tutors, students and teachers, whose encouragement was always appreciated, even when phrased as the question: *Have you finished yet?*

The challenges of a PhD can only be overcome when balanced by incredible friends. Thank you all for helping in more ways than you are probably aware, from much-needed stress relief, to technical support, distractions, laughter and copious amounts of chocolate. Jonathan Usmar, thank you for encouraging me to actually give this computer science thing a try. Lachlan Howe and James

Bailey, thank you for the music and camaraderie. Sophia Di Marco, thank you for always being there, and for your unfaltering confidence in me.

Finally, thank you to my family for being constantly supportive, for raising me with a love of learning, and for being encouraging and patient even while not understanding my work.

And to Sam, perhaps the most patient of all, who has been by my side throughout every minute of this PhD. Thank you.

Statement of compliance

I certify that:

- I have read and understood the University of Sydney Student Plagiarism: Coursework Policy and Procedure;
- I understand that failure to comply with the Student Plagiarism: Coursework Policy and Procedure can lead to the University commencing proceedings against me for potential student misconduct under Chapter 8 of the University of Sydney By-Law 1999 (as amended);
- this Work is substantially my own, and to the extent that any part of this Work is not my own I have indicated that it is not my own by Acknowledging the Source of that part or those parts of the Work.

Name: *Nicky Ringland*

Signature: Date: *14th March 2016*

Contents

List of Figures

List of Tables

1 Introduction

> What's Montague? it is nor hand, nor foot,
> Nor arm, nor face, nor any other part,
> Belonging to a man. O! be some other name:
> ***What's in a name?***
>
> William Shakespeare

The goal of natural language processing (NLP) is to develop computational systems for the interpretation, storage and manipulation of natural language.

People, locations, organisations and other *named entities* play central roles in our lives, and their mentions are central in language. Therefore, named entity recognition (NER) has been a focus for NLP research, especially with statistical methods, and is a core component of many NLP applications, including search engines, question answering and machine translation.

The standard *named entity recognition* task involves identifying proper noun mentions of entities and classifying them according to a pre-defined category scheme. In most schemes, a contiguous sequence of tokens is identified and annotated with a single course-grained category. These *linear spans* are mutually exclusive — they cannot be nested within or overlap each other.

General domain NER includes identifying people (PER), locations (LOC), organisations (ORG) entities, and sometimes includes a catch-all category for miscellaneous (MISC) entities. This coarse-grained task has been extended by splitting these categories into finer-grained distinctions in an entity hierarchy.

It can also include numerical (NUMEX) and temporal (TIMEX) expressions, which may or may not be proper noun mentions, but are often distinguished by a small number of lexical patterns.

Unfortunately, this linear approximation of NER takes us no closer to a detailed semantic interpretation of named entities than chunking is to a full syntactic analysis. NER, so construed, reflects neither the syntactic nor semantic structure of NE mentions, and typically provides insufficient categorical distinctions to represent that structure.

Representing this nested structure, where a mention contains mention(s) of other entities, is critical for applications such as coreference resolution, and the lack of this structure creates spurious ambiguity in the linear annotations.

Research in NER, as with most NLP tasks, has been shaped by the quantity and quality of the available annotated corpora. As we discuss in Chapter 2, the existing structured named entity corpora are either small, in specialist domains, or in languages other than English. As such, they are unsuitable for our use, and we instead build on the unstructured BBN (Weischedel and Brunstein, 2005a) entity scheme and annotations, adding structural and finer-grained categories.

This thesis addresses this deficiency, presenting the *Nested Named Entity* (NNE) corpus, the first large-scale corpus of manually-annotated, structured, fine-grained named entities for English newswire, taken from the WSJ portion of the Penn Treebank (PTB, Marcus et al., 1993). We explore how well existing phrase structure parsers and NER systems perform on this complex nested named entity task, and demonstrate that it is feasible to learn.

1.1 Motivating nested entity structure

A named entity can contain mentions of other entities: [Twinings of [London]$_{\text{LOC}}$]$_{\text{CORP}}$ and [[Cambridge]$_{\text{LOC}}$ University]$_{\text{EDU}}$ both contain a nested mention of a CITY: London and Cambridge.[1] These nested mentions are extremely common.

In the linear approximation of NER, all tokens in the span boundaries are labelled with the category of this larger span, so London and Cambridge would be labelled as a corporation (CORP) and educational institution (EDU) respectively, when they both function semantically as a mention of a city.

This introduces a type of *spurious ambiguity* into linear NER. Even in the case where the single token Cambridge (without University) refers to the university, the analysis should similarly be [[Cambridge]$_{\text{CITY}}$]$_{\text{EDU}}$, since the metonymy is simply due to the elision of University.

The lack of corpora annotated with nested named entities has limited progress on the structured task, and the most influential resources that have directed research effort (e.g. the CoNLL shared task corpora) lack this information. This has resulted in the vast majority of general domain NER research (especially in English) focused on annotations without internal structure, meaning spurious ambiguity is forced onto the analysis of each token.

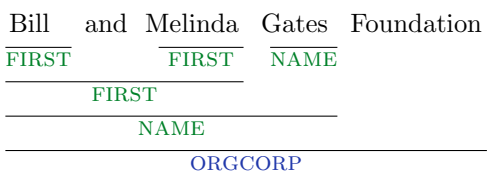
The lack of nested structure in entities also impacts downstream applications. Consider the task of question answering, for example In which city's stock exchange did Nike, Inc. first list? A system may identify that Nike listed on the New York Stock Exchange, but without knowledge of how to unpack that entity (e.g. [[New York]$_{\text{CITY}}$ Stock Exchange]$_{\text{CORP}}$) and identify the CITY nested within it, further processing would be required.

[1] In this thesis, we annotate examples with square brackets and subscripts, and abbreviate unambiguous categories, e.g. we often drop the ORG prefix on ORG subtypes. We use colour to differentiate coarse-grained entity types (PER, ORG, LOC, FACILITY, MISC, NUMEX, TIMEX).

Similarly, in question answering information must often be combined from several sources to answer a question. How many days are there between Christmas in 2014 and Easter in 2015? Finding the dates of those two events may be straightforward, but finding a sentence that has already done that calculation is less likely. In this case, we must interpret the two dates as points on a timeline and calculate the difference, which requires an understanding of the internal structure of temporal expressions.

Understanding the internal structure of numerical expressions is also critical. For example, one to two hundred people is a single numerical reference (100-200 people), while rooms can accommodate bookings of one hundred and sixty people respectively refers to two different numbers (not 160).

Learning the structure of an entity also gives us more evidence for tasks such as coreference resolution and relation extraction, for example, understanding that people's names are constructed of first names (FIRST, e.g. [Bill]$_{\text{FIRST}}$) and family names (NAME e.g. [Gates]$_{\text{NAME}}$) helps us identify that [Mr Gates]$_{\text{PER}}$, [Bill Gates]$_{\text{PER}}$ and [Bill]$_{\text{PER}}$ all refer to the same person, one of the founders of the [Bill and Melinda Gates Foundation]$_{\text{ORG}}$.

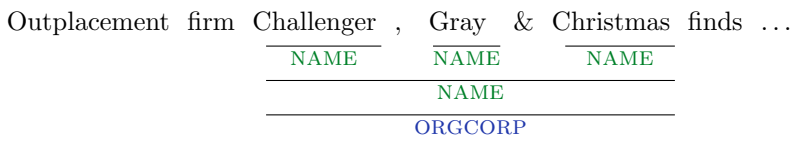


A better representation of the internal structure of a named entity can help in identifying its boundaries, a particularly difficult part of NER. This is especially true of entities containing prepositions (the [Bank of [England]$_{\text{COUNTRY}}$]$_{\text{CORP}}$ or conjunctions ([[[Proctor]$_{\text{NAME}}$ & [Gamble]$_{\text{NAME}}$]$_{\text{NAME}}$]$_{\text{CORP}}$).[2] Syntax can also offer further information for making NER decisions. Consider that I told Lucy Ester couldn't lose, does not refer to a [Lucy Ester]$_{\text{PER}}$. Similarly I gave Oprah Spiderbait

[2]The NAME nesting structure is explained in Chapter 3.

's new album, can use syntactic clues to separate those entities, and to add information that Oprah is likely to be an animate entity.

By better learning the structure of entities, and combining this knowledge with syntactic and semantic information from a wider context we can hope to correctly analyse even very difficult entities, such as this CORP from sentence WSJ1249_34 of the PTB:



1.2 Motivation for fine-grained entities

Most NER tasks involves the broad categories PER, LOC and ORG, with some tasks also including a miscellaneous category (MISC), and temporal (TIMEX) and numerical (NUMEX) expressions.

In many real-world applications such as relation extraction, named entity linking, and question answering, these very coarse-grained categories are insufficient and while NER is used in the NLP pipeline, further post-processing and classification into fine-grained categories is required.

Sekine et al. (2002) introduce a detailed category hierarchy, consisting of more than 200 categories. This category hierarchy has been used by Hashimoto et al. (2008) to annotate a Japanese corpus of 8,500 newspaper articles and 400 white papers, though it has not yet been used to annotate a large-scale English corpus. Finer category distinctions are more common in domain-specific NER. For example, the GENIA corpus in the biomedical domain contains 36 different categories with finer-grained distinctions (and shallow nested structures). In general domain, English corpora, the most widely used corpora: MUC-6 (Sund-

heim, 1995), MUC-7 (Chinchor, 1998), and CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003), are not annotated with fine-grained categories.

A fine-grained category hierarchy also allows for clearer signals in machine learning. Consider the MISC category, a particularly heterogeneous collection of entities, ranging from works of art to nationalities, products and laws. These are all quite different types of entities, and their surface forms show substantial variation. The names of laws or works of art often contain numerical or temporal references (e.g. [Proposition [13]$_{\text{CD}}$]$_{\text{LAW}}$, or [[1984]$_{\text{YEAR}}$]$_{\text{BOOK}}$), while the names of religions or nationalities rarely do. By distinguishing between these types, we can allow for a much clearer signal for each type of entity.

The clearer signal for fine-grained categories is most evident when combined with syntactic information, such as verb sub-categorisation restrictions, since fine-grained categories behave differently and are used in different syntactic structures in text. A person can attend an educational organisation, such as a school or university, but cannot attend a corporate organisation. Similarly, in coreference resolution, [Mr [Vinken]$_{\text{NAME}}$]$_{\text{PER}}$ is a much more likely reference than [Mr [Pierre]$_{\text{FIRST}}$]$_{\text{PER}}$. These distinctions are important in order to learn high-accuracy models of entity types.

While some English corpora do exist with fine-grained entity annotations, such as BBN (Weischedel and Brunstein, 2005a) which is annotated with 64 types of named entity, numerical and time expression, it has not been widely used, and the CoNLL 2003 data set and the four-category NER task (PER, LOC, ORG, MISC) remains the de-facto standard.

In this work, we build on the BBN (Weischedel and Brunstein, 2005a) entity scheme and annotations, adapting the scheme with the addition of structural and finer-grained categories including some from Sekine et al. (2002). Using the existing BBN annotations allows us to create our nested corpus much more

rapidly, both in terms of annotation speed and the ability to find instances of specific types for designing our schema.

1.3 Contributions

In this thesis, we present the Nested Named Entity (NNE) corpus, the first corpus of nested named entity structure in English newswire text. It comprises nearly 50,000 sentences of fine-grained, structured entity annotations over the Wall Street Journal portion of the Penn Treebank. We use the BBN Pronoun Coreference and Entity Type Corpus (Weischedel and Brunstein, 2005a) as the starting point for our highest layer of entity annotation, and add in nested structures using a hierarchical classification scheme based on the existing annotations (Brunstein, 2002). The NNE corpus has high inter-annotator agreement, achieving a Fleiss' kappa of 0.834. We perform substantial consistency analysis to ensure a high quality corpus.

We define twelve high-level principles that guided the development of the annotation scheme and guidelines, and resolved difficult decisions for annotators. We also monitored the induced semantic grammar as it evolved during the annotation process, seeking to identify and document common patterns, such as FIRST + NAME $\rightarrow$ PER, to maintain consistency between annotators and attempt to minimise rule proliferation.

We present a set of highly detailed NNE annotation guidelines, covering 118 named entities, and numerical and temporal expressions at a fine-grained level. The guidelines are a separate document, and this thesis contains an abridged version. These guidelines bring fine-grained, structural named entities to the same level of detail as the bracketing and part of speech annotation guidelines for the Penn Treebank.

We developed annotation tools that allow annotators to easily see previous annotation decisions within this document and in the entire corpus, and enables annotators to make decisions on a per-document and entire corpus level. This, combined with our detailed annotation guidelines and annotation principles, and with substantial consistency checking, has ensured a highly consistent corpus of structured named entities.

We merged our NNE with the PTB, requiring some systematic and one-off changes to both annotations. This allows the NNE corpus to complement other PTB resources, such as PropBank (Kingsbury and Palmer, 2002; Palmer et al., 2005) and NomBank (Meyers et al., 2004), and inform PTB-derived corpora for other formalisms, such as Combinatory Categorial Grammar (Hockenmaier, 2003) and LFG (Cahill et al., 2002). We compare this merged corpus against the original BBN annotations.

1.3.1 Learning nested named entities

We approach the task of learning nested named entities from a parsing perspective, presenting the first results of learning nested entities in English newswire text, and demonstrating that statistical methods can recover them with reasonable accuracy.

We consider several approaches to integrating the PTB and NNE annotations, which affect the sparsity of grammar rules and visibility of syntactic and NE structure. We explore their impact on parsing the NNE and merged variants using the Berkeley parser (Petrov et al., 2006), which performs surprisingly well without specialised NER features. We perform extensive error analysis on these different approaches.

We experiment with flattening the NNE annotations into linear NER variants with either top-most, bottom-most and stacked categories, and explore the ability of a maximum entropy and a CRF NER system to reproduce them. The

state-of-the-art CRF performs substantially better, but is infeasible to train on the enormous stacked category sets.

We compare the results of two NER systems to the performance of our parsing models on these semi-structural projections, finding that flattened output of the Berkeley parser are almost competitive with the CRF.

1.3.2 Analysing nested named entities

We present an analysis of the nested entities in the NNE corpus we create, finding a high number of structural entities with multiple layers of annotation. We discuss the analyses of the internal structure that form these nested named entities.

After merging our nested named entities with the syntactic structure of the PTB, we present a novel analysis of the effects of different merging algorithms on both parsing and named entity recognition. We also perform substantial error analysis in both the subtasks of parsing and named entity recognition.

1.4 Outline

We begin by reviewing the main tasks involved in learning the structure of named entities — parsing and named entity recognition — and by investigating the available corpora for both tasks in Chapter 2. We compare various annotation schemes used in NER corpora, and describe the importance of the Penn Treebank (PTB) in both parsing and the wider NLP field.

Chapter 3 describes the entity scheme used to annotate the corpus, outlining the annotation principles developed to ensure a detailed, consistent and useful corpus. We summarise the detailed entity guidelines used in the creation of the thesis, which bring the same level of detail as the syntactic PTB guidelines to the task of nested NER. We further describe why we elected to augment

the Wall Street Journal section of the Penn Treebank with nested named entity information, and outline the benefits from starting from existing, flat NER annotations from the BBN corpus.

In Chapter 4 we document the creation of the NNE corpus. We outline various pre- and post-processing steps we apply, as well as a custom annotation tool, designed to ensure the creation of an accurate and consistent corpus. We analyse the inter-annotator agreement of our annotators, conduct substantial error analysis, and analyse the resulting final NNE corpus.

Chapter 5 describes the process of merging our NNE annotations with the PTB constituency structure, to ensure a compatible corpus, which is necessary for parsing experiments. We outline the changes we make to both our NNE annotations and the syntactic constituents of the PTB, and evaluate the impact on consistence of both corpora that these changes have had.

In Chapter 6 we describe a number of ways of combining our NNE annotations and the syntactic constituents of the PTB into one corpus, and perform the first analysis of the impact of different methods of combining syntactic and NE semantic labels in a single constituency tree on the task of parsing. We use the Berkeley Parser (Petrov et al., 2006) to learn these combined structures, and report the first results of combined parsing and nested named entity recognition. We also analyse the effects of the changes made to the PTB in the merging process of Chapter 5, and demonstrate the utility of that syntactic structure in predicting nested named entity structure.

In Chapter 7 we evaluate how well existing NER systems, LIBSCHWA NER (Dawborn, 2015) and C&C NER tagger (Curran and Clark, 2003b), can learn structured NER by devising different projections of structured entities into flat individual labels. We compare the results of these NER systems to the results of the parsers trained in Chapter 6, and analyse the different types of errors made by the different systems.

Finally, Chapter 8 discusses avenues for future work, both in the specific task of NER and in various wider NLP tasks, and summarises the core contributions of this thesis.

This thesis contributes a significant new corpus for *nested named entity* recognition, the results of a number of novel experiments and sets the benchmark for the task of fine-grained, structured named entity recognition. In addition to this, it opens the field for further research, allowing nested named entities to be leveraged by systems in a wide range of NLP applications.

2 Background

> It is important that we know where we come from, because if you do not know where you come from, then you don't know where you are, and if you don't know where you are, you don't know where you're going. And if you don't know where you're going, you're probably going wrong.
>
> Terry Pratchett

The primary contribution of this thesis is a corpus of nested named entity structure. In this chapter, we outline how the definition of a named entity has developed as new datasets are released. The Message Understanding Conferences (MUC, Chinchor, 1998) originally defined named entities with the broad categories of people, locations and organisations, as well as temporal and numerical expressions. Since then, the largest developments have been the inclusion of a Miscellaneous (MISC) category, and the introduction of fine-grained category hierarchies. Nesting in NEs has always been posited as a logical area for future development, though limitations in the available corpora annotated with nested entity structures have restricted this in standard NER tasks. This thesis addresses this shortfall, providing an accurate, fine-grained and nested named entity corpus.

Since the task of predicting the nested structure of named entities is relatively new, there is no single well-established way to learn the task, nor, more

importantly a generally accepted and comparable way to evaluate how well such a learnt model can apply to unseen text. In this thesis, we propose using methods from syntactic parsing to model structured named entity recognition. We consider both the subtasks of constituency parsing and named entity recognition for evaluation.

In this chapter, we therefore discuss both constituency parsing and named entity recognition, both of which will be further discussed in their respective chapters (Section 6.1 and 7.1). We outline the development of constituency parsing, highlighting the important role that the Penn Treebank (PTB, Marcus et al., 1993) has had in allowing for the development of statistical models, and providing the mechanism on which those models are evaluated. We then discuss NER before reviewing related work on nested named entities primarily in Biomedical NLP.

2.1 Overview of tasks

In this section, we introduce the main tasks involved in learning the structure of named entities: parsing and named entity recognition. In particular we focus on the formalisms that will be important for understanding the methods used.

2.1.1 Parsing

Parsing, the process of determining the syntactic structure of a sentence, is an important natural language processing (NLP) task. Accurately parsing text is a vital step in many automatic language processing tasks, such as question answering (QA).

The most common type of syntactic representation of a parse is shown in Figure 2.1, which shows the first sentence of the Penn Treebank, a large-scale corpus annotated with gold-standard parse trees. In a constituency grammar, a

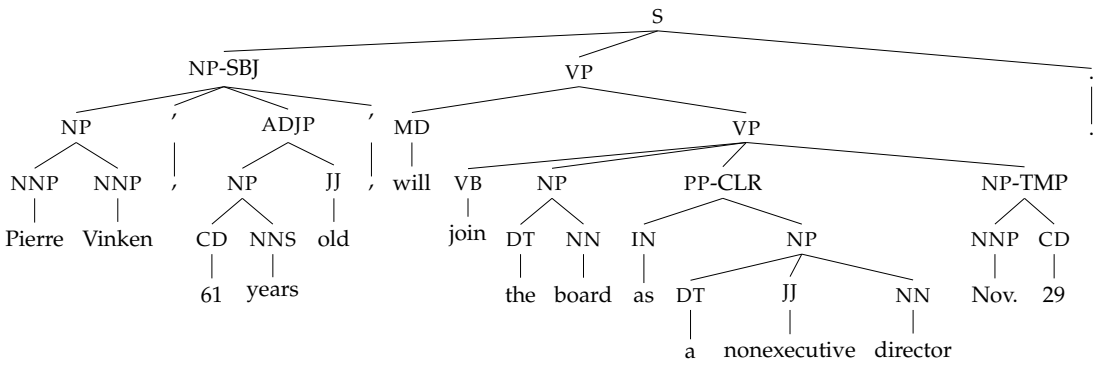


Figure 2.1: Constituent structure for the first sentence of the Penn Treebank (Marcus et al., 1993).

sentence is recursively decomposed into smaller sequences of words that are each labelled based on their internal structure. Different phrase labels relate to the dominant head word in the phrase. Phrase structures are induced by context-free grammars and are used in a variety of grammatical frameworks including Lexical Functional Grammar (LFG) (Kaplan and Bresnan, 1982), Tree Adjoining Grammar (TAG) (Joshi and Schabes, 1992), and Head-driven Phrase Structure Grammar (HPSG) (Pollard and Sag, 1994). In this thesis, we assume constituent structures as the representation of syntactic elements.

The PTB is not only a useful standalone resource of its own right, but has also enabled a large number of corpora in different grammatical formalisms to be created. The combination of both phrase labels and Trace elements in the PTB has enabled it to be used as the basis for statistical models used to generate treebanks in other grammars, such as CCG (Hockenmaier, 2003; Hockenmaier and Steedman, 2007), HPSG (Miyao et al., 2004), and LFG (Cahill et al., 2002).

Thus, an advantage of adding additional annotations to the PTB is that this layer of information can be transferred through to all derivative resources in different formalisms.

2.1.2 Named Entity Recognition

Named entity recognition (NER) is the task of automatically identifying proper names, numerical and temporal expressions in text and classifying them according to a pre-defined set of categories. General domain NER includes identifying the names of people (PER), locations (LOC) and organisations (ORG). This has been extended in different ways, both by splitting existing categories into finer-grained distinctions with or without an explicit hierarchy, and by the inclusion of additional categories beyond the scope of people, locations and organisations.

[Nicky Ringland]$_{\text{PER}}$ attends the [University of Sydney]$_{\text{ORG}}$.

Finer grained categories are often grouped in a hierarchical structure (Sekine et al., 2002; Li and Roth, 2002; Sekine and Nobata, 2004; Sekine, 2008; Ling and Weld, 2012) whereby, for instance, multiple types of organisation (CORPORATE, EDUCATIONAL, GOVERNMENT, etc.) sitting under an ORGANISATION category. Hierarchical ontologies allow for consistent labeling in certain cases of ambiguity. For instance the soccer team Red Bull Brasil may have surface form Red Bull, and in context it may be unclear whether this refers to the ORGANIZATION:CORPORATE or SPORTSTEAM. The parent label (ORGANIZATION) can be used instead. Other categorisation schemes may not allow parent categories to be used as labels, but instead have an category for cases such as this, or for other specific types of organisations not specified by individual categories. Fine grained classification hierarchies are further discussed in Section 2.4.8.

Extended named entity categorisations grow the scope of entities of interest. They usually include a miscellaneous category (MISC) which can include works of art, languages and product names. MISC also frequently includes entities which are not nouns, in certain adjectival forms, sometimes known as NORP, standing for NATIONALITY, OTHER, RELIGION and POLITICAL.

Temporal and Numerical expressions are also occasionally included in the NER task. Orthographically, their form is very different from other NEs, and semantically they also differ in that they lack a real-world referent. NUMEX expressions are lowercase, and are relatively detectable based on being primarily composed of numbers and a small set of words. TIMEX expressions are a mix of both proper (Sunday) and common noun (next week) phrases, and the scope of the category varies between annotation guidelines. NUMEX and TIMEX expressions can be quite complex, yet mostly regular. Unlike other categories in the NER task, a large number of NUMEX and TIMEX expressions can be captured with simple regular expressions.

Annotation categories and corpora vary based on the specific application for NER. In the biomedical domain, corpora are annotated with entities including genes, proteins and chemical substances. For pragmatic reasons, and the low number of instances of these entity types, these entities are rarely annotated in general domain or newswire NER corpora. Similarly, specific properties of the biomedical domain have resulted in nesting of entities being annotated in some biomedical corpora, further discussed in Section 2.4.1.

The area of NER developed from the field of information extraction (IE) in the Message Understanding Conferences (MUC) held in the 1990s (Chinchor, 1998). At the time, MUC was primarily focused on IE tasks where structured information, such as a company or financial 'event', is extracted from unstructured text such as newspaper articles.

NER has since evolved into a distinct task which is an essential pre-processing step in various NLP pipelines including question answering, information retrieval, coreference resolution and slot filling.

Due to the lack of appropriate resources, most NER work has assumed a flat NE structure. This has meant that entities such as [The University of Sydney]$_{\text{ORG}}$ and the [Sydney Swans]$_{\text{ORG}}$ are analysed as ORG, in effect forcing spurious

ambiguity onto the token [Sydney]$_{\text{CITY}}$, which is acting as CITY in both cases. It also means that we have been unable to utilise structural information in entities such as the [Bill and Melinda Gates Foundation]$_{\text{ORG}}$, from which we should be able to derive both [Melinda Gates]$_{\text{PER}}$ and [Bill Gates]$_{\text{PER}}$, which would be especially useful for coreference tasks when co-located with [Mr. Gates]$_{\text{PER}}$. The dependence of machine learning approaches on large annotated training corpora has proven to be a bottleneck in this.

2.2 The Penn Treebank corpus

The development of syntactically annotated corpora revolutionised computational linguistics, allowing the field to develop from predominantly rule-based to statistical methods.

The Penn Treebank (Marcus et al., 1993) has been hugely influential in statistical methods for learning syntactic parse structures in English. Developed between 1989 and 1996, the Penn Treebank was the first large-scale corpus to be manually annotated with gold-standard constituency trees.

The Treebank has labelled brackets describing the syntactic structure of each constituent in the sentence and part of speech (POS) tags labelling each word. These phrase brackets and POS tags are summarised in Table 2.1, and an example of their use is given in Figure 2.2, where [Cotton Inc.]$_{\text{CORP}}$ is an NP composed of two proper nouns (NNP).

The release of the Penn Treebank allowed for supervised, statistical experiments in parsing, and it remains the canonical parsing dataset for English. However, the Penn Treebank does have certain known limitations, both with respect to the grammatical structures with which it is annotated, and the presence of errors in the data.

The Penn Treebank I was released in 1991 (Marcus and Santorini, 1991), followed by the Penn Treebank II in 1993 (Marcus et al., 1993). The re-release of the corpus updated the bracketing guidelines, modifying the way trace elements are handled, and adding a set of functional markers used to indicate semantic structure. These enabled the modeling of grammatical relations (Marcus et al., 1994). In Figures 2.1 and 2.2, SBJ marks the subject NP, while LOC and TMP mark locative and temporal elements respectively. The annotation of these semantic elements proved difficult (see Marcus et al., 1994), and the resulting functional markers are used inconsistently throughout the corpus.

The handling of various types of NE are also captured in Figures 2.2 and 2.3. Some NEs, e.g. `(NP (NNP Cotton)(NNP Inc.))` and `(NP (NNP Albert)(NNP M.)(NNP Kligman))`, form constituent NPs, though only each individual token is identified as a proper noun, not the larger constituent span. The analysis of Thanksgiving Day in Figure 2.2 shows an example of inconsistent treatment of a constituent due to proper nouns being analysed with flat structure. The first instance forms its own constituent NP, while the second forms part of a larger NP: Macy 's Thanksgiving Day Parade and does not have its own constituent. This is further explored and a solution proposed in Chapter 5.

In Figure 2.3, the analysis for [University of Pennsylvania School of Medicine]$_{\text{ORG}}$ is even more problematic, with the substructure `(NP (NP the University of Pennsylvania School)(PP of Medicine))` splitting the mention School of Medicine. We further examine these inconsistencies and solve the problem in Chapter 5.

2.2.1 Adding Noun Phrase Structure to the PTB

The original Penn Treebank annotation guidelines did not include extensive noun phrase structure, electing instead to simplify the annotation task and speed up bracketing decisions by avoiding adding any structure for nominal modifiers as far as possible (Marcus et al., 1993).

```
( (S
    (NP-SBJ-1 (NNP Cotton) (NNP Inc.) )
    (VP (MD will)
      (VP (VB spend)
        (NP
          (QP (RB nearly) ($ $) (CD 2) (CD million) )
          (-NONE- *U*) )
        (PP-CLR (IN on)
          (NP (NN broadcasting) ))
        (PP-TMP (IN on)
          (NP
            (NP (NNP Thanksgiving) (NNP Day) )
            (ADVP (RB alone) )))
        (, ,)
        (S-ADV
          (NP-SBJ (-NONE- *-1) )
          (VP (VBG advertising)
            (PP-LOC (IN on)
              (NP
                (NP (JJ such) (NNS programs) )
                (PP (IN as)
                  (NP (`` ``)
                    (NP (JJ Good) (NN Morning) (NNP America) )
                    (, ,) ('' '') (`` ``)
                    (NP
                      *(NP (NNP Macy) (POS 's) )
                      (NNP Thanksgiving) (NNP Day) (NNP Parade) )
                    ('' '')
                    (CC and)
                    (NP (DT the) (NNP NFL) (NN holiday) (NN game) ))))))))
    (. .) ))
```

Figure 2.2: An example sentence (`WSJ0295_53`) from the Penn Treebank (Marcus et al., 1993), with named entities marked in blue.

ADJP	Adjective Phrase	PRN	Parenthetical
ADVP	Adverb Phrase	PRT	Particle
CONJP	Conjunction Phrase	QP	Quantifier Phrase; used within NP
FRAG	Fragment	RRC	Reduced Relative Clause
INTJ	Interjection	UCP	Unlike Coordinated Phrase
JJP	Adjectival Phrase*	VP	Verb Phrase
LST	List marker	WHADJP	Wh-adjective Phrase
NAC	Not a Constituent	WHAVP	Wh-adverb Phrase
NML	Nominal Phrase*	WHNP	Wh-noun Phrase
NP	Noun Phrase	WHPP	Wh-prepositional Phrase
NX	Head of complex NP		
PP	Prepositional Phrase	X	Unknown or unbracketable
CC	Coordinating conj.	TO	infinitival to
CD	Cardinal number	UH	Interjection
DT	Determiner	VB	Verb, base form
EX	Existential there	VBD	Verb, past tense
FW	Foreign word	VBG	Verb, gerund/present participle
IN	Preposition	VBN	Verb, past participle
JJ	Adjective	VBP	Verb, non-3rd ps. sg. present
JJR	Adjective, comparative	VBZ	Verb, 3rd ps. sg. present
JJS	Adjective, superlative	WDT	Wh-determiner
LS	List item marker	WP	Wh-pronoun
MD	Modal	WP$	Possessive wh-pronoun
NN	Noun, singular or mass	WRB	Wh-adverb
NNS	Noun, plural	#	Pound sign
NNP	Proper noun, singular	$	Dollar sign
NNPS	Proper noun, plural	.	Sentence-final punctuation
PDT	Predeterminer	,	Comma
POS	Possessive ending	:	Colon, semi-colon
PRP	Personal pronoun	(	Left bracket character
PP$	Possessive pronoun	)	Right bracket character
RB	Adverb	"	Straight double quote
RBR	Adverb, comparative	‘	Left open single quote
RBS	Adverb, superlative	“	Left open double quote
RP	Particle	’	Right close single quote
SYM	Symbol	”	Right close double quote

Table 2.1: The Penn Treebank phrase level bracket labels and part of speech (POS) tagset. *NML and JJP were introduced by Vadas and Curran (2007) and Vadas (2007) for noun phrase structure.

```
( (S
    (PP-TMP (IN In)
      (NP (NNP May)))
    (, ,)
    (NP-SBJ (NNP University) (NNP Patents) )
    (VP (VBD filed)
      (NP
        (NP (DT a) (NN suit) )
        (PP (-NONE- *ICH*-1) ))
      (PP-LOC (IN in)
        (NP (JJ federal) (NN court) ))
      (PP-LOC (IN in)
        (NP (NNP Philadelphia) ))
      (PP-1 (IN against)
        (NP
          (NP (NNP Albert) (NNP M.) (NNP Kligman) )
          (, ,)
          (NP
            (NP (DT a) (NN researcher)
              (CC and) (NN professor) )
            (PP-LOC (IN at)
              (NP
                (NP (DT the)
                  (NAC (NNP University)
                    (PP (IN of)
                      (NP (NNP Pennsylvania) )))
                  (NNP School) )
                (PP (IN of)
                  (NP (NNP Medicine) ))))
            (SBAR
              (WHNP-182 (WP who) )
              (S
                (NP-SBJ-2 (-NONE- *T*-182) )
                (VP (VBD developed)
                  (NP (NNP Retin-A) )
                  (PP-TMP (IN in)

                    (NP (DT the) (CD 1960s) ))
                  (S-PRP
                    (NP-SBJ (-NONE- *-2) )
                    (VP (TO to)
                      (VP (VB combat)
                        (NP (NN acne) ))))))))))
    (. .) ))
```

Figure 2.3: An example sentence (`WSJ0081_2`) from the Penn Treebank (Marcus et al., 1993), with named entities marked in colour. The named entity in red has incorrect substructure.

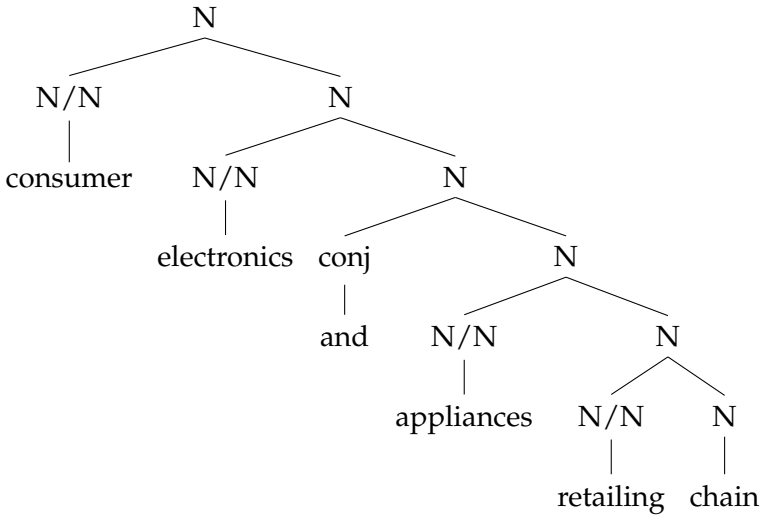


Figure 2.4: CCG derivation from Hockenmaier (2003)

Vadas and Curran (2007) and Vadas (2010) extend the existing Treebank annotations by adding the internal structure of noun phrases. They motivate the task of identifying noun phrase structure in base-NPs, citing their importance in NLP systems (NPs as factoid answers for Question Answering systems), as well as other annotated data derived from the Penn Treebank.

CCGbank (Hockenmaier, 2003; Hockenmaier and Steedman, 2007), for example, was created by semi-automatically converting the Treebank phrase structures to Combinatory Categorial Grammar (CCG Steedman, 1996, 2000), a grammatical formalism which is binary branching and, as such, cannot directly represent the flat structures of Penn Treebank noun phrases. The conversion process, in the absence of structural NP information, constructed strictly right-branching trees for all base-NPs, an example of which is shown in Figure 2.4.

The annotation guidelines used by Vadas and Curran (2007) build on those for annotating full sub-NP structures from the biomedical domain (Kulick et al., 2004), which add nominal (NML) nodes to add internal NP structure. Vadas and Curran add NML or Adjectival phrase (JJP) nodes for structures that are left-branching, and leave right-branching structures flat. All potentially ambiguous NPs, defined as NPs with three or more contiguous children that

```
(NP
  (NML (JJ Good) (NN Morning) )
  (NNP America) )

(NP
  (NP (NNP Macy) (POS 's) )
    (NML (NNP Thanksgiving) (NNP Day) )
    (NNP Parade) )
```

Figure 2.5: Changes from sentence (WSJ0295_53) following addition of noun phrase structure (Vadas and Curran, 2007). Note the inclusion of NML node labels.

are either single words or other NPs, in the PTB were annotated. Some common structures, such as an NP of three words starting with a determiner were filtered out as unambiguous.

Vadas and Curran drew structural suggestions from the boundaries of named entities from the BBN corpus (Weischedel and Brunstein (2005a), see Section 2.3.4), as well as from previous bracketings of the same words. Post-processing checks were carried out to ensure annotation consistency. 22851 NPs of a total of 60959 ambiguous NPs (37.49%) were found to be non right-branching, and had brackets inserted.

Vadas and Curran (2007) used Bikel's (2004) implementation of Collins' parser (Collins, 1999) to evaluate the addition of NP structure to the PTB, finding that the additional brackets make parsing marginally more difficult (88.46 F_1-score, down from 88.92). They further evaluate on only the NML and JJP brackets which were inserted, achieving an F_1-score of 69.63, and correspondingly evaluate excluding the NML and JJP brackets which were inserted, achieving an F_1-score of 88.89. These results show the difficulty of correctly bracketing NPs, while also demonstrating that the performance of other phrases is not badly affected by the new NP brackets that were inserted.

2.2.2 Other resources built on the PTB

In addition to enabling the automatic creation of corpora in other grammatical formalisms, such as CCG (Hockenmaier, 2003) and LFG (Cahill et al., 2002), the PTB has also served as the basis of a number of other linguistic resources such as PropBank (Kingsbury and Palmer, 2002; Palmer et al., 2005) and NomBank (Meyers et al., 2004), meaning that these resources are able to be mapped back and through to the derivative treebanks in other grammatical formalisms. PropBank specifies the predicate argument relationships between verbs and the arguments of those verbs. NomBank provides argument structure for instances of common nouns in the PTB corpus.

The goal of the PropBank and NomBank projects is to lead to the creation of better tools for the automatic analysis of text. Meyers et al. (2004) suggest that the standardisation which is imposed by the annotated data will ensure that more researchers will work within the same set of frameworks, and as such, each individual's research will be more directly applicable to the larger research community.

Additionally, as the same set of data is annotated with additional layers of annotation, new forms of multistage processing become possible. Consider an information extraction task where acquisitions are of interest. A system using PropBank and NomBank could generalise over patterns involving the verb acquire, learning that [Sony Corp.]$_{\text{CORP}}$ acquired [Columbia Pictures]$_{\text{CORP}}$ from both surface forms Sony Corp. acquired Columbia Pictures, Columbia Pictures is being acquired by Sony Corp., Sony's recent acquisition of Columbia Pictures ... etc. When combined with the fine-grained NE corpus presented in this thesis, the system could additionally learn that both the first and second argument of acquire are often CORP companies.

Unfortunately, if these corpora which offer additional annotation layers do not align properly with the Treebank, the utility of the resource is impacted. PropBank and NomBank do not modify the PTB annotations, even in cases of incompatible alignment. By electing to leave the PTB unmodified, some arguments span over incompatible syntactic spans, which is problematic for systems parsing this information, such as those which use parsers for the task of semantic role labelling. These systems essentially incur a penalty, since they are unable to produce the correct PropBank span.

OntoNotes (Hovy et al., 2006; Weischedel et al., 2010, 2013) combines multiple layers of annotation, including the PTB, PropBank and NomBank, as well as additional layers of semantic annotation. Babko-Malaya et al. (2004) describe the merging the English Treebank and PropBank, detailing the changes that were necessary to make to both corpora to address the conflicting annotations. OntoNotes further adds to these annotation layers with resolving word sense ambiguity (linking each to the Omega ontology Philpot et al., 2005), and annotating coreference. The OntoNotes project further grew to include named entity annotation in later releases.

The OntoNotes project has been developing in parallel to the work presented in this thesis, and the extension of this work to the OntoNotes corpus would be a natural progression, which we suggest as future work.

2.2.3 Tokenisation in the Penn Treebank

The preprocessing task of tokenisation, that is, splitting a sentence up into discrete tokens, is important for all NLP tasks. In most cases, this is considered a preprocessing step that has already been solved, and the tokens that for a sentence are considered independent lexemes and punctuation marks. Many phenomena complicate tokenisation, including abbreviations (e.g. USPS, the United States Postal Service), date or number expressions (e.g. the 1980s), and

certain adjectival forms (e.g. London-based). Many of these tokenisation issues are of particular interest to the combined task of parsing and NER, however substantial tokenisation changes are rarely made to an existing, establish corpus such as the Penn Treebank. In this thesis, we accept this limitation, but note the issues that are compounded based on tokenisation decisions.

2.2.4 Summary

The PTB is one of the most influential resource in NLP, alongside other resources such as WordNet (Pedersen et al., 2004). It has had an enormous impact, both directly on the task of parsing, and in the wider field of NLP. Over time, various deficiencies (such as NP bracketing for NP structure) have been rectified, and additional resources have been built both from it (CCGbank) and on top of it (PropBank, NomBank) that make it an even richer resource. This thesis builds on top of the PTB and enriches the full ecosystem of resources and systems that stem from it.

2.3 Named Entity Corpora

As with parsing, the dominant methods in current named entity recognition use data-driven statistical approaches. The costs associated with procuring the data are high. For English NER training, corpora originating from conference evaluations of named entity technology (MUC, IEER, ACE and CoNLL) are most widely used, in both general and specific domains such as biomedicine. This can affect the usefulness of the corpora since they are, to a large extent, only annotated with the specific labels that are of interest to that particular conference or shared task.

The BBN Pronoun Coreference and Entity Type Corpus (BBN) did not originate from a conference or competition. It contains a wide variety of fine-grained entities, but has not been widely utilised by the research community.

Corpora with annotated named entities are essential for developing and evaluating named entity recognition (NER) systems. NER systems can only be as reliable as their training sources and world knowledge, and can only be as detailed as the training data and annotation schemas. Machine learning approaches to NER require training corpora with gold-standard annotations which can be analysed statistically to produce a predictive model. Since training texts are traditionally annotated manually by linguistic experts, they are costly to produce and generally small in size – up to 1.2 million tokens in BBN's annotation of Wall Street Journal text (Weischedel and Brunstein, 2005b). More recent work has explored a distant supervision approach using Wikipedia to generate corpora automatically (Richman and Schone, 2008; Nothman et al., 2013), however, these large corpora are not extensively used for training or evaluation of NER systems. The domain of the corpus is also important, as performance is considerably lower when NER systems are trained on out of domain corpora. This has led to the creation of domain specific corpora, such as Liu et al. (2011), who have annotated named entities in Tweets.

An overview of the size, domain, number of tags and presence of nested entities in commonly used named entity corpora is shown in Table 2.2.

2.3.1 MUC 6 and 7 and MET

The Message Understanding Conferences (MUC) which ran from 1987 to 1997 were designed to encourage the development of information extraction methods. These competitions involved both the creation of substantial amounts of data and new standards for evaluation. The MUC-6 (Sundheim, 1995) and MUC-7 (Chinchor, 1998) shared tasks compartmentalised named entity recogni-

Corpus	# tags	# tokens	Nesting	Domain
MUC-6	7	23,773*	no	Newswire
MUC-7	7	149,249*	no	Newswire
CoNLL 2003	4	301,418	no	Newswire
ACE 2004	43	189,620	some	Newswire, Broadcast & Web
BBN	64	1,173,766	no	Newswire
OntoNotes	18	450,000†	yes	Newswire
ACE 2008	31	245,000‡	some	Newswire, Broadcast & Web
GENIA	36	436,967	yes	Biomedical
AnCora	6	n/a	yes	Non-English Newswire

Table 2.2: Statistics of various English NER datasets. Note that '# tags' includes named entity, TIMEX and NUMEX categories. *Approximate numbers based on untokenised text. †The ACE 2008 token numbers are unavailable for evaluation data; reported is tokens in training and development data. ‡Number of tokens in English documents annotated with named entity information, of a total of 1,631,995 English tokens.

tion from information extraction, and involved participants identifying named entities in text, and categorising them into seven subcategories split over three groups: entity expressions (ENAMEX, which includes LOCATION, ORGANISATION and PERSON), temporal expressions (TIMEX, which includes DATE and TIME), and numerical expressions (NUMEX, which includes MONEY and PERCENT). The MUC-6 and MUC-7 datasets are in English, sourced from the MUC-6 Text Collection and North American News Text Corpora respectively.

2.3.2 CoNLL

Following from the success of the MUC competitions, in 2002, the Conference on Natural Language Learning (CoNLL) shared task continued with NER. CoNLL added a new category to the MUC NER task: Miscellaneous (MISC), a hard

category covering a diverse range of entities including works of art, nationalities, languages, religions and products. CoNLL also added a multilingual dimension to the competition in order to encourage more statistically driven techniques in NER. This was partly in response to the primarily rule-based systems which dominated the MUC tasks, and the difficulty of building rule-based systems which work across languages. It was also an effort to encourage feature based systems in which more general features were implemented. Due to its focus on machine learning, CoNLL introduced a larger dataset than any previously available.

The CoNLL 2003 (Tjong Kim Sang and De Meulder, 2003) shared task refocused on German and English, the corpus for the latter being a subset of the Reuters 1996 news corpus (Rose et al., 2002). Following on promising results for NUMEX and TIMEX expressions using rule-based methods in MUC and MET, CoNLL focused on ENAMEX, with a view of encouraging systems that could learn to model NER language-independently. Both CoNLL shared tasks required participants to identify four categories of NEs: LOCATION, ORGANIZATION, PERSON and the new category of MISCELLANEOUS. Both the training and the development datasets are news feeds from August 1996, while the test set is drawn from news feeds from December 1996. Since the development set was drawn from the same time period as the training set, systems achieve substantially higher performance on the development set than the test set. Since its development, despite the identification of various tokenisation issues and resulting in consequent sentence boundary mistakes, the CoNLL 2003 English dataset has become the canonical evaluation dataset for English NER.

2.3.3 ACE 2004 and ACE 2008

The Automatic Content Extraction (ACE) program started in 1999 with the aim of developing technology to automatically infer entities, relations and events from human language. Over the course of several tasks, the focus shifted to entity resolution (the task of identifying, disambiguating and linking different mentions of real-world entities) as a goal, rather than purely identification and classification.

The ACE 2004 corpus (Doddington et al., 2004) includes all references to an entity, including names, descriptions and pronouns, that are then collected into equivalence classes based on reference to the same entity. As such, the ACE task involved both entity recognition and coreference resolution. ACE 2004 built from the CoNLL annotation schema, specifically introducing GEOPOLITICAL ENTITY (GPE) as a category for evaluation. This distinction dealt with the ambiguity of entities, such as countries and states, that have both organisational and locative properties.

2.3.3.1 Nested entities in ACE

The Entity Detection and Tracking (EDT) task of ACE 2004 included identifying seven entity types (with further subtypes): PERSON (no subtypes), ORGANIZATION (5 subtypes), LOCATION (10 subtypes), FACILITY (8 subtypes), WEAPON (9 subtypes), VEHICLE (5 subtypes) and GEO-POLITICAL ENTITY (GPEs) (6 subtypes). Nested mentions of entities are also captured, including both direct named entity mentions and nominal (both prenominal and pronominal) mentions.

The named entity component was framed in terms of nominal modification, and as such does not address full entity nesting. Nested mentions are only annotated in nominal mentions, not inside other named entity mentions. Fig-

$$\overline{\underline{\text{Indonesia}}_{GPEnam}\ \text{'s war} - \text{torn}\ \underline{\text{Aceh}}_{GPEnam}\ \text{province}}^{GPEnom}$$

Figure 2.6: Example of nesting of named entities in nominal phrase in ACE 2004 data.

> "The summit , which is being sponsored by $[^{GPE}_{nam}$ the European Union], is meant to show $[^{GPE}_{nom}$ the $[^{GPE}_{nam}$ Balkan] states] that $[^{GPE}_{nam}$ the EU] is preparing to welcome $[^{GPE}_{pro}$ them] into $[^{GPE}_{nom}$ the $[^{GPE}_{pre}$ European] family]."

Figure 2.7: Sentence from ACE 2004 data demonstrating the nesting of European when in nominal phrases, but not inside other entities (European Union).

ure 2.6 shows Aceh marked as a GPE embedded within a nominal span, but the larger span Aceh province is not annotated. In Figure 2.7, the token European is marked as a GPE when used as a prenominal modifier (the European family), but not marked as part of the named entity (the European Union).

The most recent ACE shared task, (Strassel et al., 2008), involved participants identifying 31 categories of NEs (see Table 2.3) as subtypes of PERSON, LOCATION, ORGANIZATION, FACILITY and GEO-POLITICAL ENTITY in English and Arabic text from a variety of domains, including newswire, weblogs, Usenet newsgroups and bulletin boards, and transcripts of broadcast news, talk shows and conversational speech.

The ACE 2008 corpus includes some nested entities, including nested region names. For example, [BORDEAUX , [France]] where Bordeaux , France, the city, and France, the country, are both locations which should be marked, since, as a series of nested region names, it evokes one entity for each region. However, despite metonymy being identified as a particular problem case, nested entities are not included in metonymous entity instances. For example, in the sentence Miami is growing rapidly, Miami is marked as a geo-political entity (GPE) named Miami, while in the sentence Miami defeated Atlanta 28 to 3, Miami is a metonymic

Type	Subtype
Facility	Airport, Building-Grounds, Path, Plant, Subarea-Facility
Geo-Political Entity	Continent, County-or-District, GPE-Cluster, Nation, Population-Center, Special, State-or-Province
Location	Address, Boundary, Celestial, Land-Region-Natural, Region-General, Region-International, Water-Body
Organization	Commercial, Educational, Entertainment, Government, Media, Medical-Science, NonGovernmental, Religious, Sports
Person	Group, Indeterminate, Individual

Table 2.3: Types and Subtypes of entities in the ACE 2008 English corpus.

mention of a sports organisation entity called the Miami Dolphins, and as such is distinct from the Miami GPE (NIST-ACE, 2008). However, non-location multi-token entities such as The New York Times are considered atomic, and are not be annotated with nested structures (for instance, embedded GPE: New York).

The adjective Russian, as in Russian grandmothers, evokes the GPE Russia, and is therefore be marked as GPE, though the noun, grandmothers, is not annotated. However, when Russian is used as a noun, e.g. Several Russians recently. . . , it confusingly is *not* marked as a GPE, but is left unannotated.

Further consider The White House, which is considered an organisation when it is taking action (e.g. The White House vetoed the bill .), and a facility when the physical building itself is referred to (e.g. A spokesperson from The White House said. . .).

2.3.3.2 HAPNIS

Although the ACE 2004 data did not include annotations for the internal structure of Person entities, Hal Daume III manually annotated a small subset of the data (totalling 220 names) and developed the High Accuracy Parsing of Name Internal Structure (HAPNIS) script[1]. The script annotates the internal structure of people's names with tags: surname, forename, middle, link (e.g. hyphen

[1]The data and script are both available at http://www.umiacs.umd.edu/~hal/HAPNIS/

between double barrel surnames), role (Ms., Dr., etc.), suffix (Jr., III, etc.), and continue (for use when forenames are separated by a space, e.g. Lee Ann). The simple Perl script uses a series of heuristics in making classification decisions based on information such as the position of tokens in a name, the total number of tokens, the presence of punctuation such as periods and dashes, as well as a small gazetteer of common first name. Although tested on a small set (120 instances), and slightly biased towards calling single-word entries surnames rather than first names, as a consequence of being trained on mostly newswire, the script achieves very high precision.

2.3.4 BBN

The BBN Pronoun Coreference and Entity Type Corpus (BBN) was created in 2005 (Weischedel and Brunstein, 2005b), contributing additional annotation layers to the one million word Penn Treebank corpus of Wall Street Journal texts (see Section 2.2). The entity annotations are split into 12 named entity types (PERSON, FACILITY, ORGANIZATION, GPE, LOCATION, NATIONALITY, PRODUCT, EVENT, WORK OF ART, LAW, LANGUAGE, and CONTACT-INFO), ten DESCRIPTOR (nominal) entity types (PERSON, FACILITY, ORGANIZATION, GPE, PRODUCT, PLANT, ANIMAL, SUBSTANCE, DISEASE and GAME), and 7 numeric and temporal types (DATE, TIME, PERCENT, MONEY, QUANTITY, ORDINAL and CARDINAL). Several of these types are further divided into subtypes. For example, Organization is split into 11 different subcategories, seen in Table 2.4. The BBN corpus contains a total of 64 subcategories of named entities, with 156,780 entity instances, making it both substantially larger than other manually annotated English language NER corpora, and of much finer-grained annotations.

BBN also annotates nominal in addition to named entity phrases, and calls these *descriptor* types. For example, the PERSON NE category has a correspond-

ORGANIZATION subcategory	number of instances
CORPORATION	23,441
GOVERNMENT	4,629
POLITICAL	413
EDUCATIONAL	366
HOTEL	60
RELIGIOUS	44
HOSPITAL	23
MUSEUM	14
CITY	2
STATE_PROVINCE	1
OTHER	1,255

Table 2.4: Number of instances of subtypes of ORG in the BBN corpus.

ing PERSON DESCRIPTOR category, with which any head words of common nouns referring to a person or group of people should be marked. This includes occupational titles in modifier positions, such as President in the phrase President Bush. In this descriptor class, however, honorific titles (Mr, Sir etc.), are not annotated.

The annotation guidelines[2] for the BBN corpus are not very detailed, and there is no discussion of inter-annotator agreement, or how the annotations were made. The extents of descriptor annotation spans are particularly unclear. [executive]$_{PER\_DESC}$ occurs more than 400 times in the corpus. The annotation span is only grown a few times to [senior executive], [ex-chief executive], [chief executive], [finance executive] or [deputy chief executive]. However, the annotation [chief executive officer]$_{PER\_DESC}$ occurs more than 200 times in the corpus. This inconsistency is not explained in the annotation guidelines, where [Chief Executive] Sir Christopher Hogg is given as an example of the category.

The BBN corpus does not include nested named entity annotations, but does provide one of the most fine-grained approaches to general domain NER. The corpus is described in more detail in Section 3.

[2] https://catalog.ldc.upenn.edu/docs/LDC2005T33/BBN-Types-Subtypes.html

2.3.5 OntoNotes

The OntoNotes corpus (Hovy et al., 2006; Weischedel et al., 2010) is a large, multilingual corpus developed in collaboration between BBN Technologies, the University of Colorado, the University of Pennsylvania and the University of Southern California's Information Sciences Institute. The OntoNotes corpus contains multiple layers of annotations, including both structural information (syntax and predicate argument structure) and shallow semantics (word sense linked to an ontology and coreference). The corpus contains sections in English, Arabic and Chinese, and is drawn from various domains including news, conversational telephone speech, weblogs, Usenet newsgroups, broadcasts and talk shows.

The syntactic annotation layer of OntoNotes follows the Penn Treebank syntactic guidelines, making it a useful parsing resource. The full OntoNotes corpus contains 15,710 documents, of which 13,109 are in English. However, only 3,637 of these English documents are annotated with named entity information.

OntoNotes entity annotations are split into 11 types of named entity (PERSON, ORGANIZATION, LOCATION, FACILITY, GPE, NORP (NATIONALITY or RELIGIOUS, POLITICAL or OTHER groups), PRODUCT, EVENT, WORK OF ART, LAW, LANGUAGE), and 7 numerical and temporal entity types (DATE, TIME, PERCENT, MONEY, QUANTITY, ORDINAL, CARDINAL) (Weischedel et al., 2013). Entities that do not fall into these specific categories are not annotated, and nested entities are not annotated.

The particular utility of the OntoNotes corpus comes from the multiple annotation layers. The corpus is annotated with syntactic constituency spans, and a number of additional semantic annotation layers, including named entities. However, these named entity spans were not annotated within constituency

boundaries, nor were they later adjusted to be compatible with them. There are over 5000 instances where the entity span does not match the closest syntactic constituency span. Many of these would be relatively straightforward to solve, such as the majority of instances where a person's role is included in the NP with their name, as is the case with entity Bill Clinton, where the closest constituent span was President Bill Clinton. Other cases would be more problematic to resolve, including incompatibilities with prepositional phrases in entities (entity bounds: War of Resistance against Japan, syntactic constituent: China 's War of Resistance against Japan) as well as numerical expressions (entity bounds: tens of millions of dollars, syntactic constituent: tens of millions of dollars in tax payments to the Palestinian Authority). Other common span mismatches included qualifying tokens or phrases found (entity bounds: six months, syntactic constituent: over six months, entity bounds: 17 %, syntactic constituent: no less than 17 %). Resolving these span mismatches into compatible annotations would be necessary to enable both datasets to be used to their full potential, and especially to allow, for example, NE information to be pushed on to syntactic constituents and beyond.

2.4 NE Corpora containing Nested Structures

In addition to the corpora discussed above, some of which contain some limited degree of nested structures, a number of other corpora contain fully-nested constructions allowing for any embedded references to named entities, consistently across entity types.

2.4.1 GENIA

The GENIA corpus (Kim et al., 2003), a semantically annotated corpus for bio-text mining, was created to evaluate information extraction for molecular biology literature. The biomedical domain's specialised terminology and com-

plex naming conventions have resulted in a situation where entities of interest, such as genes, proteins or disease names, often nest. In order to capture these entities, a nested annotation structure is needed. The GENIA corpus contains nested entities such as <RNA><DNA>CIITA</DNA> mRNA</RNA>, referring to the RNA entity: CIITA mRNA and the embedded DNA entity: CIITA. Some of these nested entity layers occur on a sub-token level.

The GENIA corpus contains nested embedding of multiple layers, up to four embedded nested layers. Approximately 17% of all entities in the corpus are embedded within another entity. Three types of nesting are identified by Alex et al. (2007): entities containing nested entities (as in the above example); entities which themselves are multiple entity types (e.g. p21ras is both DNA and a protein; this is similar to cases of metonymy in newswire text); and coordinated entities (e.g. human interleukin-2 and -4 referring to both human interleukin-2 and human interleukin-4').

The GENIA scheme is biochemistry specific and none of the nested structures in GENIA occur in the WSJ corpus. Further, there is little discussion about nested named entity schemes and their relationship with syntax for the GENIA corpus for us to base our work on.

2.4.2 AnCora

The AnCora corpus (Carreras et al., 2003) contains both Spanish and Catalan language text. It consists of both a Catalan corpus (AnCora-CA) and a Spanish corpus (AnCora-ES), each of which containing 500,000 words of newswire text. The corpora have been annotated to include a number of layers of annotation, including named entities, syntactic constituents and argument structure. Nesting occurs with *strong* entities being embedded in *weak* entities (corresponding to phrase level constituents), with nearly half of all entities are embedded (Finkel and Manning, 2009c) (see Figure 2.8).

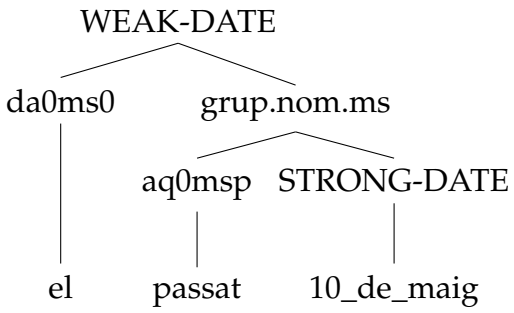


Figure 2.8: Example of nesting from AnCora (Borrega et al., 2007), 'on the 10th of May'.

The named entities annotated in the AnCora corpora are split into 'strong' and 'weak' NEs (Borrega et al., 2007). Strong entities correspond to a linguistic unit with a part of speech tag, and are "a word, a number, a date, or a string of words that refer to a single individual entity in the real world." Strong NEs are split into 8 categories: PERSON, ORGANIZATION, LOCATION, OTHER, ALPHANUMERICAL (NUMBERS), ALPHANUMERICAL (COINS), ALPHANUMERICAL (PERCENTAGES), DATE. Strong entities were analysed as a single element, even if the entity contained multiple tokens.

Weak entities are phrase level syntactic nodes which either contain a strong NE or are a noun phrases which becomes a weak NE due to syntactic, semantic or pragmatic reasons. These weak entities are split into 6 categories: PERSON, ORGANIZATION, LOCATION, OTHER, NUMBERS, DATE.

The decision to mark the prepositional phrases, e.g. on the 10th of May in 2.8, seems inconsistently applied, even within the annotation guidelines described in Borrega et al. (2007). Other prepositional phrases, 'between <location> and <location>' or '[corresponding] to the number 22' are not marked as weak NEs.

Other embedding relationships are lost by the decision to concatenate strong NEs together into one token. For example, the ship named the 'Yellow Pages Endeavour' is tokenised as a single word, losing the reference to the organisation after which it is named. Similarly, 'Premi_dels_Escriptors_Catalans_2003' ('Catalan Writer's Prize, 2003') is not annotated with information about the

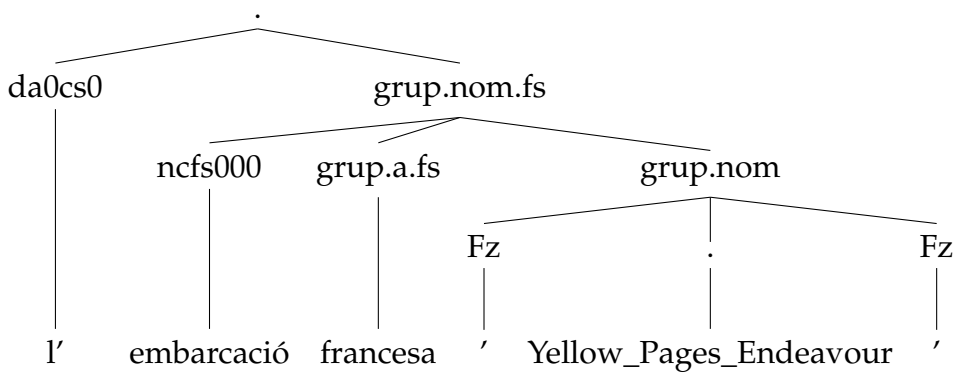


Figure 2.9: Example of nesting from AnCora (Borrega et al., 2007), 'the French ship 'Yellow Pages Endeavour''.

language or the year. This seems like a missed opportunity when other nested references are included. Indeed, the Yellow Pages Endeavour is given in a wider context, as shown in Figure 2.9, where it itself is an embedded entity.

2.4.3 ESTER

The ESTER corpus Galliano et al. (2006) contains 100 hours (approx. 1.2 million words) of orthographically transcribed news broadcast from six French Radio stations, recorded between 1998 and 2004. The corpus also includes an additional 1,677 hours of non-transcribed audio material. Direct mentions of 8 classes of named entities were annotated: AMOUNT, FACILITY, GPE, LOCALIZATION (e.g. geographical areas, addresses), ORGANIZATION, PERSON, PRODUCT (also known as Human Production), TIME, and UNKNOWN (for terms which are supposed to be named entities, but are difficult to classify in one of the other categories). These are further split into a total of 32 sub-categories.

Galibert et al. (2011) built on this work for ESTER II, electing to extend the coverage of the named entities.They added new types of entities (such as FUNCTION and TIME) and present an extended model of named entities which are both hierarchical and compositional Grouin et al. (2011), shown in Table 2.5. The taxonomy is composed of 7 main types, and 32 subtypes, the latter of which adds quantity information (singular vs collective) or precision.

Type	(Detail)	Subtype
Person		pers.ind (individual person), pers.coll (collectivity of persons)
Location	Administrative	loc.adm.town, loc.adm.reg, loc.adm.nat, loc.adm.sup
Location	Physical	loc.phys.geo, loc.phys.hydro, loc.phys.astro
Organization		org.ent (services), org.adm (administration)
Amount		quantity (with unit or general object), duration
Time	date	time.date.abs (absolute date), time.date.rel (relative to discourse)
Time	hour	time.hour.abs (absolute hour), time.hour.rel (relative to discourse)
Production		prod.object (manufactured), prod.art, prod.media, prod.fin (financial), prod.soft (software), prod.award, prod.serv (transportation route), prod.doctr (doctrine), prod.rule (law)
Functions		func.ind (individual functions), func.coll (collectivity of functions)

Table 2.5: Types and Subtypes of entities in the ESTER II corpus.

Additionally, each type also included an 'other' subtype, for those that did not fit into the proposed existing subtypes, and an 'unknown' subtype.

Entities can contain 'components', which are defined as clues that help the annotator make an annotation decision (for example, a first name is a clue that the entity type should be *pers.ind*). Components are considered *internal only* elements, and cannot be used outside the scope of a type or subtype element. Galibert et al. (2011) separate components into two types, 'transverse' components, which can be used in different types of entities, and 'specific' components, which can only be used in one type of element.

Entities can be therefore be nested or compositional, for three reasons: (i) a type contains a component; (ii) a type includes another type, used as a component; or (iii) in cases of metonymy.

For types (i) and (ii), consider the phrase 'nouveau ministre du Budget , François Baroin', shown in Figure 2.10. This is considered two separate entities, one a

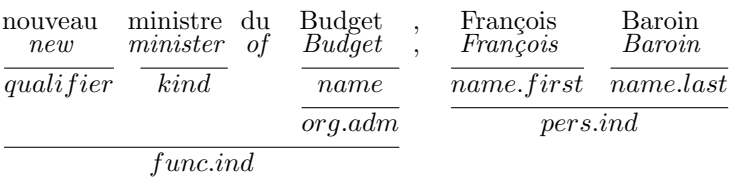


Figure 2.10: Example of Person entity nesting, separate from a function 'role' annotation, in the ESTER II corpus (Galibert et al., 2011).

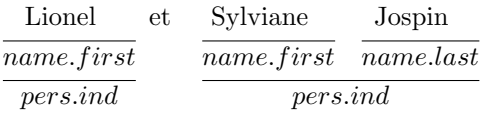


Figure 2.11: Example of coordination not combining separate Person entities in the ESTER II corpus (Galibert et al., 2011).

person, the other a function, with embedded organisational (administrative) entity, which in turn is marked with a component 'name'. All labels closest to the token are components, and additional labels are types.

In this example, the annotation decision to separate *func* from other types such as *pers* is emphasised. The reasons for keeping *func* as a separate type are not widely explained, and it is not clear if all such references to roles are annotated as *func*, even if not immediately followed by a *pers*. Galibert et al. (2011) mention that in future work they may consider folding *func* into a component, which will nest under *pers*.

The boundaries and scope of some entities is also complicated by a decision to exclude relative clauses, subordinate clauses and interpolated clauses; that is, entities must end before these clauses start, or the entity must be split. Similarly, entities are split over coordinated structures, shown in Figure 2.11, thereby unfortunately losing information that could be captured by nested annotations.

Metonymy is also marked explicitly, for example 'la Russie' (Russia) is annotated as *org.adm* when acting as the administration of the country (see Figure 2.12). Recursive cases of embedding can also occur when a subtype includes another named entity annotated with the same subtype. In cases of metonymy,

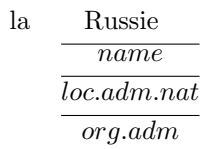


Figure 2.12: Example of metonymy, Russia acting as the organisation, in the ESTER II corpus (Galibert et al., 2011).

the inside label should correspond to the intrinsic type of the entity, and the outer should be the type that corresponds to the result of the metonymy.

2.4.4 Polish National Corpus

The National Corpus of Polish (Przepiórkowski et al., 2010) is an ambitious project aiming to manually annotate 1-million words with various layers of annotation, including named entities, together with a 1 billion word automatically annotated corpus. Named entities are annotated (Savary and Piskorski, 2010) into the categories of PERSON (including subcategories FORENAME, SURNAME, ADDNAME), ORGANIZATION, DATE and TIME are annotated, as well as two distinct types of LOCATION: GEOGNAME and PLACENAME, the latter of which is subcategorised into DISTRICT, SETTLEMENT, REGION, COUNTRY and BLOC. Events, quantities, measures, products and other entities that would fall into a MISCELLANEOUS category, such as Works of Art, are not annotated.

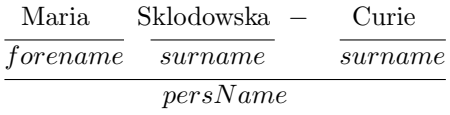


Figure 2.13: Example of Person entity nesting from in the National Corpus of Poland (Savary and Piskorski, 2011).

The decision to split the surname in Figure 2.13 into two separate surnames, thereby essentially annotating on a sub-token level, is unusual. This tokenisa-

Figure 2.14: Examples of nesting in the PKU Chinese corpus (Fu and Luke, 2005): Huizhen Zhanjiang United Christian Hospital, Saint Sophia Cathedral, United Nations Headquarters in New York

tion decision was perhaps influenced in part by the lemmatisation of Polish compound names which poses an additional challenge in the task.

2.4.5 PKU Chinese Corpus

The Peking University Chinese corpus (PKU, Fu and Luke, 2005), contains one month of news texts from the People's Daily, a Chinese language newspaper, manually annotated with 14 types, defined in the IEER-99 Mandarin named entity task: PERSON, CHINESE PERSONAL NAMES, TRANSLITERATED PERSONAL NAMES, LOCATION, ORGANIZATION, OTHER NAMES, DATE, TIME, DURATION, MONEY, MEASURE, PERCENT, CARDINAL, and OTHER NUMBERS. Embedded entities are also annotated, with roughly 18% of entities in the corpus having at least two levels of structure; 2.4% of entities having at least three, and 0.1% of entities having four layers of structure. The PKU corpus contains a total of 106,430 named entities, though in further work by Fu and Luke, they focus on only person names, location names and organisation names, reducing this set to 41,988 entities (additionally excluding a further 1,129 entities which had three or four layers of nesting).

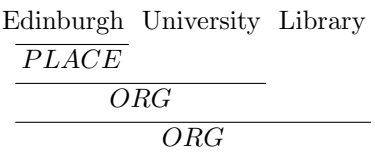


Figure 2.15: Example of nesting in Byrne's Historical Archive Corpus.

2.4.6 Historical Archive Corpus

A subset of the RCAHMS[3] historical archive database has been annotated with entities and relations, as described in Byrne (2007). The annotated corpus is composed of 1,546 documents, drawn from the much larger complete corpus. These documents contain 9,768 text fragments, most of which are in note form, with around 30% estimated to be grammatical English sentences.

The corpus has been annotated with both relation and named entity information, the latter of which is split into 11 classes: ORG, [tgPERSNAME, ROLE, SITETYPE, ARTEFACT, PLACE, SITENAME, ADDRESS, PERIOD, DATE and EVENT. Further subclasses are defined in SITETYPE, ARTEFACT and EVENT. This categorisation grew from a more restrictive earlier set of categories; for example, Location was reanalysed into the finer-grained categories PLACE, SITENAME, and ADDRESS to better accommodate the goal of producing a data querying application for non-specialists in local history.

Up to three levels of nesting are annotated in the corpus, and 9.4% of approximately 27,500 entities were found to have nesting. Unfortunately, the historical archive domain is too specific to be useful for the WSJ newswire text.

2.4.7 KBP EDL 2014

The Entity Discovery and Linking tasks in the Knowledge Base Population track at TAC 2014 (Ji et al., 2014) included an end-to-end English entity discovery

[3]The Royal Commission on the Ancient and Historical Monuments of Scotland http://www.rcahms.gov.uk/.

and linking task, an extension on the previous year's task of linking a given named entity mention to an existing Knowledge Base. The subtask of Entity Discovery & Linking (EDL) involves systems extracting all entity mentions in a document collection, and identifying whether or not those entities are included in a Knowledge Base. For this task, a corpus of 138 documents from a variety of sources, including newswire, discussion forum threads and web documents, with full document entity annotation was created (Ellis et al., 2014). The documents were a subset of those included in the TAC 2014 KBP English Source Corpus (LDC2014E13). Three types of entities were identified: Person, Organization and Geopolitical Entities. Only locations that have a government, a physical location, and a population were considered Geopolitical Entities, and other locations are not marked. Miscellaneous, temporal or numerical expressions were not annotated.

Embedded entities, referred to as 'overlapping mentions' in the annotation guidelines, were annotated only if their boundaries were different, and one was contained completely within the other (Ellis and Getman, 2015). That is, metonymous entities with the same span were not included. Four percent of entity mentions annotated included nesting, for example: [[Kentucky] Fried Chicken] and [[Kurdistan] Freedom Fighters].

The annotation contains some idiosyncrasies, most of which stem from the specific requirements of the entity linking task. All top-level governments of GPEs should also be categorized as GPEs, not as ORGs, regardless of their actual use in the text. Regions such as the southeast US should not be classified as GPEs because, though they have both physical location and population qualities, they do not have their own government. Thus, given the text southeast Texas, only Texas could be annotated as GPE, as southeast Texas has neither its own government nor a defined location. Additionally, while adjectival mentions of GPEs are marked as named mentions of GPEs (for instance, Canadian from the

string Canadian Hockey League), demonyms are not considered named mentions of their respective GPEs. For instance, Americans is not a mention of the United States. It is unclear from the guidelines how instances of metonymy are handled. A further quirk of the data is that all quoted text found in web documents or forum threads should be left unannotated. This is perhaps to avoid the repetition of entities that could happen if a specific post is quoted multiple times in the same thread. Nevertheless, the decision to keep some text unannotated seems at odds with the rest of the annotation guidelines, and the general principle of annotating all instances of entities in data.

2.4.8 Further NER corpora annotation schemas

As seen in Section 2.2, the set of categories with which corpora are annotated varies widely. In addition to the various annotation schemas outlined above, which range from around 7 to 64 categories, other entity schemes and hierarchies have been developed. Some of these are born of necessity (when annotating biomedical data, the entities of interest are substantially different to when annotating newswire text), while others are founded on different annotation principles or goals.

Fine-grained entity types allow for more useful categories when used in end tasks such as question answering, summarisation or other information retrieval systems. Fine grained annotation schemas also have benefits with respect to having greater control over the data at a later stage. A corpus annotated with fine-grained categories is then able to be down-mapped to fit in with other entity categorisation schemes. For instance, specific types of facilities may be treated as Location, Organization or even Miscellaneous entities by different evaluation schemes. Delaying this broader categorisation decision both simplifies annotation decisions (should a hotel be considered a location or organisation), and allows data to be more useful in a variety of different tasks.

2.4.8.1 Sekine's Extended Named Entities

Sekine et al. (2002) outlines a detailed hierarchy of fine-grained entities that were developed to meet the increasing need for a wider range of NE types. The hierarchy originates from the first named entity set defined by MUC, substantially extending the categories to over 200 NE types. The top two hierarchy tiers are outlined in Table 2.6.

The Extended Named Entity hierarchy was developed by combining three initial, separate hierarchies:

- manually-annotated capitalised words from newspaper articles;
- the top-level hierarchies in WordNet and Roget Thesaurus; and
- existing systems and task definitions, especially the TREC-QA[4] task.

These were combined in consultation with some of the designers of each initial hierarchy, and the resulting combination was then refined.

Sekine's hierarchy does not include any nesting or structure, which has resulted in entity boundaries that are not always clearly defined. For example, Elizabeth II of the United Kingdom is listed as a single PERSON entity in the online guidelines[5]. The hierarchy includes some categories which are very specific (e.g. CABINET for political cabinets such as Thatcher's Cabinet, SPA for hot spring resorts), and other categories which seem to contain a mix of common nouns and proper nouns (e.g. PRODUCT:FOOD OTHER includes Coca Cola and Guinness beer as well as water, rice and pork).

While Sekine's annotation guidelines contain examples for each category, these examples are all typical, recognisable entity examples, such that they can be understood out of context. More ambiguous examples, including edge cases that can only be understood with context, are particularly valuable in practical

[4] http://trec.nist.gov/data/qa.html
[5] http://nlp.cs.nyu.edu/ene/version7_1_0Beng.html

annotation guidelines, but these are not provided. Sekine's extended hierarchy guidelines do provide other valuable data, such as a list of attributes for each entity type.

Although no English corpus has been annotated with Sekine's Extended Named Entity hierarchy, Hashimoto et al. (2008) have done so with a Japanese language corpus consisting of 8584 articles (31 days) of Mainichi newspaper, and 400 government white papers. The Mainichi newspaper section contains a total of 252,763 entities (79,632 unique) and the white paper section contains a total of 74,203 entities (23,857 unique). As yet, no NER results on this corpus have been published.

2.4.8.2 TIMEBANK and TimeML

TIMEBANK (Pustejovsky et al., 2003) is a corpus of 68,555 tokens which is annotated with 7571 *events*, 1423 *times*, and 2212 *signal* relations holding between events and times. TimeML, the annotation scheme used to annotate the corpus, defines times as either points, intervals or durations which may be referred to by fully specified temporal expressions (e.g. June 11, 1989), underspecified temporal expressions (e.g. Monday), intentionally specified expressions (e.g. last week) and duration expressions (e.g. three months).

The time entities in TIMEBANK are further annotated with a *function in document* attribute, which provides a temporal anchor (e.g. the date of a newspaper article), the *type* of entity (either date, time or duration), and the *value* of the temporal expression, (e.g. June 11, 1989 would contain value="1989-06-11"). TIMEBANK also annotates temporal relations and the linking of events and temporal expressions.

The annotations are not nested or structured. However, related spans of *time* and *signal* do interact to further specify the events and time entities. For

Person	-
God	-
Organization	International Organization Show Organization Family Ethnic Group Sports Organization Corporation Political Organization Other
Location	Spa Geo-Political Entity (GPE) Region Geological Region Astral Body Address Other
Facility	Facility Part Archaeological Place Geological and Organizational Entity (GOE) Line Other
Product	Material Clothing Money Form Drug Weapon Stock Award Decoration Offence Service Class Character ID Number Vehicle Food Art Printing Doctrine Method Rule Title Language Unit Other
Event	Occasion Incident Natural Phenomenon Other
Natural Object	Element Compound Mineral Living Thing Living Thing Part Other
Disease	Animal Disease Other
Color	Nature Color Other
Other	-

Table 2.6: Top two levels of Sekine's Extended Named Entity hierarchy.

example, a minute and a half is annotated as a DURATION, and linked to adjacent token later, which is marked as a signal conveying *after* as a relation.

2.4.9 NER corpora summary

Although there are a variety of corpora annotated with embedded entities, no English language corpora exist which have detailed, extensive annotation of the internal structures of nested entities of interest in standard newswire text. The differences in annotation schemas across all corpora, from very coarse-grained entities to detailed subtypes further complicate the overall picture of available resources.

The analysis of coordinated named entities, for example, shows shortcomings in annotation schemas, where information about entities is lost. A good analysis of entities such as 'Bill and Melinda Gates' should result in capturing both 'Bill Gates' and 'Melinda Gates', identifying both as Person entities. Similarly, analysing references to entities only nested in nominal forms, rather than inside all entities, does not apply a consistent approach to what is an important task warranting further research efforts. Syntactic elements are affected by, and affect semantic components of text. This thesis addresses the shortcomings in current named entity corpora by creating a detailed and thorough analysis of both the semantic and syntactic structures of named entities.

2.5 Summary

There is little doubt that linguistically competent people make inferences from NE structure. They infer the type of an entity from its structure, but they also infer things about an entity's history or nature from the entities embedded in its name. Occasionally, these assumptions are false, only historically true, or

ambiguous, as with etymology. Nevertheless, NLP systems may be able to benefit from understanding the embedding structure of names.

In the context of GENIA and biomedical tasks, the motivation for the analysis of nested named entities is clear: nested NE annotations are used in broader information extraction annotations and tasks in which embedded entities appear to be referential and have semantic roles.

Wide interest in the concept of nested named entities in newswire text has resulted in a number of disjointed attempts at their analysis and annotation. These piecemeal attempts at capturing the structure of named entities have not resulted in either a canonical system or large, useful corpus with extensive annotations of nested named entities across entity types. This thesis addresses these shortcomings by formalising the definition and evaluation of nested named entities in English newswire text, investigating the effects of including named entity structures in a standard parsing task, offers novel methods of modifying the representation of nested named entity structure for learning and evaluation using existing NER systems, and offers the first results of combined nested named entity recognition and parsing in English newswire text.

3 Nested Entity Annotation Scheme

> "Data is a precious thing and will last longer than the systems themselves."
>
> Tim Berners-Lee

The work in this thesis is motivated by shortcomings in the available corpora with respect to nested named entity annotations in the newswire domain. This lack of available English-language corpora annotated with nested named entity structure and syntactic structure in a compatible form has substantially constrained the development of the field, preventing considerable research in nested named entity recognition, as well as research in joint learning methods.

This chapter describes the process of creating a corpus annotated with gold-standard nested named entities. This data will be used throughout the thesis, in merged variants with the Penn Treebank in Chapter 5, in parsing experiments in Chapter 6 and in named entity recognition experiments in Chapter 7.

We chose to augment the Wall Street Journal (WSJ) section of the Penn Treebank with nested named entity information, building on Vadas and Curran's (2007) addition of noun phrase structure. We chose to add additional annotations to the WSJ since it is most widely-used corpus in the field of parsing English text.

We also decided to make use of the BBN annotation (see Section 2.3.4) thereof. While the BBN annotations are not nested, and while some categories

are problematic, we can use the annotations as a valuable starting point for our own annotations.

Starting with a set of existing annotations has a number of benefits. We were able to leverage the existing annotations in the corpus and automatically introduce structure in the form of pre-annotation, described in Section 4.1.2. This allowed us to, for example, add highly consistent PERSON structure heuristically (i.e. first, middle and family names as well as initials on the basis of ordering and surface form) knowing that most of the time the labels would be correct. All sentences were manually annotated later, allowing annotators to verify and tweak these entities if necessary. Having the structure of very predictable entity types already annotated, however, makes the annotation process much faster and more reliable.

The existing top-level annotations also allowed us to calculate statistics on the numbers and types of entities in the corpus, and identify examples before we committed to an annotation scheme. This was particularly valuable for rarer categories, as it was a straightforward process of identifying examples, rather than an low-recall and time consuming process of finding them. Having an existing top-level annotation scheme for the corpus also allowed us the advantage of having BBN's final decisions about category distinctions, which informed our entity scheme.

3.1 Annotation Principles

For syntax, a large body of work has been done on devising detailed annotation schemes and corresponding annotation guidelines, such as the PTB annotation scheme. For NEs, however, even though there has been considerable work on the development of detailed schemes (Sekine et al., 2002), there has not been

substantial work on a detailed set of annotation guidelines to support those schemes, and associated annotation principles.

We use the following general principles when annotating nested named entities in the corpus.

Principle 1: Annotate all named entities, TIMEX and NUMEX entities We aim to annotate all non-sentence initial words in title or upper case. We also annotate instances of proper noun mentions that are not capitalised, and lower-case numerical and temporal expressions. In the case of entities that sit on the border of proper versus common noun, e.g. chemical names, we are guided by their capitalisation across the corpus.

Principle 2: Annotate all nested structures This principle is the core of the thesis: named entities have nested internal structure and we add annotation layers for these structural elements of entities. These elements could be other entities, [San Francisco]$_{\text{CITY}}$ in [[San Francisco]$_{\text{CITY}}$ International Airport]$_{\text{AIRPORT}}$, or structural components such as UNIT or CARDINAL substructures in [[$]$_{\text{UNIT}}$ [[3]$_{\text{CD}}$ [million]$_{\text{MULT}}$]$_{\text{CD}}$]$_{\text{MONEY}}$. This also includes internal structure induced by syntactic elements, such as coordination (see below).

Principle 3: Add consistent substructure to avoid spurious ambiguity Previous, flat annotation tasks have discarded entity structure, forcing a case of false ambiguity on a token level whereby a token in some situations is labelled a CITY, and at other times may be labelled as an ORG:EDU or SPORTS-TEAM.

We add layers of annotation to allow each token to be annotated as consistently as possible. For example, [Tokyo]$_{\text{CITY}}$ is a CITY even in the entity [[Tokyo]$_{\text{CITY}}$ Giants]$_{\text{TEAM}}$, and [four]$_{\text{CARD}}$ is a CARDINAL even in [[Four]$_{\text{CARD}}$ Seasons]$_{\text{HOTEL}}$. This removes the false ambiguity forced onto tokens which occur in different types of entities.

Principle 4: Unary stacking principle – metonymy We use the principle of unary stacking to capture metonymy while maintaining the *Add consistent substructure principle*. That is, even in cases where there are no other words forming part of the entity, we maintain the stacking principle, nesting one entity label inside another.

In some of these instances, the ambiguity is caused by elided words that would otherwise disambiguate the ambiguous mention. For instance, in He smoked [[Toronto]$_{\text{CITY}}$]$_{\text{TEAM}}$ in the playoffs... and Share prices closed sharply higher in [[New York]$_{\text{CITY}}$]$_{\text{CORP}}$ and [[Toronto]$_{\text{CITY}}$]$_{\text{CORP}}$..., Toronto would be completely disambiguated within the full name of the sports team or the stock exchange.

In other instances, such as The [[White House]$_{\text{BUILDING}}$]$_{\text{GOV}}$ said..., the mention and the metonymy is complete. In both types, only the broader context is available to determine both the literal and metonymic interpretation of the mention.

Principle 5: Underspecify to avoid arbitrary decisions In cases where an entity is genuinely ambiguous and difficult for annotators to resolve in the majority of case, we use deliberately underspecified categories to capture this ambiguity, as with the categories CITY-STATE and MEDIA.

Consider Singapore, which is both country and city at the same time – it is extremely difficult to distinguish between the uses in many cases. Preferring one by default implies a greater confidence than the annotators have in practice. Similarly, the names of media artefacts, such as a newspaper or broadcasting channel, are often shared with the name of the organisation that runs it. These are also often difficult to distinguish.

Rather than attempt to resolve these ambiguities and resorting to arbitrary decisions, we have deliberately created underspecified MEDIA and CITY-STATE categories.

Principle 6: Overspecify to avoid category confusion Some entities are easy to identify, but difficult to categorise consistently. For instance, a hotel (and any business at a fixed location) has both organisational and locative qualities, or is at least treated metonymously as a location.

Rather than requiring annotators to remember an ambiguous categorisation decision, that may not actually fit the given context, we elect to add additional categories to simplify the individual annotation decision: annotating the [Westin]$_{\text{HOTEL}}$ correctly is a simpler task than remembering if HOTEL should be categorised as an CORP, BUILDING, or some sort of LOCATION.

The principle is related to underspecifying to avoid arbitrary decisions.

Principle 7: Pragmatic annotation Many annotation decisions are ambiguous and difficult – especially the internal structure. To correctly identify and classify all entities would require substantial research and be prohibitively expensive. For instance, knowing that [The [Boeing]$_{\text{NAME}}$ Company]$_{\text{CORP}}$ was named after founder [[William]$_{\text{FIRST}}$ [E.]$_{\text{INI}}$ [Boeing]$_{\text{NAME}}$]$_{\text{PER}}$ would potentially allow us to annotate [The [Boeing]$_{\text{PER}}$ Company]$_{\text{CORP}}$ with an embedded PERSON entity. The [Sony Corporation]$_{\text{CORP}}$, on the other hand, was not named after a founder, an imagined [[Mr.]$_{\text{HON}}$ [Sony]$_{\text{NAME}}$]$_{\text{PER}}$, though this is not evident from surface form alone.

To determine the correct etymology of every organisation, substantial research would be required, and in some cases, due to organisations or records no longer existing, this might be impossible. We therefore elect to label all tokens that seem to be the names of people as NAME, regardless of whether they are

actually a person's name, and without doing the requisite and prohibitively expensive research.

The broader principle is that annotation decisions should be made without reference to external documents or research, except perhaps to learn about the structure of a whole class of entities. The NAME category allows us to identify some internal structure in organisations without the expense of having to commit to their type.

Principle 8: Defer to Penn Treebank bracketing Since we will be using the resulting annotations with the syntactic annotation of the Penn Treebank, we follow the general policy of avoiding altering the original Penn Treebank annotations as far as possible, without being constrained by them when necessary.

Therefore, when faced with ambiguous bracketing decisions, for instance the bracketing of two weeks of June as either ((two weeks) of June) or (two (weeks of June)), we favour the analysis that does not conflict with the PTB.

Principle 9: Annotate what cannot be automated We follow the principle of annotating things which cannot be automated later. By adding additional information that can be easily changed or collapsed later, we ensure the corpus is as robust to multiple uses as possible. Similar to the *underspecify to avoid arbitrary decisions* principle, we add categories for differences which are not necessarily intended as robust categories, but which reflect differences which require manual annotation. Consider FIRST and MIDDLE names, a distinction which from surface form alone, is questionable. Some people have multiple first names, others have multiple middle names. Collapsing these categories together would be far easier than attempting to separate them.

Similarly, while the distinction between CITY and STATE may not be critical for all downstream tasks, we distinguish between the two. With some mentions, a post-process dictionary lookup (or similar) could determine that London is

a CITY and Texas is a STATE, but with other cases, such as Washington or New York, the distinction requires contextual understanding.

Principle 10: Monitor the evolving semantic grammar We identify common patterns that capture a semantic grammar of our nested entities. This grammar indicates how the mentions interact to generate one type of entity from another. For example, FIRST + NAME → PER is a common template, as is NUMDAY + MONTH + YEAR → DATE.

We use these rules or templates as guides throughout the annotation process. Try to keep the semantic grammar as small and tight as possible, so that when we have edge cases, we try to fall within one of the frequent rules we have in the semantic grammar. We actively inspect the grammar as we annotate, and as much as possible try to satisfy the existing grammar when we come across an unusual construction.

Principle 11: Coordination principle We join coordinated substructures under a larger span of the same type to correctly capture dependencies. For example, we want to be able to resolve both [[Bill]$_{\text{FIRST}}$ [Gates]$_{\text{NAME}}$]$_{\text{PER}}$ and [[Melinda]$_{\text{FIRST}}$ [Gates]$_{\text{NAME}}$]$_{\text{PER}}$ from the first entity below, and [[Mr]$_{\text{HON}}$ [Bush]$_{\text{NAME}}$]$_{\text{PER}}$ and [[Mrs]$_{\text{HON}}$ [Bush]$_{\text{NAME}}$]$_{\text{PER}}$ from the latter.

In coordination cases such as these, there is an implied plural that isn't specifically marked. That is, over the larger FIRST or HON span above in each example, we really are capturing that this now refers to a plural entity. We do not, however, make the category itself plural, although this is achievable programmatically, since it can be derived from the structure and presence of conjunctions.

The ambiguity of conjunctions in entities is a particularly difficult area of NER (Dale and Mazur, 2007). By explicitly annotating coordinated substructures of entities, we aim to better capture both cases of *Name Internal Conjunction* (e.g. Proctor & Gamble) and *Copying Separators* (e.g. Bill and Melinda Gates).

Principle 12: GRP tags to cover ad hoc groups We introduce a new type of category, the *group* (GRP) category, with subtypes GRP:ORG, GRP:LOC and GRP:PER, which capture the semantics of a group of similar entities. These groups are not officially organised, but are known as one collective entity, for example [[Wall Street]$_{\text{STREET}}$]$_{\text{GRP:ORG}}$ traders, [[Third]$_{\text{ORDINAL}}$ World]$_{\text{GRP:LOC}}$ countries, and the [[Rothschilds]$_{\text{NAME}}$]$_{\text{GRP:PER}}$.

3.2 Annotation Scheme

Creating a nested named entity annotation scheme is one of the major contributions of this thesis. We start with the BBN annotations, which we treat as a base layer of outer most annotations. We augment these with some finer-grained categories, and additional structural elements of entities. The underlying annotations, changes to them, and additional elements of the annotation guidelines are discussed below.

3.2.1 Comparison to OntoNotes

Figure 3.1 compares sentences from a sample document with OntoNotes entity annotations to our own entity annotation scheme (outlined in Chapter 3) of the same sample. Note that although this sample occurs in the OntoNotes Release 5.0 annotation guidelines (Weischedel et al., 2010), it is missing a number of annotations that do exist in the actual data release.

OntoNotes Annotation	Our Annotation
Some [U.S.]$_{\text{GPE}}$ allies are complaining that President [Bush]$_{\text{PERSON}}$ is pushing conventional - arms talks too quickly , creating a risk that negotiators will make errors that could affect the security of [Western Europe]$_{\text{LOC}}$ for [years]$_{\text{DATE}}$.	Some [U.S.]$_{\text{NATIONALITY}}$ allies are complaining that [[President]$_{\text{ROLE}}$ [[Bush]$_{\text{NAME}}$]$_{\text{PER}}$]$_{\text{PER}}$ is pushing conventional-arms talks too quickly , creating a risk that negotiators will make errors that could affect the security of [Western [Europe]$_{\text{CONTINENT}}$]$_{\text{REGION}}$ for [years]$_{\text{DURATION}}$.
Concerns about the pace of the [Vienna]$_{\text{GPE}}$ talks – which are aimed at the destruction of [some 100,000]$_{\text{CARDINAL}}$ weapons , as well as major reductions and realignments of troops in central [Europe]$_{\text{LOC}}$ – also are being registered at the [Pentagon]$_{\text{ORG}}$.	Concerns about the pace of the [Vienna]$_{\text{CITY}}$ talks – which are aimed at the destruction of some [100,000]$_{\text{CARDINAL}}$ weapons , as well as major reductions and realignments of troops in [central [Europe]$_{\text{CONTINENT}}$]$_{\text{REGION}}$ – also are being registered at the [[Pentagon]$_{\text{BUILDING}}$]$_{\text{GOV}}$.
Mr. [Bush]$_{\text{PERSON}}$ has called for an agreement by [next September]$_{\text{DATE}}$ at the latest .	[[Mr.]$_{\text{HON}}$ [Bush]$_{\text{NAME}}$]$_{\text{PER}}$ has called for an agreement by [[next]$_{\text{REL}}$ [September]$_{\text{MONTH}}$]$_{\text{DATE}}$ at the latest .
But some [American]$_{\text{NORP}}$ defense officials believe [the North Atlantic Treaty Organization]$_{\text{ORG}}$ should take more time to examine the long - term implications of the options being considered .	But some [American]$_{\text{NATIONALITY}}$ defense officials believe the [[North [Atlantic]$_{\text{OCEAN}}$]$_{\text{NORP:OTHER}}$ Treaty Organization]$_{\text{ORG:OTHER}}$ should take more time to examine the long-term implications of the options being considered .

Table 3.1: Comparison of OntoNotes annotation and our annotation scheme for the same sentences

3.2.2 Comparison to Sekine's Extended Named Entity Hierarchy

We considered using Sekine's Extended Named Entity hierarchy as a basis for our annotation scheme. One option would have been to map the BBN categories, which already exist as annotations on the WSJ corpus. However, this would have required substantial manual checking as the target category would be ambiguous in many cases. Another option would have been to train an NER system on a corpus annotated with Sekine's hierarchy to provide new underlying annotations. However, this was not possible, as no such English language corpus exists. Instead, we use the BBN categories as our basis, and add a number of new categories to them, drawing inspiration from Sekine's hierarchy (the provenance of categories in our final annotation scheme is discussed in Table 3.4).

3.2.3 Underlying BBN annotations

We use the BBN Entity Annotations for Question Answering (see Section 2.3.4, Brunstein, 2002) as the starting point for our annotations. Tables 3.2 and 3.3 show the 30 most frequently occurring non-DESCRIPTOR entities and 10 DESCRIPTOR entity tags in the corpus, which contains a total of 167,263 entities. These tables also show the three most frequent examples of each category.

As outlined in Section 2.3.4, the annotation of the DESCRIPTOR categories label nominal phrases. The head words of NPs that correspond to entity types are annotated with the appropriate Descriptor label. The DESCRIPTOR annotations were added with tasks such as coreference resolution in mind, and, on the whole, are not directly applicable to the task of named entity recognition. Further, the boundaries on these entities are inconsistent, and we elect to remove them from our starting annotations. Similarly, the ANIMAL annotations, which refer almost exclusively to common nouns, were also removed. The

SUBSTANCE annotations were a mix of predominantly common nouns (drug, corn, sugar) as well as the names of elements (copper, gold) and the names of some products (RU-486, AZT, Red Delicious). Since the substantial majority of these were common nouns, SUBSTANCE labels were also removed.

We used BBN as the basis for our annotation scheme, but had to modify it for a number of reasons. Firstly, we needed to add structural elements to the existing BBN categories, such as FIRST or JARGON. While some of these structural elements are straightforward and sit nicely under existing top-level entities, other structural elements interact directly with the annotations that are there, going inside, outside and across existing annotations. These complex nesting elements require substantial changes to the annotation scheme.

Further, the BBN annotation scheme[1] does not have a high level of detail, and especially lacks context detail for entities. The examples that are given are typical, easy cases that can be readily understood without context. These examples do not address how to annotate ambiguous edge cases. These annotation guidelines provide a not dissimilar level of detail to Sekine's Extended Named Entity hierarchy, described in Section 2.4.8.1.

In cases where the existing BBN scheme does not satisfy our set of annotation principles, we have added the categories to avoid arbitrary decision. For example, the CITY-STATE and MEDIA categories were added to address the *Underspecify ambiguous categories principle* principle. By augmenting the BBN annotation scheme with our own categories, including those derived from Sekine's hierarchy, we are able to develop a robust annotation scheme, and in performing the annotation for this, we were also able to improve the data quality of the underlying BBN annotations. In places, the annotation quality of the BBN corpus is not consistently high, and in the process of adding inner and outer layers, we were able to correct issues as we went. We also removed certain

[1] documented only online: https://catalog.ldc.upenn.edu/docs/LDC2005T33/BBN-Types-Subtypes.html

category types that are not named entities as we wanted a stricter definition of NEs.

In addition, using the existing BBN annotations allows us to create our nested corpus much more rapidly, both in terms of annotation speed and the ability to find instances of specific types for designing our schema.

3.2.4 Categories added to underlying BBN annotations

In designing our annotation hierarchy, we add fine-grained entity categories from previous work we have conducted annotating Entities in Wikipedia (Nothman et al., 2013). We extended the category set for four main reasons:

1) Following the *Annotate all nested structures principle*, we include structural categories for annotation. These categories are designed to nest inside other top-level categories.

2) In many cases, categorisation, rather than identification, of entities is problematic. Should the Paris Hilton hotel be marked as an ORG or a LOC, since it can both be found on a map, and have an organisational structure of ownership? Is Glebe Point Road a type of FACILITY or LOC, since it is has a location, but is constructed? Should the New America High Income Fund be considered an ORG or a PRODUCT, since it has an organisational structure in charge of its direction, but is something intangible, that can be invested in. In most cases, ambiguities stem from entities which to a certain extent could fit into multiple categories. Following the *Overspecify ambiguous categories principle*, and marking the above examples as HOTEL, STREET and FUND respectively simplifies the immediate annotation task, and allows us to defer, and even modify and experiment on categorisation decisions at a later date.

Tag	%	Examples
ORG:CORPORATION	14.01%	UAL, New York Stock Exchange, Big Board
DATE:DATE	9.64%	yesterday, Friday, this year
PERSON	8.22%	Bush, Noriega, Reagan
MONEY	6.63%	$ 1 million, $ 1 billion, $ 1,000
CARDINAL	6.17%	one, two, three
PERCENT	3.57%	10 %, 15 %, 50 %
GPE:CITY	3.35%	New York, London, San Francisco
GPE:COUNTRY	3.04%	U.S., Japan, China
ORG:GOVERNMENT	2.77%	Treasury, Congress, Senate
NORP:NATIONALITY	1.94%	Japanese, British, American
DATE:DURATION	1.88%	years, 30-year, six months
GPE:STATE_PROVINCE	1.61%	California, Texas, Calif.
ORG:OTHER	0.75%	OPEC, Merc, Giants
ORDINAL	0.66%	first, second, third
TIME	0.64%	morning, afternoon, night
SUBSTANCE:FOOD	0.53%	corn, food, sugar
SUBSTANCE:OTHER	0.51%	oil, gas, steel
DATE:OTHER	0.47%	annual, daily, quarterly
NORP:POLITICAL	0.40%	D., Democrats, Democratic
DATE:AGE	0.37%	45, 52 years old, 44
SUBSTANCE:CHEMICAL	0.32%	gold, chemicals, copper
LOCATION:REGION	0.31%	Bay Area, Midwest, Eastern Europe
PRODUCT:OTHER	0.31%	486, Cray-3, West Texas Intermediate
WORK_OF_ART:OTHER	0.31%	D.T., Batman, Batibot
SUBSTANCE:DRUG	0.26%	drug, drugs, psyllium
ORG:POLITICAL	0.25%	ANC, GOP, Communist Party
ANIMAL	0.24%	cattle, animals, worm
PRODUCT:VEHICLE	0.23%	Galileo, Atlantis, Scorpio
LAW	0.23%	Chapter 11, RICO, Gramm-Rudman
ORG:EDUCATIONAL	0.22%	Harvard, Yale, Massachusetts Institute of Technology

Table 3.2: The 30 most frequent non-DESCRIPTOR labels in the Wall Street Journal BBN corpus, the percentage of each tag's occurrences, and the three most frequent examples.

Tag	%	Examples
PER_DESC	15.75%	people, investors, president
ORG_DESC:CORPORATION	9.08%	company, companies, unit
ORG_DESC:GOVERNMENT	1.50%	government, court, administration
FAC_DESC:BUILDING	1.08%	plant, plants, building
PRODUCT_DESC:VEHICLE	0.73%	cars, car, auto
ORG_DESC:OTHER	0.71%	group, union, groups
GPE_DESC:COUNTRY	0.59%	country, nation, countries
FAC_DESC:OTHER	0.26%	mine, mines, facilities
GPE_DESC:STATE_PROVINCE	0.24%	state, states, province
GPE_DESC:CITY	0.23%	city, cities, town

Table 3.3: The 10 most frequent DESCRIPTOR labels in the Wall Street Journal BBN corpus, the percentage of each tag's occurrences, and the three most frequent examples.

3) Although the text we are annotating for this corpus is not particularly recent, we want to update the entity scheme for use on modern text as well. In this respect, we draw heavily on our previous work annotating Wikipedia. In practice, this means the addition of categories such as SPORTS-TEAM, SPORTS-SEASON, ALBUM and BAND.

4) A fine-grained, hierarchical category set ensures the corpus is flexible for future uses, allowing for specific categorisation decisions to be changed or downmapped in the future.

Additional structural categories were added to PER to capture the first, middle, last names (FIRST, MIDDLE, NAME), initials (INI), nicknames (NICK-NAME), name modifiers (NAMEMOD) such as Jr. or III and honorific titles (HON) such as Ms. or Sir. An additional category, ROLE, was introduced to capture nominal modifiers that encapsulate vocational titles, such as Professor, Rabbi, or President.

The location category was extended to include the categories of SPACE, SUBURB and CITY-STATE, the latter of which was added based on the *Underspecify ambiguous categories principle* to address legitimate ambiguity when considering entities such as Hong Kong and Singapore. Additionally, some of

the location related entities in the underlying BBN annotations were removed. Specifically, the intended distinction between the common GPE:CITY and relatively infrequent LOCATION:CITY and ORGANIZATION:CITY was unclear, so these categories were amalgamated into one CITY tag. Similarly, ORGANIZATION:STATE_PROVINCE was removed as a category. None of these categories are mentioned in the BBN annotation guidelines.

The ORG category was extended by ARMY, BAND and SPORTS-TEAM, as well as MEDIA (Time, Wall Street Journal). The JARGON (Corp., Inc.) was also added to capture common elements to organisation names that are often elided from the surface form, for instance, [Sony [Corporation]$_{\text{JARGON}}$]$_{\text{CORP}}$ is often referred to as [Sony]$_{\text{CORP}}$. As examples of particularly difficult entities to classify, specific categories were added for indexes (INDEX Dow Jones Industrial Average, Nasdaq Financial Index), and funds (FUND Zenith Income Fund).

The Work of Art (WOA) category was expanded, with the addition of ALBUM, AWARD, FILM, and TV-SHOW. CONCERT, SPORTS-EVENT and NATURAL-DISASTER were added to the existing types of events. NATURAL-DISASTER was specifically added to accommodate events similar to hurricanes, which existed as a category in BBN. The FACILITY category was similarly extended to include STADIUM and STATION. ELECTRONICS (e.g. 80486) and CHANNEL (e.g. The Discovery Channel) were added as specific types of PRODUCT, and SCINAME was added as a category to label scientific names, some of which were capitalised (e.g. Homo sapiens, Bordetella pertussis). The BBN PRODCUT:OTHER[sic] category was also discarded as an error.

ANIMAL was removed, and ANIMATE added in its place to label all animate entities that are not PER. A separate GOD category is created, which mostly contains the tokens God and Messiah. This distinct GOD category is an example of a case where it is not always clear which broader category the entity should be labelled with (idiomatic expressions found in the corpus, such

as Oh my God... or act of God illustrate this). By creating a separate entity, albeit one with relatively few occurrences in the corpus, categorisation, rather than classification, decisions can be deferred until later, and easily changed according to the application of the annotations.

The NUMEX and TIMEX categories were extended to capture the structure of temporal and numerical expressions. DAY, MONTH, NUMDAY (18th), SEASON (autumn), YEAR and PERIODIC (annual) as well as SPORTS-SEASON were added for temporal expressions. SPORTS-SEASON was specifically added based on our previous experience annotating Wikipedia, which contained frequent mentions such as 1993 New York Yankees season. The category was not frequently used in the PTB, however, but is retained and can be remapped to other applicable categories, such as DURATION. The TIMEX category was extended with MULT (billions), UNIT ($, %, tons), FOLD (double), and RATE (five cents a share). IPOINTS (190-point) was also added as a specific category to combat confusion about index points. REL, which grounds an entity *relative* to another date (last, next), and QUAL which modifies a numerical value (about, more than) were also added as structural elements.

To follow the *Ad hoc groups principle*, and capture unorganised groups of entities, specifically locations, organisations and people, *Group* categories were added: GRP:LOC (Third World), GRP:ORG (Wall Street, Ivy League), GRP:PER (Rothschilds, Old Guard).

Table 3.4 outlines the final annotation scheme, including the number of instances of each category in the corpus, and the provenance of each category. Some of the categories added, marked as *new cat* in provenance in Table 3.4, were inspired by categories from our own previous Wikipedia work, which in turn was inspired by both the BBN and Sekine Extended Named Entity hierarchy schemes. Other categories we added based on inconsistencies in an existing scheme. For example, ANIMATE and MEDIA address deficiencies in

the BBN scheme, described further below. We also added specific categories to follow our annotation principles, for example, CITY-STATE to maintain ambiguity with a specific issue, and the novel GROUP categories. Additionally, new categories designed to capture the structure of entities are marked as struct.

Category	Frequency	Provenance	Category	Frequency	Provenance
PER	14926	struct	NATIONALITY	5194	BBN
FIRST	6796	struct	NORP:POLITICAL	731	BBN
INI	1445	struct	RELIGION	99	BBN
NAME	28540	struct	NORP:OTH	1247	BBN
MIDDLE	313	struct	SPORTS-EVENT	100	new cat
NAMEMOD	155	struct	CONCERT	2	new cat
NICKNAME	96	struct	HURRICANE	107	BBN
HON	5524	struct	NATURAL-DISASTER	2	new cat
ROLE	2215	struct	WAR	51	BBN
ANIMATE	29	BBN fix	SPORTS-SEASON	8	new cat
CORP	23347	BBN	EVENT:OTH	265	BBN
ORG:EDU	411	BBN	LANGUAGE	92	BBN
ORG:POLITICAL	434	BBN	LAW	419	BBN
ORG:RELIGIOUS	35	BBN	AWARD	37	new cat
GOVERNMENT	4671	BBN	ELECTRONICS	167	new cat
ARMY	139	new cat	PRODUCT:DRUG	116	BBN
BAND	10	new cat	PRODUCT:FOOD	80	BBN
SPORTS-TEAM	166	new cat	VEHICLE	432	BBN
MEDIA	1712	BBN fix	WEAPON	23	BBN
INDEX	657	new cat	DISEASE	246	BBN
FUND	54	new cat	GOD	29	new cat
HOTEL	55	BBN	SCINAME	7	new cat
HOSPITAL	25	BBN	PROD:OTH	656	BBN
MUSEUM	17	BBN	GRP:ORG	437	principle
ORG:OTH	1095	BBN	GRP:LOC	63	principle
JARGON	5561	struct	GRP:PER	154	principle
AIRPORT	32	BBN	CARDINAL	43807	BBN
ATTRACTION	24	BBN	FOLD	313	struct
BRIDGE	44	BBN	ENERGY	17	BBN
BUILDING	346	BBN	IPOINTS	2399	new cat
STADIUM	37	new cat	MONEY	12659	BBN
STATION	1	new cat	MULT	7852	struct
STREET	475	BBN	ORDINAL	2590	BBN
FACILITY:OTH	129	BBN	PERCENT	6541	BBN
CITY	6723	BBN	QUANTITY:1D	221	BBN
SUBURB	78	new cat	QUANTITY:2D	81	BBN
STATE	3245	BBN	QUANTITY:3D	156	BBN
GPE	334	BBN	QUANTITY:OTH	55	BBN
COUNTRY	4046	BBN	RATE	2147	struct
CITY-STATE	220	principle	SPEED	14	BBN
CONTINENT	354	BBN	TEMPERATURE	2	BBN
OCEAN	291	BBN	WEIGHT	293	BBN
REGION	865	BBN	QUAL	3904	struct
RIVER	52	BBN	DATE	17476	BBN
SPACE	53	new cat	DURATION	13742	new cat
LOCATION:OTH	261	BBN	PERIODIC	1066	new cat
ALBUM	3	new cat	AGE	661	BBN
BOOK	148	BBN	DAY	1631	struct
FILM	89	new cat	MONTH	3386	struct
PAINTING	13	BBN	NUMDAY	1495	struct
PLAY	42	BBN	REL	6170	struct
SONG	54	BBN	SEASON	337	BBN
TV-SHOW	172	new cat	TIME	296	BBN
WOA	207	BBN	YEAR	3421	new cat
			DATE:OTH	164	BBN

Table 3.4: Category overview of annotation scheme, including number of instances in the final annotated corpus, and provenance of the category. Ordering in this table follows the hierarchy used in the annotation guidelines which follow.

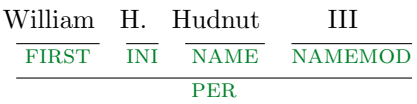
3.3 Annotation Guidelines Summary

This section includes an overview of the schema used in annotation. The full annotation guidelines include further details on edge-cases and problem cases. The choice of label name for each category was influenced by an attempt to select a short and unambiguous prefix for the annotation tool.

3.3.1 PER

PERSON entities can be referred to in a variety of ways. In addition to reducing token-level ambiguity, the intention of marking the substructures of PERSON entities is to aid in coreference resolution. In newswire, initial mentions of people frequently include both their first and last names, with subsequent mentions only referring to their last names.

Following the *Pragmatic annotation principle*, due to the difficulty in distinguishing between last names and invented names, especially in the names of organisations, we mark all things that look like names as NAME. See the discussion in Section 3.3.2.4.

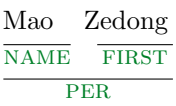


PERSON We mark up all instances of people, fictional characters, first names, last names, nicknames (if referential).

FIRST People's first names, e.g. Sophia, Nicky, Sam, Hugo, Kellie. Note, it is often difficult to distinguish first from middle names (e.g. Mary Beth Smith may use Mary Beth as her first name). Unless clear from the context that both should be considered FIRST, mark the second name up as a MIDDLE.

INITIALS R., M.J., T.S. INITIALS should be marked whether they occur in what would usually be a FIRST or MIDDLE name position.

NAME The NAME tag is used to annotate last names.
Smith, Jones, Di Marco, Gelber Note: with naming conventions of other cultures, apply first/last according to use. E.g., if a Chinese name is written with the family name first, then mark them as:



such that we can give the correct markup given the tokens:

For further discussion of NAME, especially in ORG entities, see Section 3.3.2.1.

MIDDLE This category contains the same type of tokens as FIRST. We elected to annotate these as MIDDLE names, rather than as additional FIRST names, to approach prevailing English naming conventions. FIRST and MIDDLE can be mapped to the same category, but separating them from a single category is not as straightforward, and impossible without manual annotation. We therefore follow the *Manual annotation principle*, and annotate the two categories separately. For cases where it is unclear whether a name is part of a double barrel first name (Mary Lou) or a middle name, it is marked as a middle name.

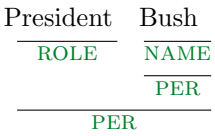
NAMEMOD Name Modifiers such as Jr., Senior, III are marked up.

NICKNAME Nicknames are annotated, excluding any quotations surrounding the nickname, if present. Bobby, The Baboon, Ruddbot, Little Tramp

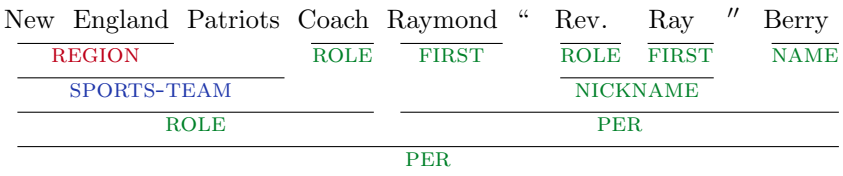
HON In this category we annotate honorific titles including Mr , Mrs, Ms, Miss, Messrs.,, Sir, Madam, Saint, Lord, Lady. These honorific titles differ from ROLE in that they do not offer any information as to the profession job or other role of the person.

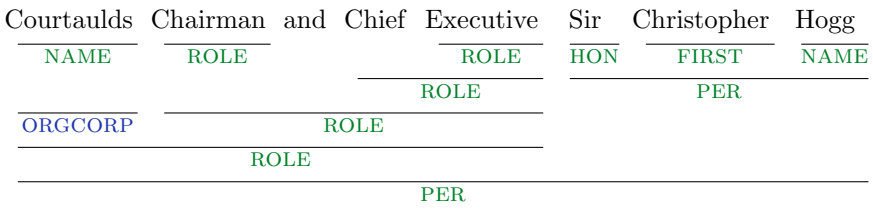
ROLE A person's ROLE is often used as part of the entity span referring to them. In the ROLE category, we annotate titles based on vocation, and embed them in the larger PER span. Only tokens that are capitalised should be annotated, for instance, do not mark up president in president Smith of Company Name.

ROLE includes professions (Dr, Professor, Prof., President, Prime Minister, Secretary of State, Attorney General, Foreign Ministers etc., Senator, Representative, Judge), religious titles (Father, Rabbi), military titles (Admiral), and government titles (the Honorable, His/Her Royal Highness, Prince, Princess).

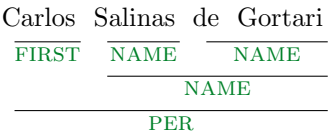


Many ROLE structures are quite complex, with organisations embedded within them. To match the Treebank annotation and preserve linguistically motivated constituent spans, we annotate these structures as one larger entity, rather than two disjoint ones:

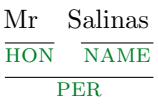




International Names We try to follow international naming conventions where possible. For example, Colombian family names traditionally are double barrel, taking one 'last name' from each parent.

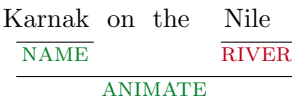


Carlos Salinas de Gortari is referred to a few sentences later as Mr Salinas.



ANIMATE This is a category for other animate, non-human entities, including racehorses, pets and fictional animals, e.g. Bugs Bunny, Phar Lap.

Note that the names of racehorses can be particularly unusual constructions:



3.3.2 ORG

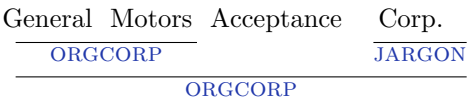
Organisations have interesting structures, often including embedded references to people and locations, as well as having structural elements denoting specific legal organisational meanings (Pty Ltd., Corp.).

ORG:CORP Corporate organisations: companies authorised to act as a single entity. Corporations can be simple flat structures, including multi-word entities, and their abbreviated forms. CORP is used to refer to ORG:CORP in this thesis.

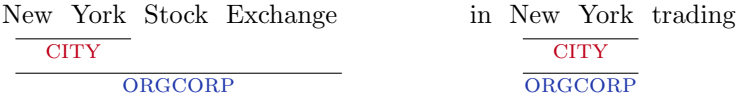
Corporation names frequently refer to their founders; these nested names are marked with the NAME tag. In the following examples, Monsanto is named after founder John Queeny's wife, Olga Mendez Monsanto, and Boeing is named after William E. Boeing.

Organisations frequently have embedded references to locations, including CITY and COUNTRY, and NATIONALITY.

Organisations can also include references to other organisations, as occurs frequently in the case of parent-subsidiary relationships:

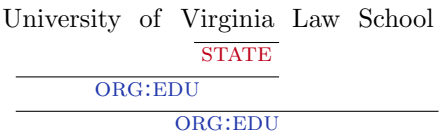


Stock Exchanges Stock exchanges are often referred to by the city in which they are located. For example, the [New York Stock Exchange] is implied by the words In New York trading. In order to both maintain a consistent 'per token' level analysis and follow the *Add consistent substructure principle*, these should be marked as LOC nested within the actual CORP meaning of the words in this context.



ORG:EDU Schools, universities and other educational organisations are marked as ORG:EDU. A particularly common template for these entities includes the CITY in which the ORG:EDU is located:

Universities, as large educational organisations, frequently have smaller educational organisations within them, such as schools and faculties, and this structure is reflected in the structure of the entity itself.



ORG:POLITICAL The names of political parties, which themselves often include an adjectival (NORP) reference to the political party itself.

ORG:RELIGIOUS The names of religious organisations which, for example, run places of worship or schools. Note that adjectival forms of religious organisations, e.g. Christian or Muslim, are a type of NORP, and should be marked as RELIGION.

ORG:RELIGIOUS, ORG:POLITICAL AND ORG:EDU maintain the ORG prefix to easily distinguish between similarly named categories (e.g. RELIGION, NORP:POLITICAL).

GOVERNMENT These are non-political organisational units within countries, excluding ARMY, down to small county, council and municipal levels. This includes non-generic governmental entity names such as Congress or 'Chamber of Deputies. Also mark up entities that are government controlled, such as NASA.

Treasury Department — GOVERNMENT

State Commission on Judicial Conduct — GOVERNMENT

South Australian Treasury — NORP:OTHER / GOVERNMENT

White House — BUILDING / GOVERNMENT

Office of Management and Budget — GOVERNMENT

White House Office of Management and Budget — GOVERNMENT

ARMY The names of any military (including army, navy, airforce etc.) unit. These should be annotated with full structure of LOC, NATIONALITY etc.

Air Force — ARMY

Navy — ARMY

U.S. — NATIONALITY

U.S. Air Force — ARMY

BAND The name of a musical group. Note: when referring to individual artists, e.g.: Jimi Hendrix, these entities should be marked as PER.

Beatles — BAND

Pet Shop Boys — BAND

SPORTS-TEAM Mentions of a sports team. SPORTS-TEAM entities should be marked up with embedded entities, most frequent of which are CITY and other LOC entities.

San Francisco — CITY

San Francisco Giants — SPORTS-TEAM

Giants — SPORTS-TEAM

Toronto — CITY

Toronto Blue Jays — CITY / SPORTS-TEAM

Toronto — CITY / SPORTS-TEAM

Note in the above examples, ['Toronto'], as well as making up part of the '[Toronto] [Blue Jays]' is also used as a stand-alone mention of a SPORTS-TEAM.

MEDIA The MEDIA tag covers entities that are news sources such as newspapers and broadcasters, for which the distinction between the product and the company is often particularly difficult to make.

Organisations that write or run a particular newspaper are frequently named after that newspaper, and similar situations are common for other media providers such as radio stations or TV channels. Further, the distinctions between products and the opinions of people producing those products (which should be treated as the organisation itself) are even harder to separate, as publications often function agentively as organisations.

To capture this ambiguity, we mark instances up as MEDIA, following the *Underspecify ambiguous categories principle*. This category does not attempt to resolve the entity in question to either the physical product (for example, a physical copy of a newspaper, or a radio broadcaster) or the organisation to which it could refer.

the Journal asked ... a random sample of business owners
MEDIA

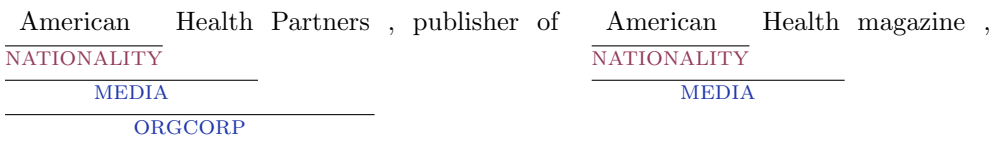
several industry analysts told the Professional Investor Report they believed
MEDIA

We can further use nested structures to identify instances where a MEDIA entity is completely unambiguously referring to the organisation which runs it. For instance, when a MEDIA entity is referred to in the same way as other CORP entities are, such as when discussing the finances of the company, or a recent board election we can nest the MEDIA tag inside a larger CORP span:

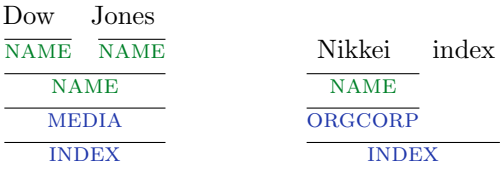
said Robert F. Erburu , Times Mirror 's chairman and chief executive
FIRST INI NAME — PER
MEDIA — ORGCORP

former CBS News President
MEDIA — ORGCORP

That is, MEDIA entities should only be nested in CORP spans when they are being reported on just as any other company would be reporting on. In the following example, the two entities refer to CORP and MEDIA respectively, evident from their contexts:



INDEX The name of a particular stock index. Indexes such as the Nikkei Index which contain references to CORPS should be marked up as INDEX, and have the embedded structure of CORP (and others as appropriate) added.

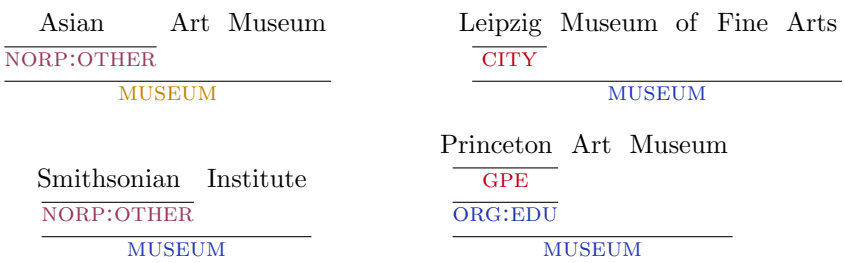


FUND The name of a particular money fund.

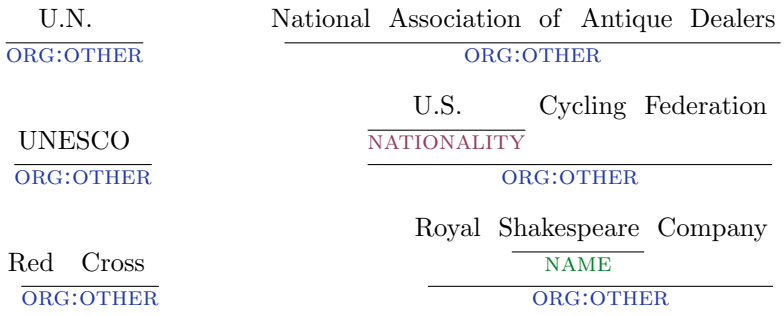
HOTEL The name of a hotel, hostel, or other accommodation provider.

HOSPITAL The name of a hospital, health clinic, or medical facility of any sort.

MUSEUM The name of a museum, gallery etc.



ORG:OTHER This category includes other organisations that are not otherwise covered above, including: libraries, unions, environmental agencies, professional associations, health associations. It also includes governing organisations that sit at a higher level than GOVERNMENT, such as the UN or European Commission, with those at a country level and below covered by GOVERNMENT.



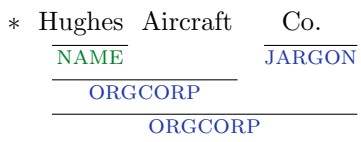
3.3.2.1 Structural elements common in ORG

JARGON The intention of marking up JARGON (and NAME) is to help coreference resolution. Corporate modifiers, such as Corp., Co. LTD., etc. simply modify the name of the organisation, and are not always used. Thus, it is useful to signify that Fujitsu Ltd is referring to the same entity as Fujitsu.

Since we are annotating NAME within ORGs, the distinction between JARGON and words that are not a NAME is an important differentiation.

E.g., when analysing entities such as:

it may be useful to have individual potential alias spans of Hughes, Hughes Aircraft and Hughes Aircraft Co. However, adding embedded CORP spans within an CORP, as in this derivation:

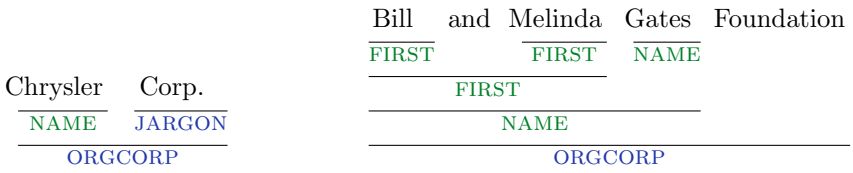


would also imply there is an embedded company reference, which is incorrect. Thus, we resolve only to add internal CORP when a separate entity is being referenced, not denoting potential alias spans.

NAME We use NAME to mark internal structure and nested references to entities that behave as references to people.

Common types of NAME:

- family names: Edison, Gates, Foley, Babcock, Brown
- Specific references to people in an organisation name: Bill and Melinda Gates Foundation, Thomas Edison Corp.



NAME is discussed in more detail below, and with reference to PER entities in Section 3.3.1.

3.3.2.2 NAME in ORG entities

Ideally, we would be able to mark up all embedded references to entities, that is, mark up [Walt Disney] as a PER within the ORG entity [Walt Disney Corp.] This is, however, sometimes not feasible.

In some cases, it is not immediately clear what type an embedded entity is without requiring substantial research. We can identify [Walt Disney] as a person easily, due to his fame.

Familiarity with an organisation does not always ensure an embedded entity's type is widely know, however. Compare [R.P. Scherer Co.], [Boeing Corp.], [Alleghany Corp.] and [Univest Corp.]. Here, [R.P. Scherer Co.] is clearly recognisable as the name of a person, in this case [Robert Pauli Scherer], the company's founder. In the organisation [Boeing Corp.], the name $[\text{Boeing}]_{\text{NAME}}$ also refers to its founder, [William E. Boeing].

The organisation [Alleghany Corp.] could seemingly have been named after a founder, but was founded by railroad entrepreneurs Oris and Mantis Van Sweringen. The name, seemingly, is not a reference to a specific entity, but simply a chosen name. In the case of [Univest Corp.], it seems clearer that the company name does not refer to a person, but instead has to do, perhaps, with banking or investment. Indeed, Univest Corporation is an American corporation offering banking, insurance and investments.

Thus, it is not always clear which elements of an organisation name refer to other embedded entities, and without substantial research into each individual organisation, many of which no loner exist, it is not possible to accurately determine these.

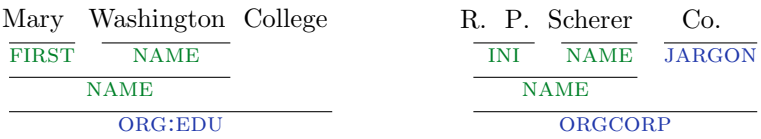
For example, consider Boeing Co.. If we knew with certainty that this company was named for a person with the last name Boeing, we may want to annotate the company in the following way:

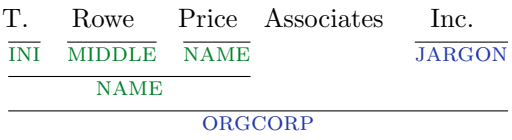
$$\overset{*}{\underset{\overline{\overline{\text{PER}}}^{\overline{\text{LAST}}}}{\text{Boeing}}}\ \underset{\overline{\text{JARGON}}}{\text{Co.}}$$
$$\overline{\text{ORGCORP}}$$

However, for many companies, we cannot determine this with great certainty, especially without a prohibitive amount of research.

On the other hand, other elements of an embedded entity are easier to distinguish, including initials and first names. We follow the *Pragmatic annotation principle*, and take the strategy of marking up all mentions that seem to be reasonably likely to be references to people, marking each likely reference as NAME, with additional structural layers added as necessary for first names and initials.

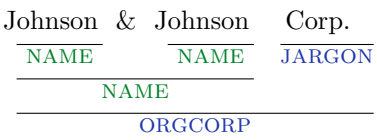
The largest span that refers to the same entity should be marked as a single NAME. If it is not clear whether an embedded reference is referring to one or more entities, we err on the side of caution, marking them as separate entities.





In this way, we can build up standardised structures in organisation names. Consider that the following entities all have structure NAME + JARGON → CORP:

We can expand on these regular patterns in the structure of entities by introducing the concept of syntactic coordination into our analysis.



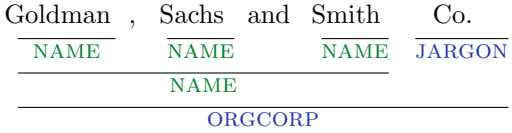
Here, the '&' is acting as a conjunction, combining the two names Johnson and Johnson to act as a single name, following the *Coordination principle*.

3.3.2.3 Company history through name coordinations

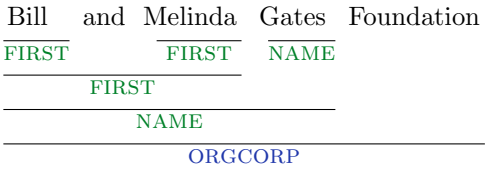
Company names often go through substantial changes as companies grow and merge. For example, the [Goldman Corp.] merged with [Sachs Co.] to produce [Goldman and Sachs Co.], now better known as [Goldman Sachs Co.]

Even in the final [Goldman Sachs Co], an elided conjunction is being used to combine the two originally separate company names into one NAME span. Thus, the structural pattern of NAME + JARGON → CORP still holds.

Similarly, mark up three plus coordinated entities like this:



By annotating embedded structures inside NAME, we can also coordinate on other elements inside the name. For instance:



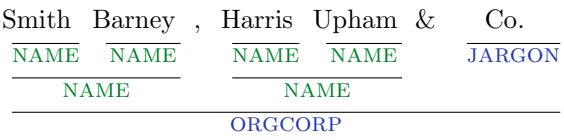
Through the merging of companies, organisation names often grow quite complicated. Take the finance company [Morgan Stanley Smith Barney] as an

example. [Charles D. Barney & Co.] and [Edward B. Smith & Co.], both named after their founders, merged in 1938 to form [Smith Barney & Co.]

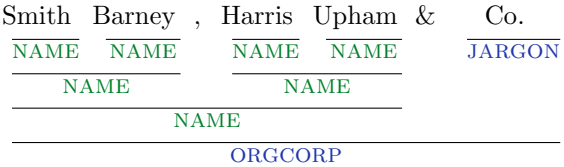
[Smith Barney, Harris Upham & Co.] was, formed in 1975 when [Smith Barney] merged with [Harris, Upham & Co.]. The company continued to acquire other businesses, including [Shearson], and for a time was known as [Travelers Group Inc.], although some part of the business continued to operate under the [Smith Barney] brand. In 1997, [Travelers] acquired [Saloman Inc.], creating [Salomon Smith Barney], before selling to [Morgan Stanley], to become [Morgan Stanley Smith Barney]. Thus, in some cases, the surface structure of an organisation's name often represents part of its history, but names can come and go.

We take the following as a starting point, including the NAME of the founders of the now merged companies.

We also consider the NAME coordination between Smith and Barney, and Harris and Upham from a syntactic perspective, respecting the comma boundaries, and coordinate to the following structure.

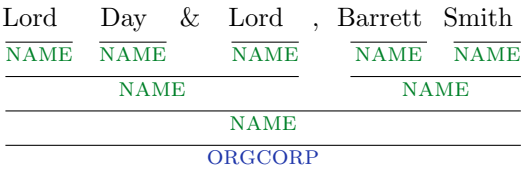


These comma boundaries often act as hints of the historical structure of the company. We then coordinate over the comma to join the distinct NAME labels:



This structure now adheres to the standard template of NAME + JARGON → CORP.

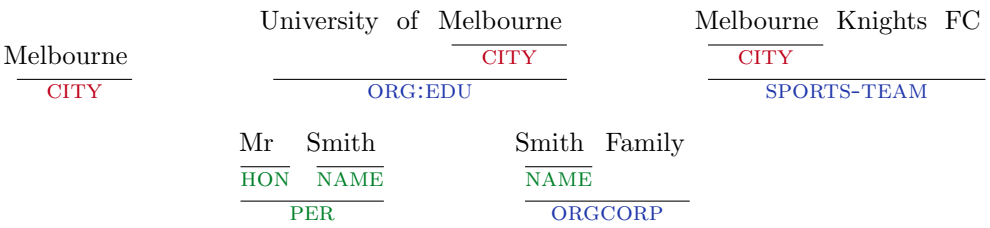
Commas in NAME often designate historical mergers and acquisitions, and this structure is reflected our bracketing. However, commas are unreliable, and are often removed, especially as the name of an organisation becomes more widely known. Where commas exist, we trust that bracketing. In entities that do not have commas, we do not guess at or look up structure.



3.3.2.4 Why NAME and not LAST?

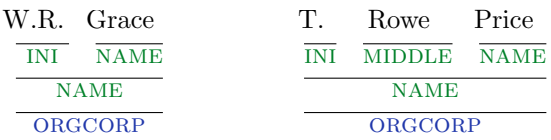
To avoid marking up names that are not surnames as LAST, but also avoid enormous effort researching every organisation named in the corpus, we considered a number of options. We follow the *Pragmatic annotation principle*, and use NAME as an underspecified and general tag, which when contextualised, can take on a more specific meaning.

For our analysis, we want to have as much of the same structure as possible for each word. That is, that Smith should be marked consistently wherever it appears, just as Melbourne is consistently marked as a CITY.



Flat annotation structures force an ambiguous analysis onto words that are not inherently ambiguous. Forcing flat annotations on the examples above would result in the word Melbourne marked as CITY, ORG:EDU and SPORTS-TEAM respectively, when all cases are clearly referring to the CITY. Similarly with names, 'Smith' would otherwise be marked up both as PER and CORP.

NAME as an annotation tag can also contain more than just single names, for instance, when used in CORP, NAME can contain other elements:



3.3.2.5 What's in a NAME?

How 'namey' should a word be in order for it to be considered NAME? The decision to annotate this entity structure as NAME and not LAST is reflective of the greater leniency with which we will annotate these structures. To avoid prohibitively lengthy research for each organisation appearing in the corpus, we adhere to the general policy of marking up tokens which conceivably are last names as NAME. Taken to an extreme, we annotate 'name-like' tokens in the structure of organisations that are not also common words likely to be associated with the business.

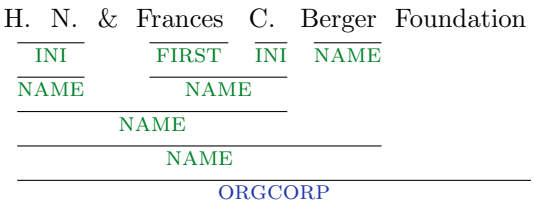
For instance, even though [Sony]$_{\text{ORG}}$ is not, in fact, named after a founder with the same last name, we still mark this up as NAME.

Similarly, [Boeing]$_{\text{CORP}}$ is marked up as NAME, without requiring extensive research to identify that it was, in fact, founded by [William Boeing]$_{\text{PER}}$.

In the case of common nouns being used as part of an organisation name, use common sense to mark Walker & Sons as NAME and not to mark Walker up in an organisation such as Baby Walker Co. In an organisation such as Walker Designs, again following the *Pragmatic annotation principle* principle, we do annotate Walker as NAME. In cases where it is unclear, we default to adding NAME as a label.

3.3.2.6 Edge cases in NAME

Inevitably, some CORP examples will not be able to be correctly analysed. In this case, we try to do the best we can. Take the name H.N. & Frances C. Berger Foundation:



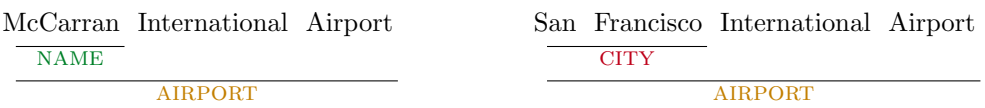
In order to correctly coordinate on the two people, there must be a span of a single given label that is on both sides of the &. For this, the best answer is to use NAME, even if this means that we have a resulting rule of NAME plus NAME combining to make a NAME.

3.3.3 FACILITY

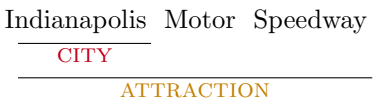
Facilities exhibit both organisation and location qualities. For instance, a hotel may have both a CEO and an address. It can be bought and sold, or opened and closed. As such, facilities should be annotated with embedded entities following roughly the same pattern as ORG and LOC.

We follow the *Overspecify ambiguous categories principle*; it is easier to mark an entity as a HOTEL or MUSEUM rather than trying to remember whether it should be categorised as an ORG or FACILITY. These coarse-grained categorisation decisions can be put off to a later time to ensure annotation consistency.

AIRPORT Heathrow Airport, Schiphol, Charles de Gaulle, Kingsford Smith



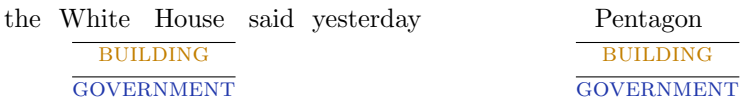
ATTRACTION The name of an attraction, including theme parks, monuments, etc. Note that not all ATTRACTION entities are run by CORP entities. See the discussion of FACILITY edge cases below. Wet 'n Wild, Memorial Coliseum, Statue of Liberty



BRIDGE The name of a bridge. Note, Locations should be marked up as embedded. Golden Gate Bridge, [G Street]$_{\text{STREET}}$ Bridge, [San Mateo]$_{\text{CITY}}$ Bridge

BUILDING The name of a building. Chifley Tower, Eureka Tower, Taj Mahal, Hundertwasserhaus

We follow the *Unary stacking principle* in cases of metonymy, which are common with the names of buildings. For example, the names of buildings which are also used as the name of the government body housed within should have be nested in a government entity when its use as a building is synonymous with the government. For example, when The White House, Parliament House or The Pentagon is used in a way that does not refer to the buildings themselves, but rather the organisations they represent, they should be nested in a GOVERNMENT entity.



STADIUM The name of a stadium. Note, references to locations etc should also be marked up. Superdome, Candlestick Park

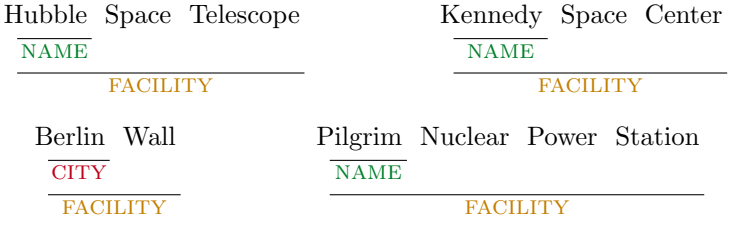
STATION The name of a station. Town Hall, Central, Lilyfield Station

STREET The name of a street, road or highway. STREET as a category has more in common with locations than organisations. It clearly fits with the BBN description of Facilities:

> Names of man-made structures, including infrastructure, buildings, monuments, etc.

so we elect to keep it categorised under FACILITY. However, streets should be considered with a more 'location' centered strategy. As such, we do not mark up embedded NAME references in STREET entities.

FACILITY:OTHER Other types of facilities not expressly covered by other categories, e.g. factories, sewage treatment plants etc. Kimbriki Resource Recovery Centre



FACILITY edge case Facilities, having both ORG and LOC characteristics, often are involved in tricky edge cases. In the following excerpt, it is not the actual Euro Disneyland theme park that is being discussed, but rather the company which owns and runs it.

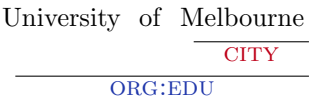
Traders credited Euro Disney's share performance to ...

Since this article is discussing the finances of the organisation, rather than the theme park for which it is named, it should be marked as CORP, with an embedded ATTRACTION.

3.3.4 LOCATION

As a category, LOCATION spans both physical (e.g. rivers, oceans, mountains) and geo-political entities (e.g. cities, suburbs, countries), both on Earth and beyond. The naming conventions of locations are complex, and deeply rooted in history, with the origin of many place names now forgotten. Many locations are named after people, and many people are named after locations. It would be prohibitively time consuming to check the history and naming origin of every location.

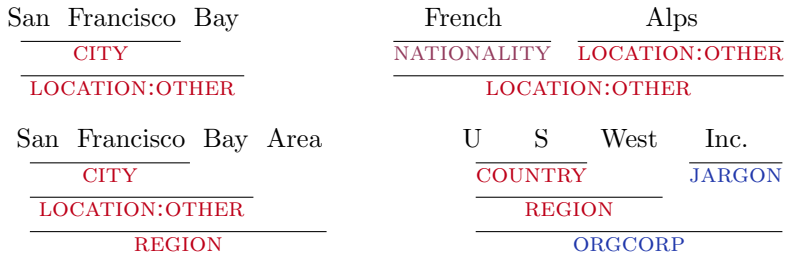
Take, for instance, the University of Melbourne. We annotate this as:



In 1873, the settlement now known as Melbourne was named after the then British Prime Minister, William Lamb, 2nd Viscount Melbourne, whose seat was Melbourne Hall in the market town of Melbourne, Derbyshire. Even with this history, it is unclear whether Melbourne could be considered to have an embedded

reference to an entity, and if so, it is not clear whether this should refer to a person (e.g. his father, Viscount Melbourne), his own title as the 2nd Viscount Melbourne, the market town, or his parliamentary seat.

Researching and correctly representing these relationships would be prohibitively time consuming for annotation. We therefore follow the *Pragmatic annotation principle* principle, and take the policy not annotating potential embedded NAME referents in LOC entities, but continue to annotate other nesting structures.



CITY Mentions of cities, towns and villages should be marked as CITY. Tokyo, New York, Chicago, Warsaw, Ho Chi Minh City, Mudgee

When a city is mentioned as the reduced form of the name of an organisation (e.g., In Tokyo, stocks fell. . . , referring to the Tokyo Stock Exchange), it should be marked as an organisation with embedded CITY.

This is also the case for capital cities, which can be used as a kind of spokesperson for the government of that country. Yesterday, Washington released new findings on... In this case, Washington should be marked as a CITY embedded in a GOVERNMENT.

SUBURB A smaller region within a CITY. Glebe, Beverly Hills, Croydon

STATE The largest internal administrative region in a country, including provinces of China (e.g. Shandong), prefectures of Japan (e.g. Nara Prefecture) etc.

New Hampshire, Indiana, N.J., Mass, Massachusetts

GPE This category captures geo-political entities that are larger than cities, that do not qualify as STATE, and that have fixed administrative boundaries. Even if a country (e.g. the U.K.) does not have states, mark small administrative regions (e.g. Bedfordshire County as GPE, not STATE. For example, Brooklyn (a borough) and Greenville County.

Brooklyn (GPE) Manhattan (GPE) Oakland (GPE) Puerto Rico (GPE)

San Andreas (GPE) Fault (LOCATION:OTHER)

U.S. (NATIONALITY) Virgin Islands (GPE)

COUNTRY Mark up mentions of countries such as Australia and France. The definite article the in countries such as the [Netherlands]$_{\text{COUNTRY}}$ is not marked.

France (COUNTRY) the Phillipines (COUNTRY) West Germany (COUNTRY) Viet Nam (COUNTRY)

CITY-STATE Locations such as Singapore, Hong Kong, Luxembourg and Monaco should be marked as CITY-STATE. Mentions of these entities represent genuine ambiguity between STATE and COUNTRY. Following the *Underspecify ambiguous categories principle* principle, we use this category to capture this ambiguity.

For example, in this sentence, the PTB derivation coordinates between crown prince and grand duke, though one can be crown prince of a country but duke of a city, meaning that Luxembourg must be interpreted as both concurrently.

```
( (S
    (NP-SBJ (NNP PRINCE) (NNP HENRI) )
    (VP (VBZ is)
      (NP-PRD
        (NP (DT the)
          (NX
            (NX (NN crown) (NN prince) )
            (CC and)
            (NX (JJ hereditary) (JJ grand) (NN duke) )))
        (PP (IN of)
```

```
        (NP (NNP Luxembourg) ))))
    (. .) ))
```

CONTINENT Mark up the name of continents. Africa, Australia

OCEAN Mark up the names of bodies of water other than RIVER, including oceans, seas and lakes. Pacific Ocean, Caribbean Sea, Atlantic Ocean

REGION REGION entities are named areas, usually larger than a city. Regions are contiguous areas for which the precise boundary may be not clearly defined, or may be disputed. Unlike GPE entities, which have clear legal distinctions, REGION entities often have *fuzzier* boundary distinctions.

RIVER The names of rivers and river deltas. Orange River, Hudson, River Danube, Mississippi

SPACE Any celestial body. Saturn, Milky Way, Andromeda, Titan

LOCATION:OTHER This category captures locations that don't fit into other location categories, for instance mountains, plateaus, plains. Mt Everest

A note on locations names with multiple possible referents In cases such as New York or Washington, which can be either a city or another location such as state, country or geopolitical entity, entities should be marked up as accurately as possible based on how they are used in context. For instance, if discussing a state-wide competition, the location should be marked as STATE. Many examples are more ambiguous, but general principles apply, for instance,

organisations usually discuss their offices being located in cities, not countries. If context is insufficient to determine which location is intended, we default to the larger geographic entity.

3.3.5 NORP: Nationality, Other, Religion, Political

NORP refers to the adjectival forms of entities.

> "This type is named after its subtypes, nationality, other, religion, political. The distinction between NORP and other types is morphological. American and Americans is a nationality, while America and US are GPEs, regardless of context."[2]

We take this definition, but disagree with the last point, that US should be marked as a GPE "regardless of context". While the distinction is difficult to make in cases where the adjectival and nominal form have the same surface realisation, such as U.S. in the phrase the U.S. National Anthem, it does nevertheless constitute a real distinction, and is marked as NATIONALITY in this context.

NATIONALITY Adjectival forms of or references to countries. American, Australian, U.S., French

NORP:POLITICAL Adjectival forms of political affiliations. Democratic, Liberal

NORP:POLITICAL only applies to references to a specific political party. For instance, if someone is described as 'conservative', this would usually be considered a general political view, and not a specific political party. To complicate matters, however, 'Conservative' is a specific political party in the U.K., compared to a set of any parties that are right of centre in the U.S.

[2] Annotation guidelines for Answer Types, BBN Technologies, Ada Brunstein http://catalog.ldc.upenn.edu/docs/LDC2005T33/BBN-Types-Subtypes.html

RELIGION Adjectival forms of religions. Muslim, Muslims, Jewish, Christian, Catholic, Jews

NORP:OTHER Other adjectival forms, including GPE names and locations. NATIONALITY, RELIGION and NORP:POLITICAL refer to adjectival forms for which there is a corresponding noun. All others should be marked as NORP:OTHER.

NORP:OTHER contains a variety of adjectival forms, relating to background, cultural or otherwise (Arab, Hispanic, Palestinian, Persian, Inuit, Western, African-American) to linguistic (Semitic, Alexandrine), historical (Victorian), theory or policy related (Darwinian, Thatcherite), and more esoteric (SONGsters, Jovian, New Environmentalism).

3.3.6 EVENT

The EVENT tag covers references to specific events including sports events, wars, natural disasters and named stock market crashes. Just as FACILITY as a category combines aspects of ORG and LOC, EVENT combines aspects of TIMEX and other named entities.

The types of TIMEX entities associated with EVENT entities is varied, ranging from a one-off natural disaster to a recurring festival. Some events are so connected with a specific date that they can be used as that date. For instance, We will see them on Christmas Day. or I'll be leaving on April Fools' Day. Since events and dates are so intrinsically linked, we use EVENT as another type of *top-level* TIMEX entity, which can also be embedded in TIMEX expressions, and have them embedded in it.

Named Events that are also Dates For entities like Christmas or Anzac Day, which convey both date and event information, the event is considered the *innermost*, intrinsic layer. For example, Christmas should be marked up first as an EVENT, then as a DATE.

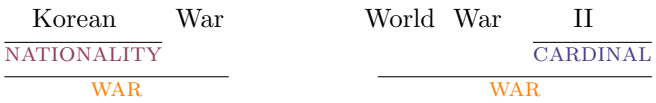
SPORTS-EVENT Tag the names of sports events, including chess games, motor sports, lawn bowls, etc. the Pan-American Games, the Ashes

CONCERT Named concerts. Live Aid, Wave Aid, Big Day Out, Rock Am Ring

HURRICANE The names of hurricanes, cyclones and other named tropical storms. The naming convention for tropical cyclones dates back to World War II, with first names being used to identify storms throughout their lifetimes. We keep the category name HURRICANE but expand its definition to include other tropical storms that follow the same naming convention: e.g. Hurricane Hugo, Cyclone Tracy. Since the naming pattern of these storms is clearly defined as including first names, HURRICANE should be embedded with FIRST entities, rather than NAME.

NATURAL-DISASTER The name of a natural disaster. This category captures natural disasters which do not share the naming conventions of tropical storms (HURRICANE). 2004 Boxing Day Tsunami, 2011 Tohoku Earthquake and Tsunami, 1971 San Fernando Earthquake, 1976 Tangshan earthquake.

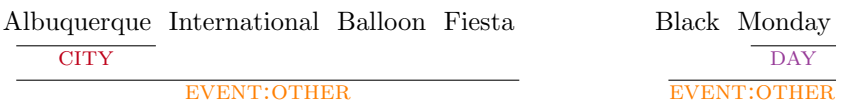
WAR The name of a war The Second World War, WWI, Hundred Year War



SPORTS-SEASON References to a particular sports season:

the 1989 Toronto Blue Jays season

EVENT:OTHER Types of event that are not captured in other specific event categories.



3.3.7 WORK OF ART

Works of art (WOA), generally considered a subcategory of MISC, and here separated for clarity, are annotated in a variety of subcategories. The names of WOA entities are often quite complicated, infrequently following patterns or naming conventions, and often including embedded mentions of other entities. They can have almost any internal syntactic structure.

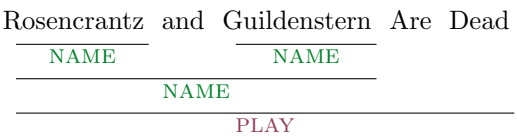
ALBUM The name of an album. Back in Black

BOOK Book titles. Pride and Prejudice, War and Peace

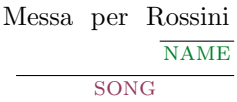
FILM The titles of films and movies: Batman, Gorillas in the Mist, When [Harry]$_{\text{FIRST}}$ met [Sally]$_{\text{FIRST}}$

PAINTING The names of paintings, for example: [Abraham]$_{\text{FIRST}}$ and [Sarah]$_{\text{FIRST}}$ in the Wilderness, Starry Night, Lighthouse [II]$_{\text{CARDINAL}}$

PLAY Twelfth Night, Death of a Salesman



SONG The names of songs and musical pieces, for example: Somewhere over the Rainbow, Violin Concerto in G Minor, When [Irish]$_{\text{NATIONALITY}}$ Eyes are Smiling



TV-SHOW The names of TV show series. For example, A Current Affair, [Sesame Street]$_{\text{STREET}}$, [Miami]$_{\text{CITY}}$ Vice.

WOA Other types of works of art including sculptures (e.g. The Impossibility of Death in the Mind of Someone Living) newspaper articles Rural Enterprise : Tough Row To Hoe, dance performances etc.

3.3.8 MISC

MISC entities are the names of other named entities that do not directly fit into other categories outlined.

LANGUAGE The name of a language, officially recognised or otherwise. French, English, Yimas

Russian (LANGUAGE) Streetspeak (LANGUAGE)

LAW The name of a law or constitutional amendment, act of parliament or constitution etc. The Kerrigan Decision, [Americans]$_{\text{NATIONALITY}}$ With Disabilities Act, [Fifth]$_{\text{ORDINAL}}$ Amendment

Roe (NAME) v. Wade (NAME) — (LAW)

Section 89 — (CARDINAL) — (LAW)

Clean Air Act — (LAW)

Johnson (NAME) Act — (LAW)

RICO — (LAW)

1974 (YEAR) Budget " Reform " Act — (DATE) — (LAW)

AWARD The name of an award. The Oscars, Luce Fellowship

the Oscars — (AWARD)

Luce (NAME) Fellowship — (AWARD)

ELECTRONICS The name of a type, brand or product of (generally consumer) electronics. Walkman, HyperCard, Cray-3, GameBoy

80486 — (CARDINAL) — (ELECTRONICS)

Apple (ORGCORP) II (CARDINAL) — (ELECTRONICS)

Intel (ORGCORP) 286 (CARDINAL) — (ELECTRONICS)

PRODUCT:DRUG The name of a drug (product), e.g. Viagra.

Proleukin — (PRODUCT:DRUG)

Retin – A — (PRODUCT:DRUG)

AZT — (PRODUCT:DRUG)

PRODUCT:FOOD The brand/product name of a food, e.g. Coco Pops, Weetabits, Vegemite.

Fuji
PRODUCT:FOOD

Frosted Flakes
PRODUCT:FOOD

VEHICLE The name of a type of vehicle, e.g. Hummer, Jeep Cherokee, as well as the names of individual boat and planes, e.g. Airforce [One]$_{\text{CARDINAL}}$.

Ford trucks
NAME
ORGCORP

Jeep Cherokee
ORGCORP NORP:OTHER
VEHICLE

WEAPON The name of a weapon AK47

DISEASE The name of a disease, including mentions of diseases not starting with a capital letter.

Parkinson's
NAME
DISEASE

Parkinson's disease
NAME
DISEASE

AIDS
DISEASE

retinoblastoma
DISEASE

GOD The name of a deity, or reference to a god. God, Zeus

SCINAME The name of a specific scientific name such as genus.
Homo sapiens, Nymphicus hollandicus, Ailurus fulgens

Nymphicus hollandicus
SCINAME

PRODUCT:OTHER The name of other products, including both tangible and intangible products. This also includes the names of documents such as standardised tests. Birkenstocks, SATs

Cheer with Color Guard
PRODUCT:OTHER

Personal Retirement Account
PRODUCT:OTHER

Tide with Bleach
PRODUCT:OTHER

Dalkon Shield
NAME
ORGCORP
PRODUCT:OTHER

3.3.9 GROUP

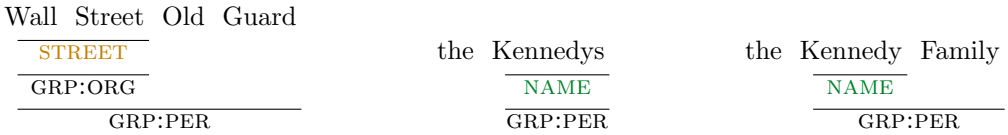
GROUP as a category captures unorganised groupings of entities.

GRP:ORG GRP:ORG is for not officially organised groups of organisations. Wall Street Here Wall Street is referring to a group of corporations in the finance industry. They are not officially organised, and are known by this term. Not all members of Wall Street need, in fact, have an office on Wall Street. Similarly, other organisations have offices on Wall Street, but do not belong to the finance industry, and therefore to the group to which Wall Street refers.

GRP:LOC Groups of locations that are not regions. These do not necessarily need to be contiguous, but instead are groupings based on things in common, for example financial situation or language spoken. PIIGS, First World

If the grouping were due to geographical location, they would instead be marked as REGION.

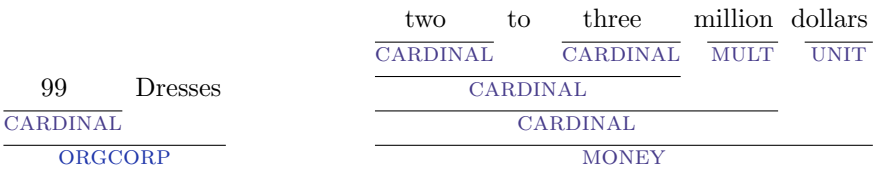
GRP:PER Unorganised (or unofficially organised) groups of people, including families.



3.3.10 NUMEX

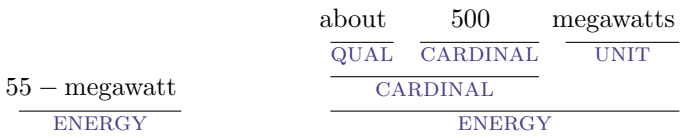
The structure of numerical expressions (NUMEX) is of particular interest when considering resolving these expressions into one canonical form.

CARDINAL CARDINAL includes all mentions of counting and fractional numbers, both in and outside other entities. We mark up all cardinals, both as stand-alone entities and embedded in other entities. We also mark up several and few as cardinals, as they do convey cardinality, albeit underspecified.

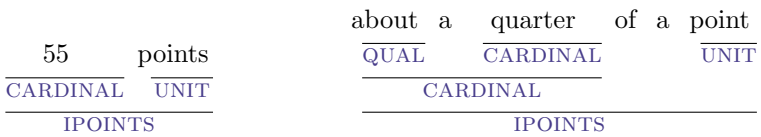


FOLD The FOLD tag is similar to the MULT in use, covering cases such as fivefold, 10 times as much, twice etc. FOLD indicates the multiplication factor of a number.

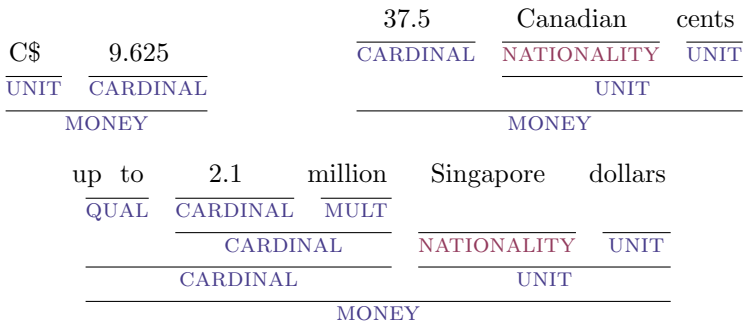
ENERGY Annotate mentions of energy



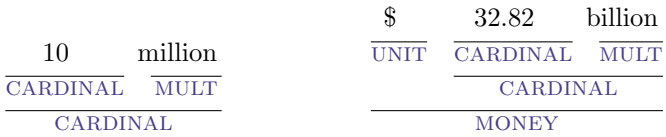
IPOINTS Short for Index points, we mark up mentions of index points, taking care not to mark up percentages. Although technically unitless from a science perspective, IPOINTS are treated as a unit in finance newswire, so we annotate these accordingly.



MONEY Any mention of an amount of money.



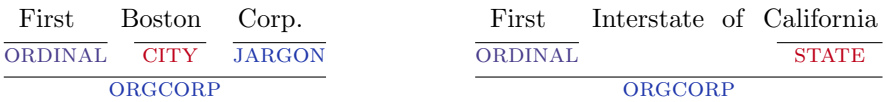
MULT A 'multiplier' marker - e.g. million, billion, trillion. This is used to mark up structure of numbers, especial in MONEY.



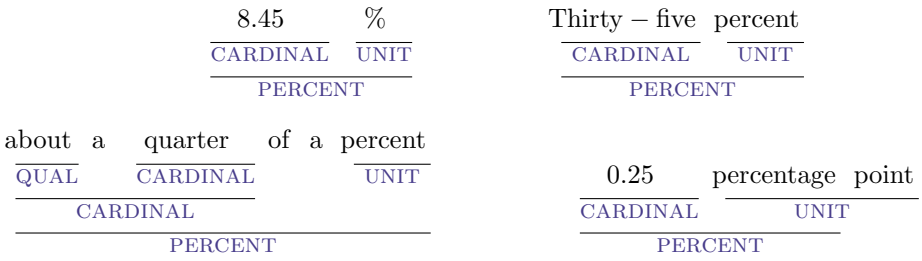
This additional MULT layer is useful in cases of an extended description of an amount, where the amount is referenced later with just a CARDINAL, excluding the MULT. For instance: The total number is expected to be $[[3]_{\text{CD}}$ $[\text{million}]_{\text{MULT}}]_{\text{CD}}$, "but that might go as high as $[4]_{\text{CD}}$", says... In this case, the extra MULT (referring to 4 million) has been elided. This added structure will therefore assist in future coordination tasks.

ORDINAL Ordinals are numbers defining a position in a series, such as first, second, third, or last. Ordinal numbers are used as adjectives, nouns, and pronouns.

We also annotate ORDINAL tokens inside ORG entities even in cases where it is not clear that this semantic concept has been retained.

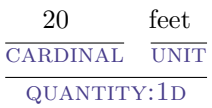


PERCENT The PERCENT category captures references to percentages and percentage points, including nested structural elements such as the token percentage as a UNIT.

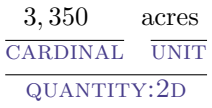


QUANTITY Quantities are usually used with nested CARDINAL and UNIT of some sort, and are often found within RATE structures. We only look at QUANTITY measures that have defined units, for instance, we would mark up 3 teaspoons but not 3 spoons. Similarly, we do not annotate non-standard measures such as 10 men or 20 chickens.

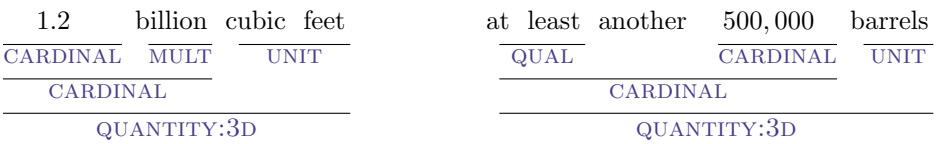
QUANTITY:1D One dimensional quantities include measures of distance. The SI base unit for QUANTITY:1D is meter.



QUANTITY:2D Two dimensional quantities include measures of size, such as feet, acres, square kilometers etc. QUANTITY:2D measurements are of the same form as SI derived unit square meter.

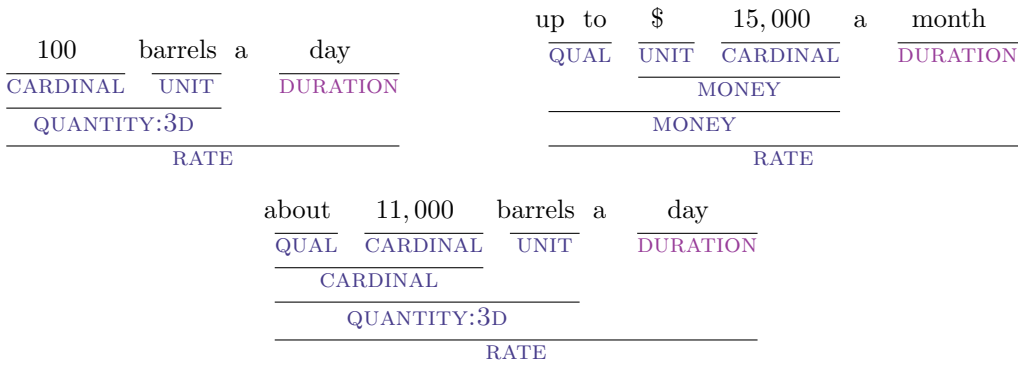


QUANTITY:3D Three dimensional quantities include measures of volume, the SI derived unit for which is cubic meter.



QUANTITY:OTHER Some quantities that do not fit into either the QUANTITY:1D, QUANTITY:2D or QUANTITY:3D categories, which we classify as QUANTITY:OTHER. These include measures of computational size and degrees, i.e. not spatial measurements.

RATE The RATE category covers all rates: measures, quantities or frequencies measured against some other quantity or measure (usually time). RATE includes measures such as dollars an hour, dollars an ounce, beats per minute.

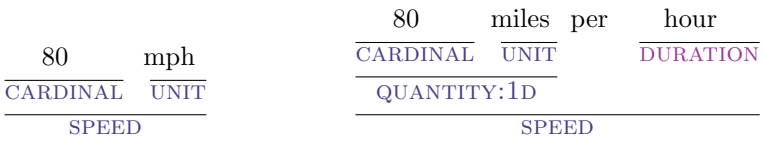


SPEED SPEED is a measure of QUANTITY:1D per DURATION, the SI unit for which is meter per second. We distinguish between SPEED and other types of RATE due to the frequency of SPEED mentions in text.

In cases of ambiguous potential structural bracketing, as seen below, we follow the *Defer to PTB principle*, and mark the bracketing which does not conflict with the PTB structure.

((80 miles) per hour)

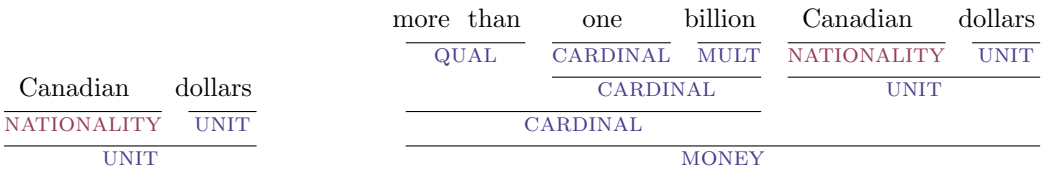
(80 (miles per hour))



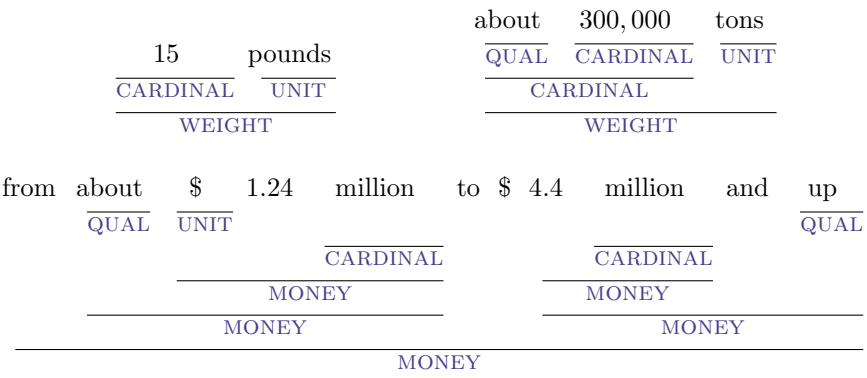
TEMPERATURE Annotate instances of TEMPERATURE, marking up cardinality and units.

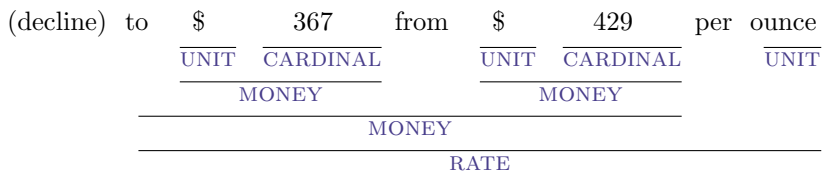
UNIT A specific unit of measurement that is not a RATE (e.g. beats per minute). For example, $, cents, yen, dollars, US$, pounds, C$, Canadian dollars, hours, acres, miles, ounces, square feet, barrels. The type of the unit is resolved by the outer layer of annotation.

Units mentioned without a CARDINAL should still be marked as UNIT. For example, . . . against the [dollar]$_{\text{UNIT}}$ and the [[West German]$_{\text{NATIONALITY}}$ mark.]$_{\text{UNIT}}$



WEIGHT Annotate instances of weights, also marking up embedded CD and UNIT.

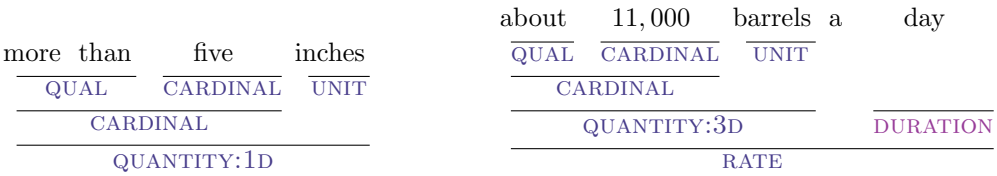




Qualifiers QUAL changes the meaning of a NUMEX or TIMEX expression, usually by adjusting one of the limits of a particular (usually numerical) amount. Common QUAL expressions include almost, around, nearly, more than, up to, at least. We attach QUAL entities as closely to the CARDINAL span (or largest CARDINAL span if applicable, which produces essentially the same semantics) as possible, and produces a larger span of the same type to which it joined.

$$\text{QUAL} + X \rightarrow X$$

QUAL spans change the meaning of an amount, but can also often express editorial content. For example, in the clause 'could be released as early as next year', the words as early as are marking a point of view, specifically, that next year would be early. Making a distinction between editorial comment and the equivalent meaning of at or before is nuanced, and is beyond the scope of this annotation task. We follow the *Pragmatic annotation principle*, and therefore mark up all instances that could be interpreted as changing the meaning of an amount as QUAL.



The policy of attaching QUAL spans as closely to the CARDINAL span as possible is consistent with the PTB analysis, where these phrases are (generally) included within the larger QP span.

```
(NP-SBJ-92 (DT the) (NN roof) )
(VP (MD could) (RB n't)
  (VP (VB be)
    (VP (VBN depressed)
```

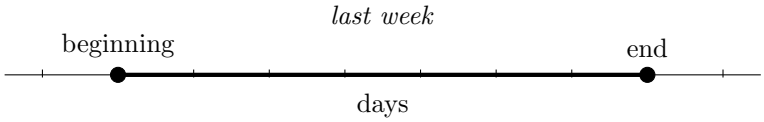
```
          (NP (-NONE- *-92) )
          (NP-EXT
            (QP (RBR more) (IN than) (CD five) )
            (NNS inches) ))))
    (. .) ))
```

3.3.11 TIMEX

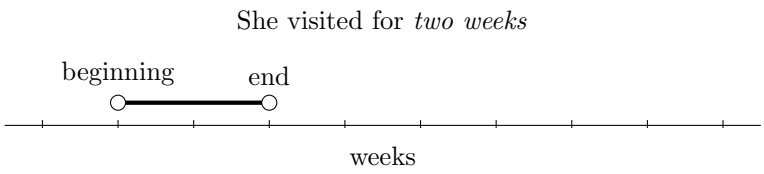
We consider TIMEX entities to cover expressions of time, both points in time and longer periods of time.

Most TIMEX expressions fall into either the DATE, DURATION or PERIODIC category. These three are the main *top-level* types of TIMEX, with more detailed sub-types building up internal structure.

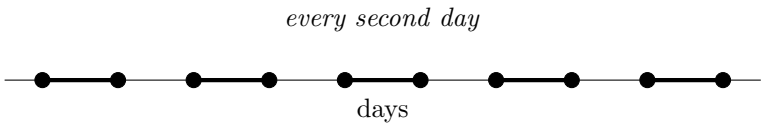
We define DATE expressions as any time expressions which can have one or more ends placed on a timeline. This includes specific punctual instances (15:00) and longer spans of time, where one or both ends is specific (January 2012). All dates and times are, at some resolution, temporal spans. Further, we assume that the date of publication, or *today* is known, and therefore treat dates such as last week or yesterday as dates which can be placed on the timeline.



DURATION covers spans of time for which neither end can be identified and placed on a timeline. For instance, for two weeks should be marked up as a DURATION, since neither end is specified, but the length of the span on the timeline is fixed.



PERIODIC expressions are recurring time events, usually repeating in a regular pattern, such as annual, fortnightly or each Monday at 5pm. They may or may not be fixable on the timeline.



DATE:OTHER covers any times which do not fit into this categorisation. There are four main 'template' rules for TIMEX structures:

CARDINAL + DURATION → DURATION

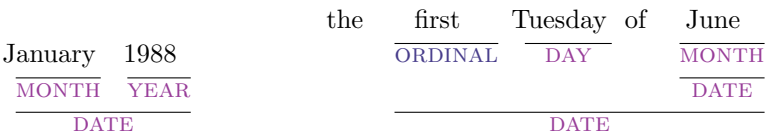
DURATION + REL(+DATE) → DATE

REL + DATE → DATE

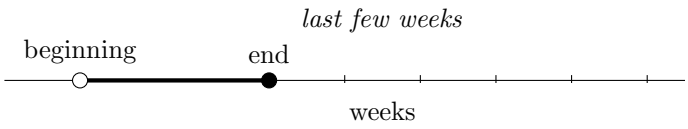
DURATION + DATE → DATE

Temporal expressions are particularly complex and productive, and following the *Monitoring grammar principle* has been particularly useful in trying to maintain a parsimonious annotation scheme that still captures the syntactic and semantic structure of temporal expressions.

DATE DATE should be annotated with inner structure, predominantly using the categories following in this TIMEX section. For instance, the names of days or months, date and year structures should all be marked up. Additionally, other aspects of the date including REL markers (see Section 3.3.11), such as early should be marked up.

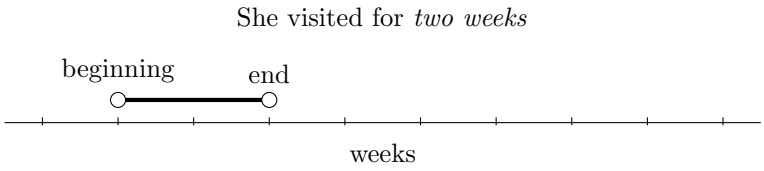


Date entities can also have only one date which is pinned down. Common examples of these include in recent years and for the last few weeks.



recent years
REL DURATION
DURATION

DURATION A DURATION is a time period that has neither end anchored to a time line, but the length of the line is fixed.



She stayed for two weeks
CARDINAL DURATION
DURATION

another day or two
DURATION CARDINAL
DURATION
DURATION

A common pattern inside TIMEX entities is the combination of a DURATION, a time span which, by itself, cannot be pinned down to a specific start or finish time, and a REL span, which pins the DURATION to the timeline, forming a DATE span.

last week
REL DURATION
DATE

A note on Financial Quarters Financial quarters should be marked as *quarter* years, with embedded CD:

quarter
CARDINAL
DURATION

A quarter as a token implies a duration, specifically a quarter of a year. When combined with an ordinal or other date structure, e.g. first quarter 1980,

it resolves to a DATE. Though the dates for quarters vary between countries, from context, we can identify the bounds of the fiscal first quarter, or the last quarter of the year.

PERIODIC The PERIODIC tag is for recurring DATE entities, usually a set of DATE entities. This includes mentions such as annual, weekly, fortnightly, every Tuesday etc.

We only mark up these as PERIODIC if they are specifically acting as recurring, periodic markers. For instance, the example Tuesday does not always convey PERIODIC and so should not be marked as such except for instances such as:

On a Tuesday, like clockwork, Thomas goes swimming.

Similarly, On the weekend should be marked as periodic if it refers to a recurring time, but not if it refers to a one-time event:

On the weekend (PERIODIC) , Thomas goes swimming.

On the weekend (DATE) , Thomas will go swimming.

Meetings are every Wednesday (DAY) (PERIODIC) .

... supplies programs on Saturdays (PERIODIC) , Sundays (PERIODIC) and Mondays (PERIODIC) .

AGE The AGE tag marks up ages.

12 (CARDINAL) years (DURATION) old (AGE)

under (QUAL) 45 (CARDINAL) years (DURATION) of age (AGE)

DAY Mark up the names of the days of the week: Monday, Tuesday, Wednesday etc.

When found without other TIMEX constructions, DAY is embedded in a DATE tag, taking extra specification from context. That is, because newswire generally discusses the recent past or obvious close future, we assume that the larger context is clear from the sentence or article.

on Tuesday , she was elected ...
(Tuesday: DAY, DATE)

DAY can also occur within larger DATE constructions, where it combines with them immediately with other elements, usually MONTH, NUMDAY or YEAR, to add specificity to the larger DATE structure:

Tuesday , October 10 , 1989
(Tuesday: DAY; October: MONTH; 10: NUMDAY; 1989: YEAR; all: DATE)

MONTH Annotate the names of months of the year: January, February, March etc.

August 1985
(August: MONTH; 1985: YEAR; all: DATE)

NUMDAY This is for references to days using numbers, ie., the 12th. It is always embedded in a DATE. 15th, 1st, ...

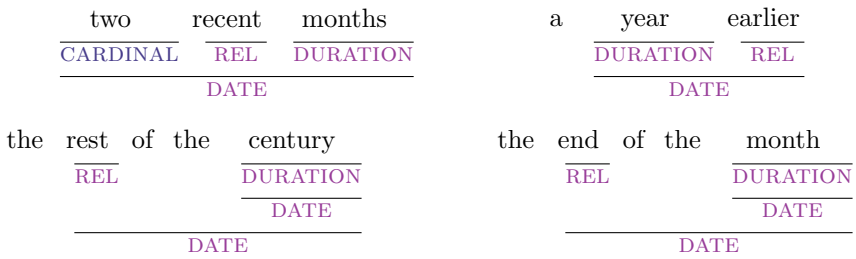
Nov. 21
(Nov.: MONTH; 21: NUMDAY; all: DATE)

REL Certain expressions show the relationship between another TIMEX expression and the current day. These should be marked as REL.

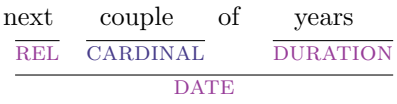
Common REL expressions are next, ago, last, this, past, end, ended, ending, previous, early, after, following, later, before and most recent.

REL spans are frequently combined with DURATION to form a larger DATE span, effectively pinning a duration to a fixed date, for example, next week, last month etc. REL + DURATION → DATE

We use this *Monitoring grammar principle* with patterns like this when making complex nesting decisions.

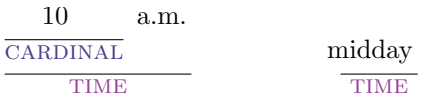


The analysis of some phrases are ambiguous with respect to bracketing. For example, consider the next couple of years. This could be analysed as either next (couple of years) or (next couple) of years. Since they are semantically equivalent, following the *Defer to PTB principle*, in order to avoid conflicting with the PTB bracketing, we elect to leave such ambiguous structures as flat as possible.

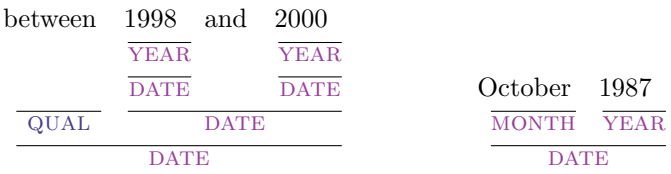


SEASON We use the *Overspecify ambiguous categories principle* and mark the temporal seasons of the year, summer, autumn, winter, spring, as SEASON.

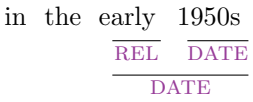
TIME The distinction between DATE and TIME is difficult to identify with our definition of DATE as any time expression which can be placed on a timeline, as TIME expressions such as 2pm Tuesday also satisfy this constraint.



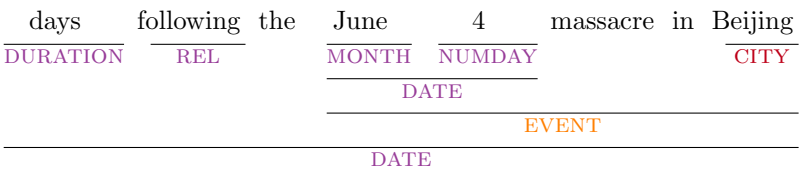
YEAR Years should be marked YEAR. Just as we would mark up plurals of other entities as the entity in question, we also mark up YEAR spans, such as 1983-1988 as YEAR.



We do not have a specific *Decade* category, but instead mark up mentions of decades, such as the 1950s, directly as a DATE.



Further discussion of DATE and EVENT Black Monday in this corpus refers to a stock market crash in 1987. It is used to refer to the event itself, but also frequently used in references to the date of the event.

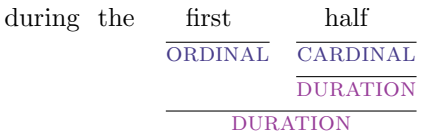


Flat TIMEX entities with structural ambiguity In many TIMEX entities, multiple possible spans are potentially valid interpretations. Consider: the first half of the year We annotate this as a flat structure, leaving both of the following phrasings possible:

[first half] of the year

first [half of the year]

If there is no DURATION specified, we can follow this interpretation:



Note that the following analysis is problematic for the same reason as TIMEX expressions such as two weeks of June or NUMEX expressions similar to three barrels of oil in that it is unclear where attachment should happen. Both first half and half of the year are equally valid options, and as such, we elect to keep the structures as flat as possible, allowing the process of merging the PTB corpus and our NNE corpus, to be discussed in Chapter 5, to determine the bracketing.

3.4 Summary

In this chapter, we introduced the annotation principles governing entity spans, their nesting, and the granularity of the category inventory – collectively, the annotation scheme. We summarised the annotation guidelines for each entity type, and described the use of the BBN corpus as underlying entities for our nested entity annotation process. For further details, with many more edge cases discussed, please see the full *Nested Named Entity Corpus Annotation Guidelines* that will be published along with the corpus.

Many existing NER corpora have minimal annotation guidelines, with poor coverage of edge cases. The well-defined annotation principles and detailed guidelines presented as part of this thesis allow for the creation of a consistent corpus of nested entities. These guidelines bring the same level of detail to the task of NER as the annotation guidelines for the Penn Treebank do to syntactic parsing.

4 Annotating the NNE corpus

> If we have data, let's look at data. If all we have are opinions, let's go with mine.
>
> Jim Barksdale

Now that we have a robust set of annotation principles and a highly detailed set of annotation guidelines, the next step is to annotate the corpus. This chapter will describe the process of manually annotating structured named entities in our corpus, the Wall Street Journal section of the Penn Treebank. The resulting NNE corpus will be used throughout the rest of this thesis, being merged with the constituent structure of the PTB in Chapter 5 for subsequent parsing experiments (Chapter 6) and NER experiments (Chapter 7).

4.1 Annotation Process

The first step in the annotation process was to align BBN to the PTB, so that the underlying BBN annotations are compatible with the target corpus. Once the corpus with underlying BBN annotations was created, we then applied a number of pre-annotations to add specific particularly regular structures automatically which would be checked by the annotators in the next step, when they worked through the corpus, adding in the bulk of entity structure. Following that, the corpus was post-processed to improve consistency and fix

both changes to the annotation guidelines that occurred during (and on the basis of experience) annotating, and annotation errors.

4.1.1 Aligning BBN and PTB

Missing data A small subset of sentences in the PTB were missing from BBN. These were manually added and annotated for inclusion in the corpus.

Sentence Boundary issues The BBN corpus tokenisation differs from that of the PTB.

Since we want the resulting corpus to be as compatible with the Penn Treebank as possible, we first align the BBN corpus to it, correcting various sentences with invalid XML, sentence boundary problems and modifying the tokenisation, which differs between BBN and the PTB. For example, BBN includes 1958 hyphenated compound tokens, where at least one section is an entity (e.g. [London]$_{\text{CITY}}$-based, [three]$_{\text{CARDINAL}}$-run). Since our aim is to create nested named entity annotations that are compatible with the PTB corpus, we extend the entity boundary to cover the whole token ([London-based]$_{\text{CITY}}$).

Most of the alignment errors were caused due to tokenisation discrepancies between the two corpora. One example of this is full stops. When periods occurred within entities, the algorithm used by BBN to detect sentence boundaries seems to have incorrectly identified any periods followed by a capital letter as a sentence boundary. This is due to periods not being repeated when occurring consecutively for different reasons. One does not see, for instance, this punctuation: I watched Monsters Inc.. where the sentence is ended with two full stops. For instance, the entity [Cie . Financiere de Paribas]$_{\text{CORP}}$ was split into two sentences:

. . . by [Cie .

Financiere de Paribas]$_{\text{CORP}}$ at . . .

causing invalid XML in both sentence fragments. These errors were identified and corrected.

Invalid XML Additional cases of invalid XML, such as an entity starting with a TIMEX label but being closed by NUMEX tag were manually corrected.

```
<TIMEX TYPE="TIME">more than 4,000 hours</NUMEX>
```

Tokenisation issues A number of tokenisation issues were corrected, most notably that BBN included sub-token level annotations.

[Washington]$_{\text{CITY}}$-based → [Washington-based]$_{\text{CITY}}$

pre-[Communist]$_{\text{NORP:POLITICAL}}$ → [pre-Communist]$_{\text{NORP:POLITICAL}}$

[U.S.]$_{\text{NATIONALITY}}$-[Japanese]$_{\text{NATIONALITY}}$ → [U.S.-Japanese]$_{\text{NATIONALITY}}$

The changes exemplified above were changed automatically, with more complex changes, such as the examples below, made manually.

[capitalist-exploiters-greedy-American-consumers-global]$_{\text{NATIONALITY}}$

a [[[$]$_{\text{UNIT}}$ [3-a-person]$_{\text{CARDINAL}}$]$_{\text{MONEY}}$]$_{\text{RATE}}$ tax

4.1.2 Annotation Pre-process

The structure of certain types of named entities follows regular patterns. We exploit this by performing a pre-annotation processing step, adding expected structure to PER, ORG, NUMEX and TIMEX entities that match particular linguistic patterns.

PER entities that are only a single token have NAME added as an additional structural layer, inside the PER annotation. PER entities of two tokens are annotated with either FIRST and NAME respectively, or INI instead of FIRST if the first token is an initial. PER entities of more than two tokens are annotated with FIRST, INI if present, and NAME, but not further annotated with additional

labels, since distinguishing between double barrel first names, MIDDLE names and double barrel surnames is beyond the scope of this preprocessing task.

PER entities preceded by a predetermined set of honorifics (HON) and roles (ROLE) had them marked up, and the PER span grown to incorporate them. Before the annotation process started, the distinction between HON and ROLE had not been made, and all were marked as HON. Once the category was split into two separate entity tags, existing annotations, including these pre-annotations, were changed programmatically.

CARDINAL and MULT entities were added to TIMEX and NUMEX entities, as well as the UNIT tag being added to a number of frequent money entities (e.g. '$', 'dollars', '$US', yen 'C$' etc.) and other common units.

The days of the week, and months of the year, as well as common contractions thereof, were annotated with DAY and MONTH respectively. Four-digit words starting with 19 that occurred within TIMEX entities were marked as YEAR, and numbers less than 32 occurring in TIMEX entities were marked as NUMDAY. While we know that this will occasionally be inaccurate, for example, if an age of less than 32 is given it should not be marked as NUMDAY, but adding these structures everywhere, and occasionally correcting them substantially reduced the amount of annotation time spent on each sentence.

Common JARGON words (e.g. Co., Corp, Corp., Inc., Ltd, Ltd., Co, PLC, Inc, Inc.) were added, as were a number of QUAL words (e.g. about, last, roughly, almost, fewer than, at least,less than, more than, well over).

4.1.2.1 Automatic Nested Annotation Suggestions

In addition to pre-annotating the corpus with common structures, we also develop an annotation suggestion system that will allow annotators to easily add consistent structure for the same entity where applicable. As an annotator adds entity spans, these are added to their personalised annotation suggestion

system. When the user is looking to annotate an entity span, a number of suggestions, ranked on frequency, will appear for tokens or token spans within the entity that have been previously annotated.

In addition to previous annotations, this list is seeded with frequent entities. The most frequent 1000 entities occurring in the BBN corpus were manually annotated and added to each annotator's suggestion list.

4.1.3 Annotation Tool

A custom annotation tool was built that allows the annotation of nested structures. While some existing annotation tools do allow nested structures to be annotated (e.g. Brat[1], MMAX2[2]), building a custom tool allowed us to create a clear, simple, and fast way to let annotators quickly and easily add layers of named entity annotations, and reuse existing annotations for the same span (see below).

Using the annotations from BBN as underlying annotations, further enhanced by our pre-annotation step, the annotator is shown a screen with the target sentence. The previous and next sentences, if any, are shown discretely in small font above and below the current sentence. In Figure 4.1, the next sentence is visible at the bottom of the page, and in Figure 4.2, that sentence is seen at the top of the page. This helps the annotator with contextual cues, and a view of the whole article is also possible for particularly ambiguous entities. The annotation tool also allows annotators to *flag* difficult sentences for further discussion, allowing them to delay particularly difficult annotation decisions or edge cases.

When an annotator selects a span, they are prompted with suggestions based on their own previous annotations, and common entities (see Section 4.1.2.1),

[1]brat rapid annotation tool, http://brat.nlplab.org/
[2]http://mmax2.sourceforge.net/

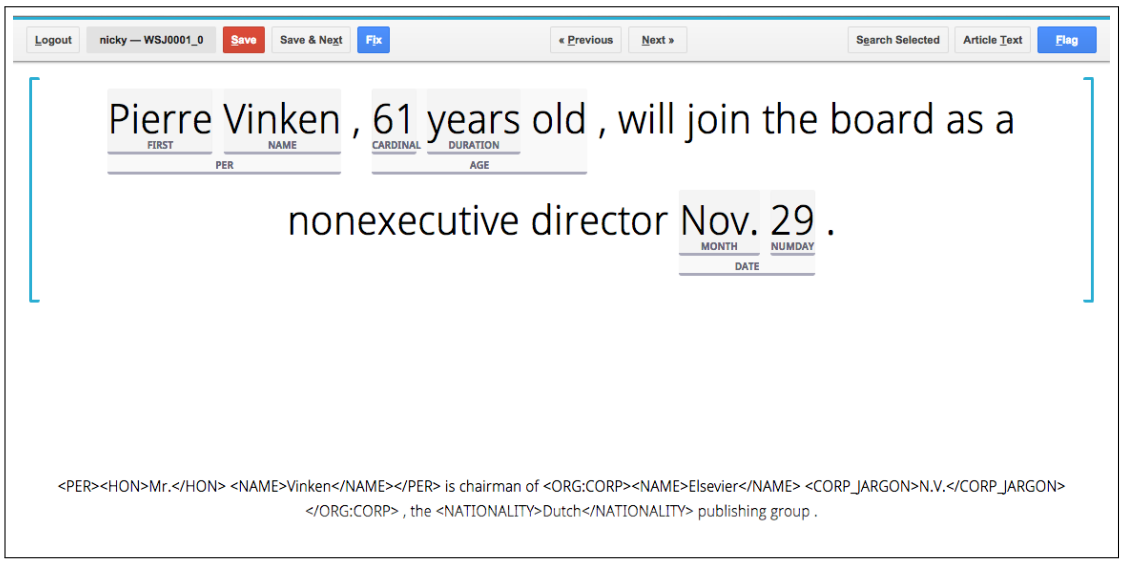


Figure 4.1: Annotation tool showing pre-annotation of sentence `WSJ0001_0`.

seen in Figure 4.2. In this example, the first suggestion of adding a NAME span over Elsevier and the second suggestion of adding JARGON over N.V. are both combined in the third suggestion, which is the correct nested structure for the entity. The annotator can simply press the 1, 2 or 3 number key to have this annotation span added. As more entities are annotated, these suggestions improve.

Some entities are repeated frequently in an article, or over many articles in the corpus. As shown in Figure 4.3, the annotation tool allowed a user to add a specified annotation to all strings matching those token(s) in the same article, or in all articles. The current analysis of those tokens is also shown. This allowed annotators to easily maintain consistency over their own annotations, and reduced the chance of typos causing annotation errors.

4.1.4 Annotation Time and Process

Four annotators, each with a background in linguistics and/or computational linguistics were selected and briefed on the annotation task and purpose. Each annotator started with a subset of section 00 as annotation training, and was

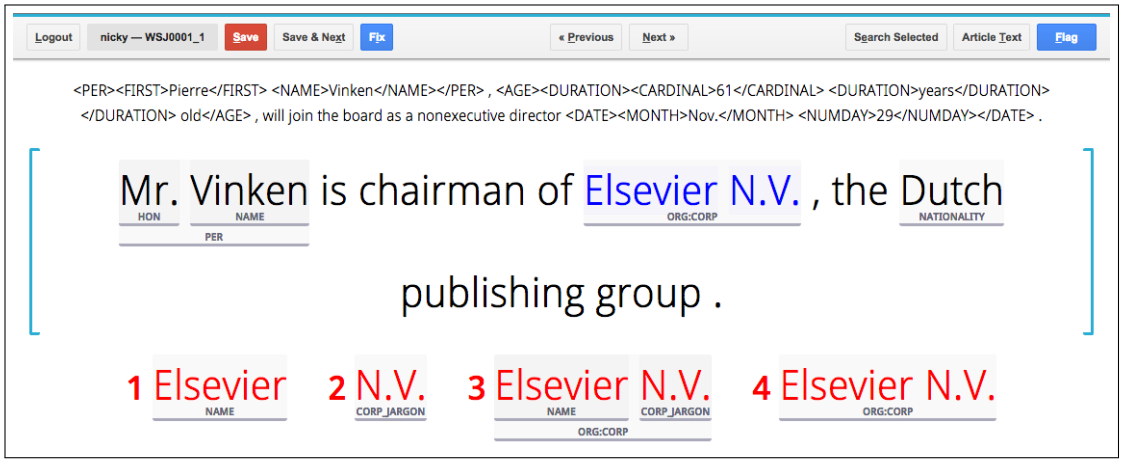


Figure 4.2: Annotation tool showing suggestions for Elsevier N.V. in `WSJ0001_1`. Suggestions, discussed in Section 4.1.2.1, are shown below in red when a span is selected.

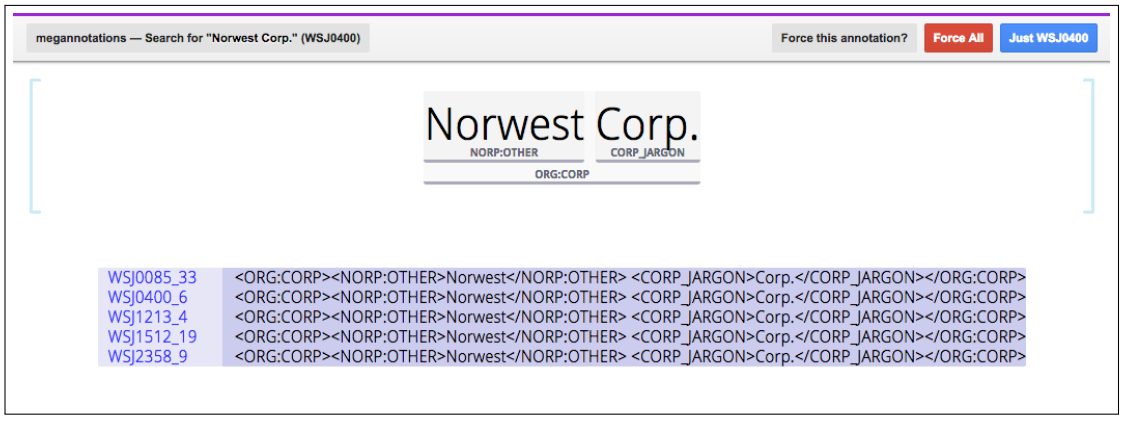


Figure 4.3: Annotation tool showing annotation of a particular entity, Norwest Corp., over all sentences in the corpus. From this page, a user is able to add that particular annotation structure to all sentences from the origin article, or all sentences in the corpus.

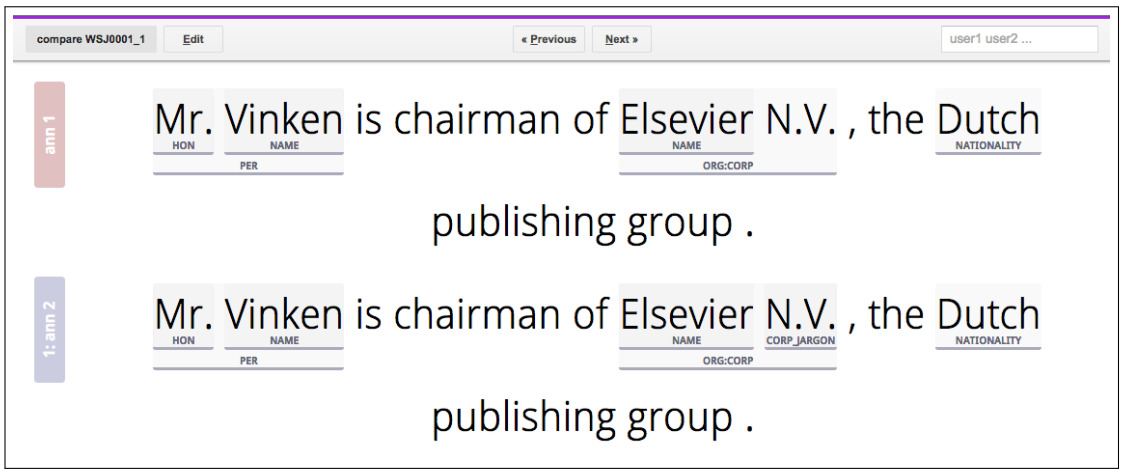


Figure 4.4: The comparison mode of the annotation tool, showing the annotations of two separate annotators. This is used for adjudication between doubly annotated sections 00 and 23, and section 02 which was annotated by all annotators.

given feedback before moving on to other sections. Weekly meetings were held with all annotators to discuss ambiguities in the guidelines, gaps in the annotation categories, edge cases and ambiguous entities and to resolve discrepancies.

Total annotation time for the corpus was 270 hours, split between the four annotators. After the initial training annotation of a subset of section 00, annotators averaged between 4,100 and 6,500 words per hour. Sections 00 and 23 were doubly annotated, and section 02 was annotated by all four annotators. An additional 17 hours was used for adjudicating these doubly annotated sections.

4.1.5 Inter-annotator agreement

To measure both how clearly the annotation guidelines delineated each category, and how reliable our annotations are, inter-annotator agreement was calculated. All annotators annotated section 02, and all four annotators' data was used to create a gold-standard adjudicated version. The adjudicated version was created by deciding a correct existing sentence from within the four possibilities, or by adjusting one of them on a token level. Annotator 2 performed the

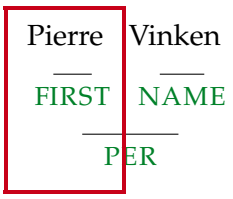


Figure 4.5: Example derivation showing *tag stack* used for inter-annotator agreement. Highlighted here is the composition of the label: FIRST_PER for the token Pierre.

adjudication, using the annotation tool comparison feature, shown in Figure 4.4 (depicted comparing two annotations). The tool showed all four annotators' sentences, with Annotator 1's annotations at the top. The comparison tool skips all sentences where all annotators' decisions were identical, instead progressing to the next sentence where at least one annotator differed in one decision.

For the purposes of inter-annotator agreement, a *tag stack* was calculated for each word, essentially flattening each tokens' nested annotation structure into one label. For instance, Figure 4.5 shows the entity $[[\text{Pierre}]_{\text{FIRST}}\ [\text{Vinken}]_{\text{NAME}}]_{\text{PER}}$, which would count as two separate tokens: Pierre with label: FIRST_PER and Vinken with label NAME_PER. Using the tag stack as a form of analysis allows us to capture all annotation layers, however it does make this a particularly harsh analysis, given that there are over 600 unique tag stack categories in section 02.

The four annotators' agreement, in the form of precision, recall and F_1-score to the adjudicated gold standard for section 02 is seen in Table 4.1. These results are very promising, given that an error in any level of annotation would result in that token being marked as an incorrect label. On the other hand, the inter-annotator agreement achieved is also boosted by starting with identical *base* annotations from the BBN corpus. Additionally, Annotator 1's F_1-score of 90.6% was also potentially affected by its prominence in the adjudication tool.

Annotator	Precision	Recall	F_1-score
Annotator 1	90.4	90.9	90.6
Annotator 2	84.5	83.5	84.0
Annotator 3	82.2	81.8	82.0
Annotator 4	84.3	83.7	84.0

Table 4.1: Inter-annotator agreement for each of the four annotators, calculated on a tag stack, on section 02 against final adjudicated annotations.

To investigate this further, we analysed the inter-annotator agreement between annotator pairs, shown in Table 4.2. This total micro-averaged score shows especially high agreement between Annotators 1, 2 and 4.

Annotator	Annotator	Precision	Recall	F_1-score
Annotator 1	Annotator 2	86.9	85.4	86.1
Annotator 1	Annotator 3	83.8	82.9	83.4
Annotator 1	Annotator 4	85.9	84.8	85.4
Annotator 2	Annotator 3	84.3	84.9	84.6
Annotator 2	Annotator 4	86.1	86.5	86.4
Annotator 3	Annotator 4	84.7	84.5	84.6

Table 4.2: Inter-annotator agreement between each pair of annotators, calculated on the innermost tag only, on section 02.

Fleiss' kappa, a measure for assessing the reliability of agreement between a fixed number of raters, was also calculated, and found to be 0.834. Fleiss' kappa is similar to Cohen's kappa, which only works when assessing the agreement between two raters. Fleiss' kappa can be interpreted as the extent to which the observed agreement between annotators exceeds what would be expected if all annotators made their decisions randomly.

Fleiss' kappa can be defined as

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e} \tag{4.1}$$

with factor $1 - \bar{P}_e$ giving the degree of agreement that is above chance, and $\bar{P} - \bar{P}_e$ giving the degree of agreement achieved above chance.

Fleiss' kappa was calculated on the annotations in section 02, amounting to a total of 679 different categories over the 48,134 tokens.

To calculate Fleiss' kappa, let N be the total number of tokens that are annotated (48,134), and n be the number of ratings per subject (in our case, we have 4 annotations for each token). Let k be the number of categories (in this case, 679 tag stacks that form the categories). Tokens are indexed by $i = 1, \ldots, N$ and categories indexed by $j = 1, \ldots, k$. Let n_{ij} represent the number of annotators who assigned the i-th token to the j-th category.

We first calculate p_j, the proportion of all annotations which were made to the j-th category:

$$p_j = \frac{1}{Nn} \sum_{i=1}^{N} n_{ij} \tag{4.2}$$

We next calculate P_i, the extent to which the annotators agree for the i-th token:

$$P_i = \frac{1}{n(n-1)} \sum_{j=1}^{k} n_{ij}(n_{ij} - 1) \tag{4.3}$$

$$= \frac{1}{n(n-1)} [(\sum_{j=1}^{k} n_{ij}^2) - (n)] \tag{4.4}$$

We can then compute $\bar{P}$, the mean of the P_is, and $\bar{P}_e$:

$$\bar{P} = \frac{1}{N} \sum_{i=1}^{N} P_i \tag{4.5}$$

$$\bar{P}_e = \sum_{j=1}^{k} p_j^2 \tag{4.6}$$

Consider the following partially worked example, using the second row in (partially filled) Table 4.3. In this example, the token Vinken has been correctly

labelled with tags that resolve to tag stack NAME_PER by three annotators, and labelled with tags resolving to FIRST_PER by another annotator. For a fully worked example, see Wikipedia.[3]

Taking the first column, FIRST_PER in Table 4.3, the sum of all instances that category has been chosen, divided by the total number of annotation decisions:

$$p_1 = \frac{4 + 1 + 0 + \ldots + 0}{48134 \times 4} \tag{4.7}$$

Taking the second row:

$$P_2 = \frac{1}{4(4-1)}(1^2 + 3^2 + 0^2 + \ldots - 4) \tag{4.8}$$

We do this for each row and column, and can then calculate the sum of P_i to in turn calculate $\bar{P}$.

n_{ij}	1 (FIRST_PER)	2 (NAME_PER)	3 (INI_PER)	...	679	P_i
1 (Pierre)	4	0	0	...		1.0
2 (Vinken)	1	3	0	...		...
3 (,)	0	0	0	...		...
...	...	...	...	...		...
48,134	0	0	0	...		...
p_j				...		

Table 4.3: Table of values for computing Fleiss' kappa

According to Landis and Koch (1977), a Fleiss' kappa of between 0.60 and 0.79 shows "substantial agreement", and of 0.80 and above shows "almost perfect agreement". However, this metric is not without contention, and the metric is both the number of categories and the number of subjects, meaning a universal rating of 'good' agreement poses substantial problems. A smaller number of categories will result in a higher kappa value, so the kappa value

[3] https://en.wikipedia.org/wiki/Fleiss%27_kappa

achieved (0.834) for a very large number of categories shows particularly good inter-annotator agreement between the four annotators.

4.2 Annotation inconsistencies

In any annotation task, it is inevitable that annotation errors will occur. These typically arise for different reasons; partly, annotators' levels of interest or fatigue may affect specific annotation decisions, or ambiguities or gaps in the annotation guidelines may make a particular decisions difficult. Annotation meetings to discuss and address these issues do mitigate annotation errors, but some be unavoidable.

4.2.1 Annotation Errors

As with any large annotation task, there are bound to be simple annotation errors, caused by annotators accidentally entering an incorrect label. In this task, these have predominantly been caused by typing in an incorrect tag name, often starting with similar letters to the intended tag.

For example, here MONEY has been selected, probably due to hitting the 'M' key instead of 'N' for NAME: [[Mitsubishi]$_{\text{MONEY}}$ Estate]$_{\text{CORP}}$ In another example, CITY has been selected instead of the intended CARDINAL: [[8 716]$_{\text{CITY}}$ [%]$_{\text{UNIT}}$]$_{\text{PERCENT}}$

Consistency checking was performed, identifying anomalous structural patterns on identical token strings. These errors were corrected as far as possible. For one particularly problematic category, LANGUAGE, all instances were checked.

4.2.1.1 Errors identified in PER

In the PER category, the most problematic distinction was found to be caused by non-anglicised names, and by cases of double-barrel names not joined by hyphens. Both occur in the following example, Prime Minister Lee Kuan Yew, which was annotated in four different ways by the annotators.

[[Prime Minister]$_{\text{ROLE}}$ [Lee]$_{\text{NAME}}$ [Kuan]$_{\text{FIRST}}$ [Yew]$_{\text{MIDDLE}}$]$_{\text{PER}}$

[[Prime Minister]$_{\text{ROLE}}$ [Lee]$_{\text{NAME}}$ [Kuan Yew]$_{\text{FIRST}}$]$_{\text{PER}}$

[[Prime Minister]$_{\text{ROLE}}$ [Lee]$_{\text{FIRST}}$ [Kuan]$_{\text{MIDDLE}}$ [Yew]$_{\text{NAME}}$]$_{\text{PER}}$

[[Prime Minister]$_{\text{ROLE}}$ [Lee Kuan]$_{\text{FIRST}}$ [Yew]$_{\text{NAME}}$]$_{\text{PER}}$

Names occurring frequently in the corpus were manually checked for consistency.

4.2.1.2 Errors identified in LOC

Errors in the LOC category were dominated by cases of ambiguity in deciding between different entities with the same name (e.g. New York city or state). For example, given the following sentence: Average of top rates paid by major [New York]$_{\text{LOC}}$ banks it is not immediately clear whether CITY or STATE should be used to annotate the span of New York. These errors are corrected, described in Section 4.2.2.

In much the same way, in this sentence, Singapore could be referring to either the COUNTRY, GPE or CITY of Singapore. Further, in the sentence, it is acting adjectivally, even if is not in the usual adjectival form (Singaporean). . . . gives the [Singapore]$_{\text{LOC}}$ company more than. . . These were all reinspected and changed to CITY-STATE, described in Section 4.2.2.

4.2.1.3 Errors identified in ORG and FACILITY

The main discrepancy within the ORG category is in the structural marking of NAME mentions. Over the course of annotating, and based on discussions with the annotators, we modified the guidelines with respect to how to annotate these structures. For instance, marking separate NAME entities first, before coordinating over comma boundaries, as described in Section 3.3.2.3, was inconsistently done. All cases of 'NAME NAME' structure combinations were programmatically corrected and manually checked.

The other main error in the ORG category was caused by difficulty distinguishing between FACILITY and CORP. Indeed, in many cases the distinction between ORG and FACILITY is open to interpretation. Take, for instance, the following sentence:

Critics say Mitsubishi Estate 's decision to buy into Rockefeller reflects the degree to which...

Annotators for this disagreed on whether Rockefeller here was acting as CORP or BUILDING, or both. It is only clear from reading and retaining the 42 sentence article that in this sentence the CORP should be referenced. An earlier sentence in the same article, however, discussed the purchase of the Exxon Building , part of Rockefeller Center. These nuanced errors are very difficult to detect without doubly annotating the entire corpus. For sections where we have multiple annotations, any identified discrepancies such as this were adjudicated. However, it is likely that other errors still remain in the corpus.

4.2.1.4 Errors identified in MISC

Much like the earlier example of Rockefeller, MISC entities are frequently difficult to annotate. Take the sentence: Under the program , dubbed Chivas Class , customers who...

In this case, it is clear that Chivas Class is a program, but not immediately clear how it should be annotated. Chivas is the name of an alcohol, (which presents ambiguity between PRODUCT:DRUG and PRODUCT:FOOD), and should be marked with an embedded NAME entity. The larger entity span, the program, could conceivably be a WOA, a PRODUCT:OTHER or perhaps an ORG:OTHER, leading to possible annotations, of varying accuracy, including:

More complicated MISC entities also posed difficulties with complex structures, for example Macy 's Thanksgiving Day Parade, which was annotated as a WOA by one annotator, an EVENT by another, as a TV-SHOW by the third, and an EVENT nested in a TV-SHOW by the remaining annotator. In many respects, all four of these annotations are correct, as the televised event is quite a spectacle. In this case, most of the internal structure was consistent (Thanksgiving Day marked as a DATE and EVENT, and Macy 's marked as an CORP with nested NAME). Such inconsistencies are difficult to identify without multiple annotations. For entities with complex structure which occur frequently, their structure was checked and if necessary, corrected, using the structural comparison tool shown in Figure 4.3.

4.2.1.5 Annotation inconsistencies in TIMEX and NUMEX

Annotators had particularly high agreement in TIMEX and NUMEX categories. The main disagreements in TIMEX had to do with when a temporal expression should be considered a DATE or a DURATION, for instance the phrase later in the day.

In NUMEX, the RATE category proved comparatively problematic. Compared to other NUMEX expressions, these phrases are particularly complicated,

so the scope for a single error to affect the larger entity is higher. In one example from section 02, one annotator analysed cubic feet as [cubic [feet]$_{\text{UNIT}}$]$_{\text{UNIT}}$, nesting one UNIT inside the other.

The other discrepancy between annotators within the NUMEX category is the inclusion of a CARDINAL span over adjacent CARDINAL and MULT entities. That is, that [500]$_{\text{CARDINAL}}$ [million]$_{\text{MULT}}$ should be combined to make one larger CARDINAL span. The decision to add this span came only after annotation had started, so this annotation span was programmatically added to any entities in which it was missing.

4.2.2 Annotation Post-Process

In order to increase the reliability and consistency of the corpus, a number of post-processing steps have been conducted on the annotated corpus. In addition to the errors identified in the multiple-annotated section 02, one hundred sentences were randomly sampled from the annotated corpus, and the accuracy of entities therein was checked. Of these 100 sentences, 17 contained no entities, a further 63 contained only correctly annotated entities. The remaining 20 sentences contained a total of 33 annotation errors. Some of these errors were caused by changes to the annotation guidelines that had not yet been systematically applied (e.g. as early as being considered a QUAL, Group being incorrectly labelled as JARGON), while others were annotation errors (Fallon being labelled as REL instead of NAME).

While checking these sentences gave an overall feel for the consistency of the annotations, they did not include examples of each annotation category. Therefore, for categories for which consistency checking coverage was inadequate (fewer than 5 instances were checked), supplementary sentences which contained that category's tag were randomly selected. The results of this survey can be seen in Table 4.4.

Category	Correct	Incorrect	Category	Correct	Incorrect
AGE	1	0	NAMEMOD	3	0
AIRPORT	3	0	NATIONALITY	8	1
ALBUM	3	0	NATURAL-DISASTER	2	0
ANIMATE	3	0	NICKNAME	2	1
ARMY	3	1	NORP:OTHER	3	4
ATTRACTION	2	1	NORP:POLITICAL	4	0
AWARD	2	1	NUMDAY	5	0
BAND	3	0	OCEAN	3	0
BOOK	3	0	ORDINAL	6	0
BRIDGE	4	0	CORP	37	0
BUILDING	3	0	ORG:EDU	3	1
CARDINAL	50	1	ORG:OTHER	5	2
CITY	11	1	ORG:POLITICAL	2	2
CONCERT	2	0	ORG:RELIGIOUS	2	2
CONTINENT	3	0	PAINTING	3	0
JARGON	8	1	PERCENT	11	0
COUNTRY	6	0	PERIODIC	5	1
DATE	40	4	PERSON	25	1
DATE:OTHER	3	0	PLAY	3	0
DAY	4	0	PRODUCT:DRUG	2	1
DISEASE	2	1	PRODUCT:FOOD	3	0
DURATION	22	1	PRODUCT:OTHER	1	2
ELECTRONICS	3	0	QUAL	19	3
ENERGY	3	0	QUANTITY	2	1
EVENT	6	0	RATE	2	1
FACILITY	3	0	REGION	3	0
FILM	3	0	REL	8	3
FIRST	14	1	RELIGION	3	0
FOLD	2	1	RIVER	3	0
FUND	3	1	ROLE	3	0
GOD	3	0	SCINAME	3	0
GOVERNMENT	12	2	SEASON	3	0
GPE	3	0	SONG	3	0
GRP:LOC	2	1	SPACE	3	0
GRP:ORG	3	0	SPEED	3	0
GRP:PER	2	1	SPORTS-SEASON	2	1
HON	13	0	SPORTS-EVENT	2	1
HOSPITAL	3	0	SPORTS-TEAM	4	0
HOTEL	3	1	STADIUM	3	0
HURRICANE	4	0	STATE	3	1
INDEX	3	0	STATION	1	0
INITIALS	3	0	STREET	4	0
IPOINTS	2	1	SUBURB	3	0
LANGUAGE	3	0	TEMPERATURE	2	0
LAW	2	1	TIME	2	1
LOCATION:OTHER	3	1	TV-SHOW	3	0
MEDIA	4	0	UNIT	24	2
MIDDLE	2	1	VEHICLE	4	0
MONEY	14	0	WAR	3	0
MONTH	7	0	WEAPON	3	0
MULT	8	0	WEIGHT	3	1
MUSEUM	2	1	WOA	2	1
NAME	37	4	YEAR	12	0
			Total	632	63

Table 4.4: Per category results of consistency checking in the corpus.

In total, around 700 entities were checked, with 640 entities being found to be correct, and 66 instances of errors detected. Note that these errors are, in many cases, double counted. That is, if an entity is incorrectly labelled, it counts as both an error for the incorrect label (much like a false positive) and an incorrect annotation for the absent label (false negative).

In this analysis, we identified categories and rules that had high levels of inconsistency, either due to underspecification in the annotation guidelines or genuine linguistic ambiguity on an entity level. A combination of manual and automatic corrections were applied, covering both annotation spec changes and annotation mistakes. Illustrative examples of these errors and fixes are given below.

Addition of CITY-STATE tag Locations such as Hong Kong, Singapore, and Monaco present challenges to human readers and are often used not just ambiguously but in a sense that conveys both their city and country properties. Given the inconsistencies in annotation of these named entities, we elect to add a new category: CITY-STATE which encompasses both these senses and removes the need for the annotator to make an arbitrary classification decision.

CITY GPE confusion Similar to the previous discussion are cases of CITY and GPE that share the same name (e.g. New York, Honolulu). Sometimes it is clear whether or not the CITY or STATE is being referred to. In cases of ambiguity, we elect to label the token(s) as the larger of the entities. All occurrences of New York, Honolulu and Washington were manually checked.

In certain rare cases, a correct analysis is not possible. For example, the coordination of GPE types in the phrase City and County of Honolulu (`WSJ0704_10`) means we cannot correctly capture both concurrent meanings.

Sentences where D.C. refers to Washington D.C. are further checked and analysed following to the principle that D.C. is referring to the GPE, which

itself is standing in for the CITY. Further complicating this analysis are cases where the CITY is acting as a spokesperson for its GOVERNMENT, e.g. Sentence WSJ1256_6: The FBI 's role is to complement the D.C. initiative through.... In such cases, entity nesting is used to represent this, e.g. the [[[D.C.]$_{\text{GPE}}$]$_{\text{CITY}}$]$_{\text{GOV}}$ initiative.

GOVERNMENT should be within country only Some occurrences of large multi-national named entities, such as the United Nations and the European Commission were inconsistently labelled as ORG:OTHER or GOVERNMENT. We decided that GOVERNMENT should only refer to (at most) COUNTRY level entities, and should not span multiple countries. These entities should instead be labelled as ORG:OTHER. The guidelines were clarified, and all instances of the United Nations and the European Commission were corrected.

Cardinal directions referring to elided LOC Instances of cardinal directions, often used embedded in the names of organisations. In these cases, they refer to an elided LOC (usually REGION) and act adjectivally. Therefore, they should be labelled as NORP:OTHER. They were found to be inconsistently annotated. All instances of north western, northwest, norwest, northeastern, northeast, northern, north, southern, southwestern, southwest, southeast, southeastern, midwest, midwestern, eastern, east, western, west in the corpus were manually corrected.

Cities analysed as CORP for stock exchanges Stock exchanges are frequently referred to by the name of the city in which they are located. In these cases, the token(s) should be marked as CITY embedded within CORP. This was found to be inconsistently annotated and all instances of major cities with stock exchanges[4] were manually corrected.

[4]including Brussels, Sydney, Singapore, Wellington, Hong Kong, Manila, Seoul, Taipei, Jakarta, Bangkok, Milan, Stockholm, Frankfurt, Madrid, Amsterdam, Paris, Zurich, London, Tokyo, Osaka, Chicago, New York and Toronto

Named Entities with complex structures Additionally, a number of named entities which have complex structural composition, (e.g. the [[[Wall Street]$_{\text{STREET}}$]$_{\text{GRP:ORG}}$ Journal]$_{\text{MEDIA}}$) were manually checked for consistency across sentences.

Certain Named Entities that occurred numerous times and which were non-trivial to annotate were identified. These entities were either identified by annotators as problematic during the annotation process or were found to be inconsistently annotated during corpus evaluation and analysis. An additional pass was made over sentences which contained these entities to ensure these cases were consistently annotated. Particular care was taken with entities which are homonymous, such as the state and city of New York , and U.S. acting as a country or nationality.

4.2.3 Effect of post-processing corpus

In order to gauge the scale of the consistency improvements outlined in the above sections, we calculate the agreement between these two versions of the corpus before and after performing these changes using standard CoNLL evaluation. This is seen in Table 4.5, showing a 10% and 11% difference in recall and precision between the versions, which demonstrates this considerable post-processing effort was worthwhile.

	Precision	Recall	F_1-score
Agreement with corpus before fixes	90.2	89.0	89.6

Table 4.5: A measure of inter-annotator agreement between the original corpus and the finished corpus, with error and consistency fixes.

4.3 Corpus analysis

The resulting NNE corpus includes a large number of entities of substantial depth, with more than half of all entity annotations occurring inside another entity. The numbers of annotations occurring at each entity *depth* is shown in Table 4.6, and is found to be considerably large. In the GENIA corpus (Alex et al., 2007), around 17% of entities are embedded inside another entity, 18% of entities having at least one layer of nesting in the PKU Chinese corpus (Fu and Luke, 2005), and 9.4% of entities found to have nesting in the Historical Archive Corpus (Byrne, 2007).

Of the 118495 top-level entities, only 46949 (39.6%) did not have any nested structure embedded. The remaining 71546 entities contain 161265 entity annotations, averaging 2.25 structural entities per each of these top-layer entities. Interestingly, comparing the raw numbers of annotations that occur as top-level entities or at a depth of 1, we can see that more annotations occur at one layer of nesting than in total at the top level. Considering the three most frequent types of top-level entities, CORP, DATE and PER, and their frequent *template* structural composition, discussed more in Section 4.3.2, (for example, NAME + JARGON → CORP, REL + DURATION → DATE, HON + NAME → PER), this is not unsurprising.

Figure 4.6 shows the two deepest entity annotations, UNIT and CITY-STATE, and demonstrates how easily very complex substructures can manifest. In this example, a number of common templates can be seen, including: UNIT + CARDINAL → MONEY, QUAL + MONEY → MONEY, MONEY + a or per + DURATION → RATE. The unrestricted nesting principles have allowed us to capture a complex entity using straightforward rules.

Depth	Number	%	3 most frequent categories
1	118495	42.4%	CORP (22687), DATE (15963), PER (13451)
2	119661	42.8%	NAME (21359), CARDINAL (21198), UNIT (14732)
3	36762	13.1%	CARDINAL (12782), NAME (6407), MULT (5939)
4	4486	1.6%	CARDINAL (1715), MULT (1075), NAME (723)
5	354	0.1%	CARDINAL (165), MULT (100), UNIT (62)
6	2	0.0%	UNIT (1), CITY-STATE (1)

Table 4.6: Number of entities at each layer of nesting, with the most frequent three categories occurring at each nesting layer.

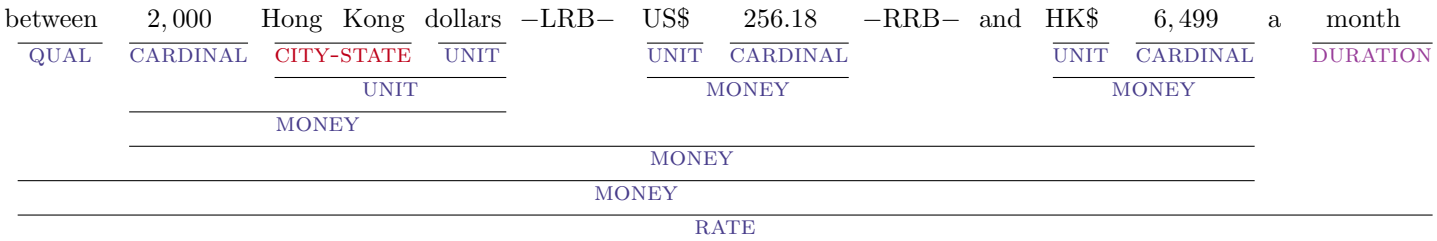


Figure 4.6: Example of an entity with 6 layers of nested entity annotation, with tokens Hong Kong and dollars at the sixth layer of nesting.

4.3.1 Entity Exemplars

Table 4.7 shows that 40 most frequent entity categories, the percentage of all entities that they represent, and the three most frequent examples of those entities. The most frequent entity annotation is CARDINAL, which we have seen from Table 4.6 occurs very frequently within other entities. Similarly, NAME occurs frequently within other entities, with the three most frequent tokens labelled NAME being Bush ([[Mr]$_{\text{HON}}$ [Bush]$_{\text{NAME}}$]$_{\text{PER}}$), Dow and Jones (from the [[[[Dow]$_{\text{NAME}}$ [Jones]$_{\text{NAME}}$]$_{\text{NAME}}$]$_{\text{MEDIA}}$ Industrial Average]$_{\text{INDEX}}$). It is only after CARDINAL and NAME that we get to CORP, the most frequent of

top-level entities, accounting for 19.1% of all top-level entities, but only 8.4% of all entity annotations at any level of nesting.

4.3.2 Template analysis

To get an overview of common types of entity nesting, the corpus was analysed, and the types of entities with embedded entities, a concept introduced earlier in this section as *templates*, were captured. Table 4.8 shows the most frequent 47 of these template rules, amounting to all such rules which occur more than 200 times in the corpus. Each entity span, or non-entity span (shown as an *o*) can consist of multiple tokens, and may themselves contain nesting. The table shows the rule, for example MONEY being formed by a UNIT and a CARDINAL entity (e.g. $[[\$]_{\text{UNIT}}\ [10]_{\text{CARDINAL}}]_{\text{MONEY}}$) , how many times this template rule occurs in the corpus, and how many entities of parent (in this case, MONEY) type occur overall. The final two columns show the percentage of all template rules this particular rule makes up, and a cumulative running tally. We can see that these 47 rules make up more than 80% of all nesting rules in the corpus, and 50% are made up by the most frequent 10 alone, meaning that the majority of entity nestings are a small group of frequent templates. There is also a very long tail of templates, with a total of 1935 rules in total generated from the corpus.

Tag	%	Examples
CARDINAL	15.66	one, two, quarter
NAME	10.20	Bush, Dow, Jones
CORP	8.35	UAL, [New York]$_{\text{CITY}}$ Stock Exchange, Big Board
UNIT	6.90	$, %, cents
DATE	6.25	yesterday, [this]$_{\text{REL}}$ [year]$_{\text{DURATION}}$, [Friday]$_{\text{DAY}}$
PER	5.33	[Bush]$_{\text{NAME}}$, [Mr.]$_{\text{HON}}$ [Bush]$_{\text{NAME}}$, [President]$_{\text{ROLE}}$ [Bush]$_{\text{NAME}}$
DURATION	4.91	year, years, [quarter]$_{\text{CARDINAL}}$
MONEY	4.52	[$]$_{\text{UNIT}}$ [[1]$_{\text{CARDINAL}}$ [billion]$_{\text{MULT}}$]$_{\text{CARDINAL}}$, [$]$_{\text{UNIT}}$ [[100]$_{\text{CARDINAL}}$ [million]$_{\text{MULT}}$]$_{\text{CARDINAL}}$, [$]$_{\text{UNIT}}$ [[200]$_{\text{CARDINAL}}$ [million]$_{\text{MULT}}$]$_{\text{CARDINAL}}$
MULT	2.81	million, billion, thousands
FIRST	2.43	John, Robert, James
CITY	2.40	New York, Chicago, London
PERCENT	2.34	[10]$_{\text{CARDINAL}}$ [%]$_{\text{UNIT}}$, [15]$_{\text{CARDINAL}}$ [%]$_{\text{UNIT}}$, [20]$_{\text{CARDINAL}}$ [%]$_{\text{UNIT}}$
REL	2.21	last, this, next
JARGON	1.99	Corp., Inc., Co.
HON	1.97	Mr., Ms., Mrs.
NATIONALITY	1.86	U.S., American, Japanese
GOVERNMENT	1.67	Treasury, Congress, Senate
COUNTRY	1.45	U.S., Japan, China
QUAL	1.40	about, more than, at least
YEAR	1.22	1988, 1987, 1989
MONTH	1.21	Oct., September, August
STATE	1.16	California, Texas, Calif.
ORDINAL	0.93	first, third, First
IPOINTS	0.86	[1]$_{\text{CARDINAL}}$, [1 14]$_{\text{CARDINAL}}$, [78]$_{\text{CARDINAL}}$
ROLE	0.79	President, Chairman, Sen.
RATE	0.77	[[five]$_{\text{CARDINAL}}$ [cents]$_{\text{UNIT}}$]$_{\text{MONEY}}$ a share, [$]$_{\text{UNIT}}$ 300-a-share, [[10]$_{\text{CARDINAL}}$ [cents]$_{\text{UNIT}}$]$_{\text{MONEY}}$ a share
MEDIA	0.61	[[Dow]$_{\text{NAME}}$ [Jones]$_{\text{NAME}}$]$_{\text{NAME}}$, CBS, Time
DAY	0.58	Friday, Monday, Tuesday
NUMDAY	0.53	30, 1, 31
INI	0.52	J., A., E.
NORP:OTHER	0.45	European, Western, Eastern
ORG:OTHER	0.39	EC, [European]$_{\text{NORP:OTHER}}$ Community, OPEC
PERIODIC	0.38	annual, daily, quarterly
REGION	0.31	West, New England, Northeast
NORP:POLITICAL	0.26	D., Democrats, Democratic
AGE	0.24	[44]$_{\text{CARDINAL}}$, [45]$_{\text{CARDINAL}}$, [65]$_{\text{CARDINAL}}$
INDEX	0.23	[[[Dow]$_{\text{NAME}}$ [Jones]$_{\text{NAME}}$]$_{\text{NAME}}$]$_{\text{MEDIA}}$ Industrial Average, [S&P]$_{\text{CORP}}$ [500]$_{\text{CARDINAL}}$, [[Dow]$_{\text{NAME}}$]$_{\text{MEDIA}}$
PRODUCT:OTHER	0.23	Class A, Series [[1989]$_{\text{YEAR}}$]$_{\text{DATE}}$, CDs
STREET	0.17	Wall Street, [Wall]$_{\text{NAME}}$ Street, [Fifth]$_{\text{ORDINAL}}$ Avenue
GRP:ORG	0.16	[Wall Street]$_{\text{STREET}}$, [Hollywood]$_{\text{SUBURB}}$, [[Wall]$_{\text{NAME}}$ Street]$_{\text{STREET}}$

Table 4.7: The 40 most frequent entity labels in our Wall Street Journal NNE corpus, the percentage of each label's occurrences, and the three most frequent examples, with substructure marked.

Template Rule	Children	Count	of	% Total	Cumul've
MONEY	→ UNIT + CARDINAL	9021	12580	9.73%	9.73%
CARDINAL	→ CARDINAL + MULT	7470	10103	8.06%	17.79%
PERCENT	→ CARDINAL + UNIT	6038	6510	6.51%	24.30%
PER	→ HON + NAME	5367	14911	5.79%	30.09%
CORP	→ NAME	5001	14664	5.39%	35.48%
PER	→ FIRST + NAME	4572	14911	4.93%	40.41%
DATE	→ REL + DURATION	3261	14977	3.52%	43.93%
DURATION	→ CARDINAL + DURATION	2532	4081	2.73%	46.66%
DATE	→ YEAR	2190	14977	2.36%	49.02%
CORP	→ o + JARGON	2002	14664	2.16%	51.18%
CARDINAL	→ QUAL + CARDINAL	1939	10103	2.09%	53.27%
MONEY	→ CARDINAL + UNIT	1880	12580	2.03%	55.30%
IPOINTS	→ CARDINAL	1863	2221	2.01%	57.31%
PER	→ NAME	1686	14911	1.82%	59.13%
RATE	→ MONEY + o	1410	2107	1.52%	60.65%
PER	→ ROLE + PER	1391	14911	1.50%	62.15%
DATE	→ MONTH	1281	14977	1.38%	63.53%
CORP	→ NAME + o	1219	14664	1.31%	64.84%
DATE	→ DAY	1144	14977	1.23%	66.08%
NAME	→ NAME + NAME	1144	2600	1.23%	67.31%
CORP	→ NAME + o + JARGON	1139	14664	1.23%	68.54%
DATE	→ DURATION + REL	1070	14977	1.15%	69.69%
DURATION	→ CARDINAL	1068	4081	1.15%	70.84%
MONEY	→ QUAL + MONEY	1014	12580	1.09%	71.94%
DATE	→ MONTH + NUMDAY	953	14977	1.03%	72.97%
CORP	→ NAME + JARGON	909	14664	0.98%	73.95%
PER	→ FIRST + INI + NAME	839	14911	0.90%	74.85%
DATE	→ ORDINAL + DURATION	509	14977	0.55%	75.40%
DATE	→ DURATION	507	14977	0.55%	75.95%
DATE	→ REL + DATE	499	14977	0.54%	76.48%
NAME	→ NAME + o + NAME	489	2600	0.53%	77.01%
CORP	→ CITY	482	14664	0.52%	77.53%
CORP	→ CITY + o	465	14664	0.50%	78.03%
CORP	→ NATIONALITY + o	371	14664	0.40%	78.43%
DATE	→ MONTH + NUMDAY + o + YEAR	344	14977	0.37%	78.80%
GRP:ORG	→ STREET	340	401	0.37%	79.17%
IPOINTS	→ CARDINAL + UNIT	315	2221	0.34%	79.51%
DATE	→ MONTH + YEAR	314	14977	0.34%	79.85%
MEDIA	→ NAME	290	661	0.31%	80.16%
CARDINAL	→ CARDINAL + o + CARDINAL	284	10103	0.31%	80.47%
NAME	→ FIRST + NAME	280	2600	0.30%	80.77%
CORP	→ MEDIA	274	14664	0.30%	81.07%
PER	→ FIRST	239	14911	0.26%	81.32%
GOVERNMENT	→ BUILDING	233	837	0.25%	81.58%
AGE	→ CARDINAL	228	438	0.25%	81.82%
CARDINAL	→ MULT	224	10103	0.24%	82.06%
NAME	→ NAME + NAME + NAME	216	2600	0.23%	82.30%
DATE	→ REL + MONTH	208	14977	0.22%	82.52%

Table 4.8: Template rules occurring more than 200 times in the corpus, showing the number of times each occurred, the number of entities of that (parent) type in the corpus, the percentage of all embedding rules that this contributed to, and the cumulative total of all such template rules.

4.4 Comparison to BBN corpus

To assess the quantity and type of changes between the underlying BBN annotations and our final NNE corpus, we analyse the differences between the spans in the two corpora.

In Table 4.9 we compare the number of entities in BBN and NNE which use the BBN label as a top-level category in NNE, (*Same label*), which use the BBN label as a nested label within the entity (*Same label in stack*) or which use a different category label (*Different label*) and do not contain the BBN label in the nested stack. This analysis is further split between entities in BBN and NNE which have the same span, and entities which have tokens included or excluded from the entity span (*Larger span in* BBN and *Smaller span in* BBN respectively).

From Table 4.9 we can see that a large number of entities in NNE share the same token span and top-level category as BBN. Of the 119280 non-*Descriptor* entities in BBN, 72% occur in the aligned NNE corpus, with the same entity bounds and same label. Many of these have had also internal structure added.

4294 entities which are not *Descriptors* occur in BBN, and for which a matching entity in NNE was not found. Of these, 2484 (more than 50%) were of a subtype of SUBSTANCE, either OTHER, FOOD, CHEMICAL or DRUG. A further 560 (13%) were either of type ANIMAL or PLANT. 309 DATE:DATE entities were not found, which include phrases now, season and a holiday. 130 DISEASE entities were also not included in the final NNE corpus, including diseases, illness and nausea.

The 20 most frequent mismatches between BBN and NNE categories, where the BBN category is not found anywhere in the NNE stack, for entities with exact matching spans are shown in Table 4.10. The most common discrepancy is of entities analysed as GPE:COUNTRY in BBN and NATIONALITY in NNE.

Span match	Same label	Label in stack	Different label	Total
Exact span	85617	3428	6397	95442
Larger span in NNE	10377	4003	1302	15682
Smaller span in NNE	3102	344	416	3862
Total	99096	7775	8115	114986

Table 4.9: Analysis of matching entities in BBN and our final aligned NNE corpus. 4294 entities occurred in BBN which did not occur in NNE (with a further 51017 *DESC* entities excluded). 8464 entity spans were found in NNE that do not exist in BBN entities occurred in NNE which did not occur in BBN. In total, BBN has 170297 entities, and NNE has 117242 top-level entities.

This is due to BBN not distinguishing between adjectival and nominal forms of words such as U.S.. The BBN guidelines[5] state:

> "The distinction between NORP and other types is morphological. American and Americans is a nationality, while America and US are GPEs, regardless of context."

We, however, have analysed these according to their use in context, making a distinction between whether U.S. is used as a NATIONALITY (i.e. meaning American) or as a country (America).

A number of category differences are caused by the introduction of new categories, including MEDIA from ORG:CORP, TV-SHOW and FILM from WORK_OF_ART:OTHER, ARMY from ORG:GOVERNMENT, SPORTS-TEAM from ORG:OTHER and ELECTRONICS from PRODUCT:OTHER. Some of these, such as TV-SHOW, FILM and ELECTRONICS, were added as a distinction from existing OTHER categories; that is, they represent a specific subtype of a larger category that was previously not separated out. Other categories have been included because they are often used in a different way from the other category type. For instance, entities that belong to the MEDIA category

[5] https://catalog.ldc.upenn.edu/docs/LDC2005T33/BBN-Types-Subtypes.html

have viewers in addition to stakeholders. The SPORTS-TEAM category was separated from ORG:OTHER because the names of sports teams are often quite different from the names of other organisations, often including location elements (Los Angeles Dodgers) which are occasionally used metonymously (for example, Oakland won last night's game against the 49ers). Still other categories, such as ARMY or STADIUM being split from FAC:BUILDING, were split to enable simpler annotation decisions.

Other clarifications to our guidelines resulted in shifts between the BBN and our categories. ORG:GOVERNMENT in the BBN scheme includes entities which exist above the government level, such as the European Union. We instead classify these separately as ORG:OTHER, allowing us to better capture idiosyncratic metonymous references to GOVERNMENT (e.g. The [[White House]$_{\text{BUILDING}}$]$_{\text{GOV}}$, [[Washington]$_{\text{CITY}}$]$_{\text{GOV}}$ announced...).

Another example of a clarification to guidelines resulting in a discrepancy is the ORG:GOVERNMENT CORP change, caused by entities such as Freddie Mac (The Federal Home Loan Mortgage Corporation), which is a public government-sponsored enterprise. Although it is linked to the U.S. government, it is also a publicly traded company so is classified as an CORP.

A number of changes are within the TIMEX categories. The TIME category in BBN included:

> Any time ending with A.M. or P.M. The a.m. and p.m. must be tagged along with the numbers. Other times of day (units smaller than a day) and time durations may be marked: morning, noon, night, 3 hours.

The distinction between time durations of less than 24 hours and those greater than 24 hours, which are labelled as DURATION, seems arbitrary.

> Duration: answers the question "how long" and includes a period of time (2 years, centuries, 16 weeks, less than 2 years, 6 months, 52-week).

We therefore keep the TIME category for only specific mentions of a time, and label all durations as DURATION.

We define a new TIMEX category, PERIODIC, split from DATE:OTHER. We also fix a number of inconsistencies between the BBN categories DATE:DATE and DATE:DURATION caused by incorrect category choice.

We also introduce a new category type: GRP, denoting *group* in a category. GRP:ORG, GRP:LOC and GRP:PER denote groups of entities of the same type, which though not officially organised, are recognised. For example, [Hollywood]$_{\text{GRP:ORG}}$, when not referring to the suburb, refers to a group of film and movie related organisations. The [Carolinas]$_{\text{GRP:LOC}}$ are annotated as GRP:LOC, rather than GPE:STATE_PROVINCE in BBN, which does not capture the fact that both [North Carolina]$_{\text{STATE}}$ and [South Carolina]$_{\text{STATE}}$ are being referenced.

Count	BBN	NNE	Reason	Example
1249	GPE:COUNTRY	NATIONALITY	New distinction	U.S., U.K., New Zealand
910	ORG:CORPORATION	MEDIA	New category	CBS, NBC, ABC
726	DATE:OTHER	PERIODIC	New distinction	annual, daily, quarterly
385	NORP:NATIONALITY	NORP:OTHER	Clarification	European, Asian, Texans
313	TIME	DURATION	New distinction	afternoon, morning, hours
236	DATE:DATE	DURATION	Fix Inconsistency	one month, two years, three years
156	WORK_OF_ART:OTHER	TV-SHOW	New category	Batibot, Teddy Z, Cosby
125	ORG:GOVERNMENT	ORG:OTHER	Clarification	EC, European Community, U.N.
113	ORG:GOVERNMENT	ARMY	New category	Navy, Air Force, Army
109	ORG:OTHER	SPORTS-TEAM	New category	Giants, Cowboys, A 's
104	PRODUCT:OTHER	ELECTRONICS	New category	Cray-3, 486, 80486
89	DATE:DURATION	DATE	Fix Inconsistency	1988, between 1990 and 1994
87	ORG:OTHER	CORP	Fix Inconsistency	Merc, Manville, Farmers
85	ORG:CORPORATION	GOVERNMENT	Fix Inconsistency	RTC, Securities and Exchange Commission
81	GPE:COUNTRY	CITY-STATE	New category	Hong Kong, Singapore, Monaco
80	TIME	DATE	Changed category	last night, this morning, tonight
80	WORK_OF_ART:OTHER	FILM	New category	Batman, Sidewalk Stories, Rainman
73	DATE:DATE	PERIODIC	New category	each year, every day, every year
61	ORG:GOVERNMENT	CORP	Clarification	Freddie Mac, Fannie Mae, Ginnie Mae
53	GPE:CITY	GRP:ORG	New category Type	Hollywood

Table 4.10: Analysis of the 20 most frequent category mismatches for entities with the same spans in BBN and NNE.

A further nearly 3500 entities share the same span in BBN and NNE, have a different top-level category, but contain the expected (BBN) category within the nested structure of NNE category labels. Table 4.11 shows the most frequent entity confusions for the most frequent of these cases. The majority of these instances are cases of metonymy, for example [[Washington]$_{\text{CITY}}$]$_{\text{GOV}}$ acting as the US Government, or elided context, for example [[100]$_{\text{CARDINAL}}$]$_{\text{MONEY}}$ from a larger phrase The price rose from [[50]$_{\text{CARDINAL}}$ [dollars]$_{\text{UNIT}}$]$_{\text{MONEY}}$ to [[100]$_{\text{CARDINAL}}$]$_{\text{MONEY}}$.

Two cases that are slightly different are those of PERSON/GRP:PER and DATE:DATE/DATE:OTHER. The former stems from the introduction of the GRP:PER category, which captures unofficial groups of people, in many cases family groups. These have nested PERSON entities. The latter DATE:DATE / DATE:OTHER confusion is caused by our nesting being able to capture with more precision the specific tokens which carry temporal meaning. For example, BBN analysed recent months as DATE:DURATION, but we split those tokens, keeping the DURATION label only on the [months]$_{\text{DURATION}}$ token: [[recent]$_{\text{REL}}$ [months]$_{\text{DURATION}}$]$_{\text{DATE}}$.

4.4.1 Entity spans that do not match

To analyse bounds that had changed slightly, we also looked at spans that were off by one token, that is included one more or one fewer tokens at either the beginning or end of the entity. The main reasons for entities having either a larger or smaller span in NNE as compared to BBN are due to differences caused by nesting entities, and tokenisation issues.

Count	BBN	NNE	Example
1826	CARDINAL	IPOINTS	85
578	GPE:CITY	CORP	Tokyo, Chicago
170	DATE:DURATION	DATE	recent months
151	GPE:CITY	GOVERNMENT	Washington
132	CARDINAL	MONEY	millions, 100
97	DATE:DATE	DATE:OTHER	today, those days
70	PERSON	GRP:PER	Lehmans, Rothschilds
69	MONEY	RATE	37.5 cents

Table 4.11: Analysis of entity confusions occurring more than 50 times with the same spans and correct span embedded within the nested NNE span.

Count	BBN	NNE	E.g. BBN	E.g. NNE
7083	PERSON	PER	Bush	[Mr.] [[Bush]]
1874	MONEY	RATE	five cents	[[five] [cents]] a share
614	DATE:DATE	DATE	yesterday	[yesterday] [morning]
601	CARDINAL	CARDINAL	one	one-time
450	CARDINAL	IPOINTS	190	[190] points†
319	GPE:CITY	CITY	New York	New York-based
298	ORG:GOV	GOV	Senate	House-Senate
242	ORG:CORP	CORP	Cray	$[\text{Cray}]_{\text{NAME}}$ Research
209	PERCENT	PERCENT	50 %	below [[50] [%]]
200	MONEY	MONEY	$ 10,000	[[$] [10,000]] to [[$] [[1] [million]]]

Table 4.12: Most common confusion matrix (those occurring more than 200 times) of entities with larger spans in NNE than BBN. †Occasionally, the embedded CARDINAL tag was missing, as an annotation error.

NNE span is larger than BBN span Looking at the entity spans which have grown in NNE from the original spans in BBN, the most frequent types of which are shown in Table 4.12, by far the largest cause for change is the inclusion of ROLE and HON spans inside larger PER spans. That is, entities such as [[Mr.]$_{\text{HON}}$ [Vinken]$_{\text{NAME}}$]$_{\text{PER}}$, which in BBN are analysed as Mr. [Vinken]$_{\text{PER}}$, with Mr. left unannotated. In total, 7084 of these cases are of type PER, amounting for more than 68% of spans that are larger in NNE than in BBN.

The impact of nesting entities is seen in other categories, especially NUMEX. BBN spans such as [less than one]$_{\text{CARDINAL}}$ in [two]$_{\text{CARDINAL}}$ or [one]$_{\text{CARDINAL}}$ in [four]$_{\text{CARDINAL}}$ have had the CARDINAL span expanded out to capture the actual number: [less than [one]$_{\text{CARDINAL}}$ in [two]$_{\text{CARDINAL}}$]$_{\text{CARDINAL}}$ and [[one]$_{\text{CARDINAL}}$ in [four]$_{\text{CARDINAL}}$]$_{\text{CARDINAL}}$ respectively. Similarly, words which affect the cardinal are included inside the larger CARDINAL span: below [50 %]$_{\text{PERCENT}}$ is analysed as [[below]$_{\text{QUAL}}$ [[50]$_{\text{CARDINAL}}$ [%]$_{\text{UNIT}}$]$_{\text{PERCENT}}$]$_{\text{PERCENT}}$.

The second most common cause of larger spans in NNE is tokenisation issues. Differences in hyphenation alone amount for a further 1358 (13%) of larger span issues. For example, what is analysed in BBN as [New York]$_{\text{CITY}}$-based, when analysed with tokenisation consistent with the WSJ is analysed [New York-based]$_{\text{CITY}}$. Similarly, [second]$_{\text{ORDINAL}}$-largest and [second-largest]$_{\text{ORDINAL}}$.

Other tokenisation differences also impact this category, such as sentence final full stops being removed from entities in BBN [Ariz]$_{\text{STATE}}$., compared to our analysis: [Ariz.]$_{\text{STATE}}$ which is consistent with WSJ tokenisation.

Another 574 spans where NNE is larger than BBN are caused by including the determiner the in the span, to be able to match syntactic structure. This necessity is further discussed in Section 5.3.2.1.

Of the entities which are larger in NNE than in BBN and in which the categories to not match, 133 are caused by TIME DATE distinction, where the BBN span is morning, and the NNE span has been expanded to [[yesterday]$_{\text{DATE}}$

[morning]$_{\text{DURATION}}$]$_{\text{DATE}}$. The names of INDEX, such as [[[[Dow] [Jones]]$_{\text{NAME}}$]$_{\text{MEDIA}}$ industrials]$_{\text{INDEX}}$, also occur frequently (72 instances of ORG:CORP / INDEX confusion), due to the inclusion of a new INDEX category, and an annotation decision to include lower case words in instances of INDEX.

NNE span is smaller than BBN span 3862 entities were identified that have decreased in size from BBN to the NNE corpus. The most frequent types of these entities are shown in Table 4.13. The majority of these are again due to differences in tokenisation, and the inclusion or exclusion of determiners, the treatment of which is inconsistent in BBN. Take the determiner the and TIMEX entities alone, the precedes 1060 TIMEX entities, and is included in the TIMEX entity 3185 times. The NNE annotations are also inconsistent with the inclusion or exclusion of determiners, but this is a consequence of combining the entities with syntactic constituents, discussed in Section 5.3.2.1, whereas BBN does not guarantee such reconciliation with syntactic structure.

A common tokenisation difference is the inclusion of sentence final full stops in BBN entities. These full stops act as both sentence ending marker and as part of the last token in the sentence (e.g. Inc. or Calif.). Including the sentence final full stop causes problems when combining it with a syntactic constituent, as further described in Section 5.3.2.7, so we elect to not include these in the entity span.

Count	BBN	NNE	E.g. BBN	E.g. NNE
1732	DATE:DATE	DATE	the third quarter	[third] [[quarter]]
536	ORG:CORP	CORP	Telerate Systems Inc .	Telerate Systems [Inc]
346	DATE:DUR	DATE	the weekend	weekend
243	DATE:DUR	DURATION	the nine months	[nine] [months]
226	CARDINAL	CARDINAL	only one	one
178	DATE:DATE	DURATION	the day	day
114	GPE:STATE	STATE	Calif .	Calif
109	MONEY	MONEY	some $ 100 million	[$] [[100] [million]]

Table 4.13: Most common entities (those occurring more than 100 times) with smaller spans in NNE than BBN. Of these, some are labelled with the same category, and some, especially TIMEX entities, are labelled with different categories.

NNE span doesn't exist in BBN In addition to the 4294 entities from BBN for which no equivalent span was found in NNE, 8464 top-level entities which exist in NNE were not found in BBN. The most frequent types of these are given in Table 4.14. Many of these are due to differences in annotation guidelines, especially with respect to new categories, for instance RATE, which often includes what was a MONEY or CARDINAL span in BBN. The nested entity structure allows us to capture that BBN layer, but also larger entities. Many other entities missing in BBN are due to differences in what should be included as an entity, especially in NUMEX and TIMEX types (e.g. the inclusion of one or several as cardinals).

Count	Category	Example
1977	RATE	57 cents a share
1272	CARDINAL	one, several, few
648	PER	Senate Majority Leader George Mitchell
533	UNIT	dollar, pound
530	INDEX	Dow Jones Industrial Average
431	DATE	yesterday, last night

Table 4.14: Most frequent types of entities which are in NNE but do not exist in BBN.

4.5 Summary

This chapter has outlined the creation of the NNE corpus, including annotation, consistency checking and analysis. The results of this chapter demonstrate that the consistent annotation of named entity structure has been achieved. The quality of our annotations has been demonstrated by measuring inter-annotator agreement in various ways, and by methodically conducting post-processing consistency checks.

The principled approach we applied while creating this corpus enabled the construction of a large-scale, highly consistent nested named entity corpus. The annotation process used, combining an innovative annotation tool, very detailed annotation guidelines, frequent meetings between annotators, and an extensive analysis and fixing of annotation errors has resulted in a high-quality corpus, with high inter-annotator agreement. This corpus enables all of the experiments in the following chapters, and we are now ready to begin experimenting with the Penn Treebank.

5 Merging Nested Named Entities into the Penn Treebank

In the previous chapter, we described the creation of the NNE corpus, a useful resource for investigating the structure of named entities. The value of this resource, however, is not just limited to the NER domain. The corpus we annotated is the same as used in the PTB, and by merging our annotations with the PTB annotations, we can combine both semantic and syntactic annotations. Further, having an aligned corpus with both annotations allows the NNE corpus to complement and be projected onto other PTB resources, such as PropBank (Kingsbury and Palmer, 2002; Palmer et al., 2005) and NomBank (Meyers et al., 2004), and also PTB-derived corpora for other formalisms, such as Combinatory Categorial Grammar (Hockenmaier, 2003) and LFG (Cahill et al., 2002). Thus, merging our NNE corpus and PTB annotations is an important next step.

The newswire for the NNE corpus we developed in the previous chapters is the Wall Street Journal portion of the Penn Treebank. While we use the same tokenisation and disambiguate difficult cases using the Treebank, the annotation structures are not necessarily compatible with the PTB's syntactic structures.

In order to use our NNE corpus for full syntactic and named entity parsing experiments, and to enable the annotation information to build on other PTB

aligned resources, such as NomBank and PropBank, we must *merge* or align the two sets of annotations, mapping NNE structure onto syntactic structure.

5.1 Merging process overview

The scope of this merging process is to align our nested named entity structures with syntactic structures, mapping NNE labels onto syntactic nodes. The desired output is a PTB style tree, enhanced with additional NNE node labels.

We follow the following process for applying labels to nodes in the PTB derivations. The process begins at the leaf (i.e. token) layer of the NNE, and works its way up breadth first.

1) If a node, p, exists that covers the required tokens and no others:

   (i) If p does not already have an NNE label, add the required NNE label to p. (See Figure 5.1.)
   For all nodes p_i where p_i is a unary parent or ancestor of node p, repeat (i).

   (ii) Else, add a new node, n, as a parent of p, and label n with the required NNE label.

2) Else if a node, p, exists which has direct children including all the required tokens and additional tokens, create a new node, n, as a child of p, and attach p's children such that n covers only the required tokens. Add the required NNE label to n. (See Figure 5.3.)

3) Else, apply rules for tree restructuring as described in Section 5.3.

5.2 Straightforward Cases

The majority of entity annotation spans are compatible with syntactic constituents, and the process of combining the two labels is straightforward.

5.2.1 Span match

The most common case is where an existing node in the syntactic tree fits the required tokens in an annotation span perfectly.

Consider the entities [Pierre]$_{\text{FIRST}}$, [Vinken]$_{\text{NAME}}$ and the larger span [[Pierre]$_{\text{FIRST}}$ [Vinken]$_{\text{NAME}}$]$_{\text{PER}}$. Each of these entity spans will be mapped to a node in the syntactic tree from PTB, as shown in Figures 5.1 and 5.2.

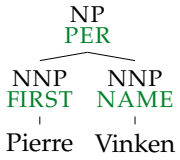


Figure 5.1: An example of a perfect match between syntactic and named entity structure.

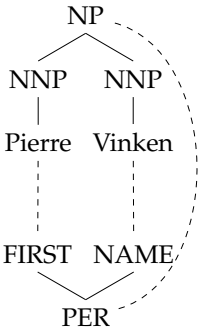


Figure 5.2: Phrase demonstrating mapping of NNE and PTB syntactic information

5.2.2 Span mismatch

If a node does not exist that spans only the required tokens, we attempt to add one. Consider the phrase the [[European]$_{\text{NORP:OTHER}}$ Common Market]$_{\text{ORG:OTHER}}$ approach with entity annotations NORP:OTHER and ORG:OTHER.

To merge this with the syntactic tree, shown in Figure 5.3 we work from the innermost entity annotation: [European]$_{\text{NORP:OTHER}}$. As seen in Figure 5.3, this entity span can be applied directly to a leaf node of the tree.

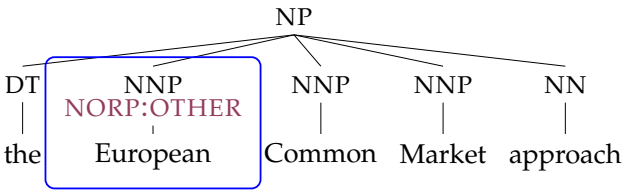


Figure 5.3: Phrase demonstrating addition of NORP:OTHER onto an existing token node.

Consider first the label over the token European, which can be added into the existing NNP node, following the steps outlined earlier.

The next entity span to add is ORG:OTHER, which should cover the tokens European Common Market. However, no node exists in the tree that covers only those tokens. We can insert a node covering the required tokens, labelling it an NML[1] syntactic node, with NNE label ORG:OTHER, as shown in Figure 5.4.

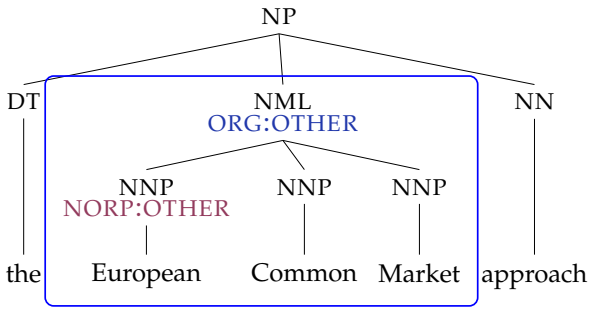


Figure 5.4: Phrase demonstrating addition of a NML node with NNE label ORG:OTHER within a subtree.

5.3 Special Cases

In other cases, inserting a node to cover only the required tokens is more problematic, as the tokens required may be in separate branches of a tree. The majority of these cases stem from discordance between our and the original PTB analyses of various structures, especially prepositional phrases. Another common source of discrepancy stems from inconsistencies in the analysis of QP phrases in the original PTB annotations or mismatches with Vadas and Curran (2007)'s added Noun Phrase Structure.

For some types of mismatches, we grow our labels to span the additional tokens required, for instance, the inclusion of certain determiners (see Section 5.3.2.1)

We categorise a number of different types of changes. Some are expansions to our NNE annotation spans to incorporate additional tokens which belong, syntactically, within structures; others are corrections to inconsistently annot-

[1]See Section 2.2.1 for a discussion of NML nodes.

ated structures already present in the PTB (such as QPs including QUALs); and others are changes to the analysis of certain structures, especially PPs, in order to create constituent spans for NNE spans.

5.3.1 Making changes to PTB syntax and NNE bounds

When our annotations and the PTB labels do not match, we must modify one or both of the corpora. To decide which corpus to modify, we must consider to what extent we should remodel PTB analyses to better capture the NE boundaries, or to what extent we should break our NE boundaries to maintain consistency with the PTB.

Our goal is to make as few modifications to each as possible, but for each modification to be linguistically sound. Where these two concerns conflict, we opt towards minimal change to the PTB corpus, since it is a standard dataset and the annotations in each in both corpora are decisions made by humans, and thus should at least implicitly reflect linguistic principles.

To a certain extent, more practical concerns short-circuit our linguistic concerns. We are constrained by the limitation of not making a large number of manual changes, and so seek to make as many linguistically motivated programmatic modifications as possible.

A further consideration is to ensure that any syntactic changes we make with respect to NEs do not contradict the grammar used outside of them, since making structural changes in only a subset of environments would lead to an inconsistent corpus. Nevertheless, we do in some cases make changes to the PTB structure only within the bounds of an entity.

We can consider NEs as noun phrases which take arbitrary grammatical structure internally. For some cases, such as people's names, noun phrases have their own idiosyncratic internal grammar. From this perspective, limiting restructuring changes to only occur within entities is a sound principle.

However, for other entities such as works of art or the names of bands, the structure more closely represents general English grammar, and does not conform to a specific, substantially different grammar. Because of this, one could consider that if we modify the PP structure inside entities, we should also make that same PP attachment change to all sentences in the PTB.

We elect to only modify structures inside named entity boundaries, rather than making changes to other parts of the Penn Treebank.

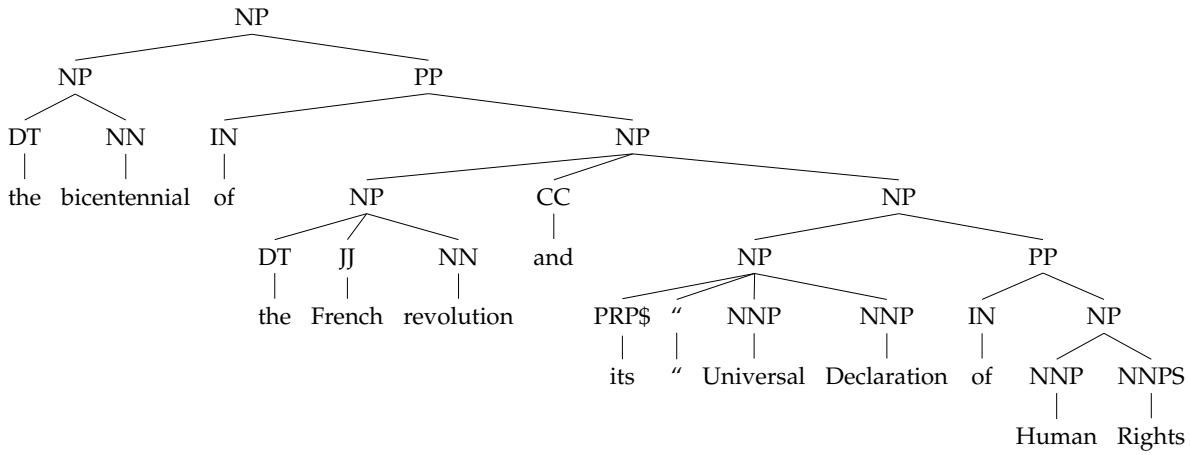


Figure 5.5: Original PTB analysis for sentence WSJ0723_21.

Take, for example, the Universal Declaration of Human Rights, the PTB derivation for which is seen in Figure 5.5. From a linguistic, named entity perspective, the span Universal Declaration of Human Rights should form a single constituent. However, due to the PRP$ attaching as part of the NP, and the PP attaching higher up to the NP, this analysis is impossible without restructuring the tree.

One possible solution to this problem is to adopt the inclusion of an additional layer to which determiners attach, as we see in some other grammar formalisms such as LFG. That is, the addition of an N layer to which determiners attach in order to make an NP would simplify PRP$ issues, and allow for PPs to be inside that N layer, unless they have explicit scope over the quantified noun phrase. While linguistically sound, adding these N layer or *nbar* nodes throughout the Penn Treebank would be a substantial change, and introducing

them only in cases with Named Entities would cause inconsistency in analysis of syntactic structures.

Another possibility for cases such as these would be to split the entity into two separate halves, and annotate each half as an entity. For instance, Universal Declaration be one marked as entity, and Human Rights be another. This doesn't capture the whole entity, but it does go some way to capturing it.

The solution which we elect to take is to modify the bounds of the NNE span to include the PRP$, growing the WOA span to cover [its " Universal Declaration of Human Rights]$_{\text{WOA}}$.

5.3.2 Include / Exclude rules for NNE spans

We use three rules to determine whether an NNE span should be changed, either expanded or constricted, in order to be compatible with the PTB constituent analysis.

5.3.2.1 Include DT in NNE span

In certain structures of NNE spans, the DT occurs within an NP which forms part of the NNE. In these cases, where the only difference between the target span and the found span is a DT, we grow the NNE span to include the DT.

This occurs frequently in NNE spans which include a PP, as in Figure 5.6 and Figure 5.7, end with a JJ or RB, as in Figure 5.8, or include a POS, as in Figure 5.9.

Occasionally, the DT span is grown in order to accommodate idiomatic expressions, such as the inclusion of the determiner when referring to a reverend, as seen in Figure 5.10.

5.3.2.1.1 Why not include the determiner in all NNE spans? Ideally, we would like to apply this more generally. However, we find determiners are not

The [National Association of Diaper Services]$_{\text{ORG:OTHER}}$, [Philadelphia]$_{\text{CITY}}$, says that . . .

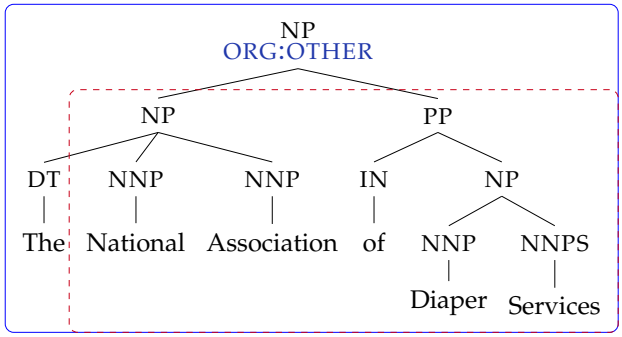


Figure 5.6: Phrase demonstrating the inclusion of DT the in `WSJ0120_43`, forced by PP attachment.

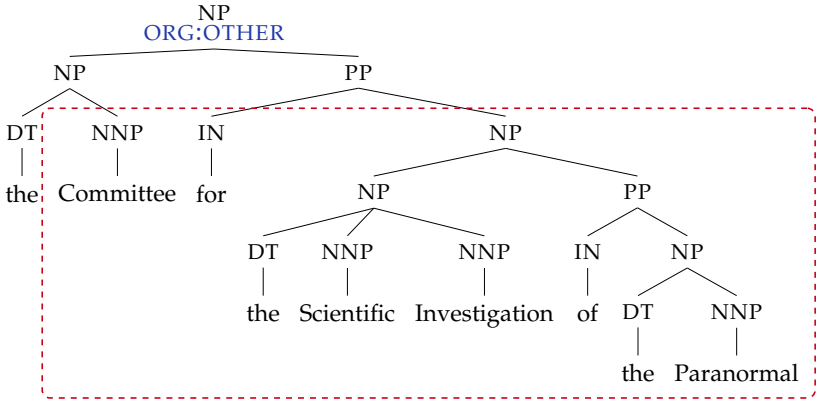


Figure 5.7: Phrase demonstrating DT inclusion in `WSJ0413_9`.

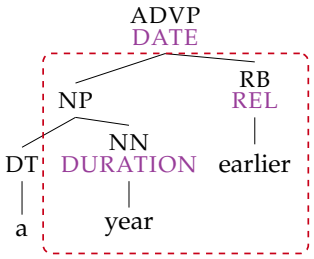


Figure 5.8: Phrase demonstrating DT inclusion in `WSJ0232_0`.

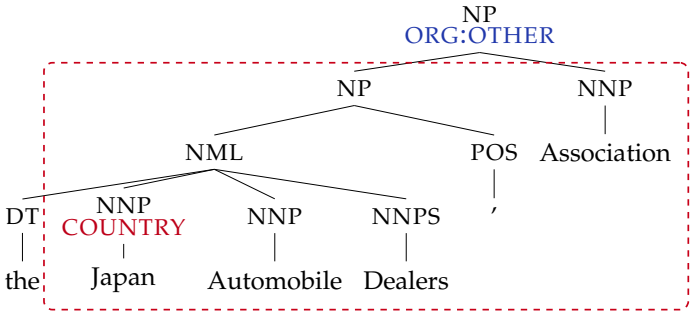


Figure 5.9: Phrase demonstrating DT inclusion in `WSJ0016_0`.

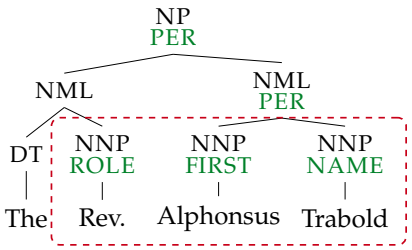
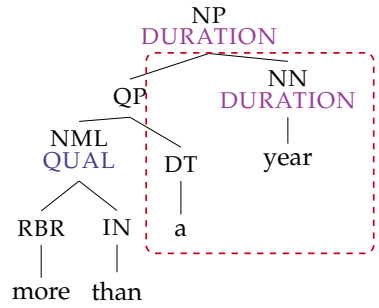


Figure 5.10: Phrase demonstrating DT inclusion in `WSJ0413_56`.

(NP (QP more than a) (NN year))

(a) Original PTB and NNE annotations



(b) Incompatible DURATION span when DT included

Figure 5.11: Phrase demonstrating problems associated with including every DT in spans, from `WSJ1374_10`.

always included into NNE spans, as this would cause spanning issues in other cases. Take for example, the common QP bracketing in Figure 5.11, and the NNE span: [[more than]$_{\text{QUAL}}$ a [year]$_{\text{DURATION}}$]$_{\text{DURATION}}$. The DURATION span over year is problematic in this instance. The PTB bracketing would not allow us to include the preceding DT into a larger NNE span: [a year]$_{\text{DURATION}}$.

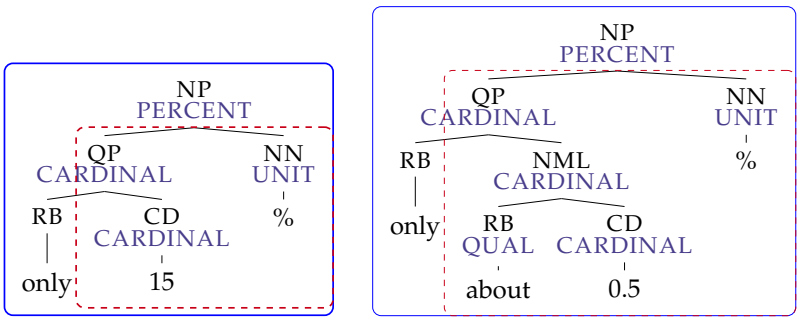
Adding preceding determiners in all cases would also create semantically incorrect spans. For example, while researchers from [the National Cancer Institute]$_{\text{EDU}}$ is valid, [the Dutch]$_{\text{NATIONALITY}}$ publishing group, director of [this British]$_{\text{NATIONALITY}}$ industrial conglomerate, smokers of [the Kent]$_{\text{NAME}}$ cigarettes and workers at [the West Groton]$_{\text{CITY}}$, Mass. , paper factory are all not correct spans. Similarly, although the larger span in TIMEX mention in [the late 1950s]$_{\text{DURATION}}$ is a semantically valid DURATION, in more complex constructions such as in [[the late 1950s]$_{\text{DURATION}}$ and 1960s]$_{\text{DURATION}}$, the inclusion of the determiner is more problematic.

We therefore elect to only grow the NE span when necessary due to the syntactic structure of an entity.

5.3.2.2 QUAL *like* tokens inside required phrase

Following our annotation guidelines, we only mark up adjectival or adverbial tokens as QUAL if they change the meaning of the CARDINAL to which they attach. That is, following the principle that QUALs attach as close to the CARDINAL as possible, we correctly add in spans such as [[[about]$_{\text{QUAL}}$ 0.5]$_{\text{CD}}$ %]$_{\text{PERCENT}}$, but have spanning issues with spans such as only [[0.5]$_{\text{CD}}$ %]$_{\text{PERCENT}}$ since the % attaches more closely than only in our NNE annotations, but not in the PTB annotations.

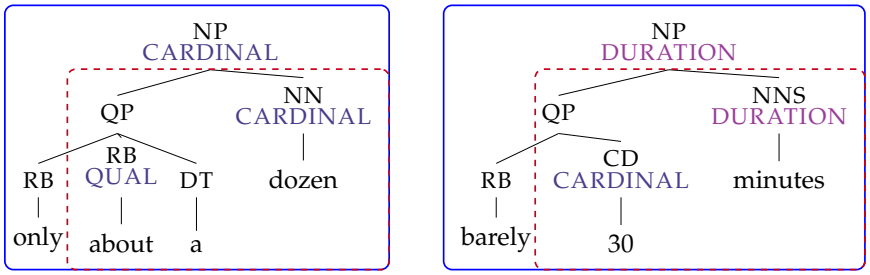
We allow our NNE spans to include these QUAL-like words (that is, words which act in the same way as QUALs do, but do not affect the cardinality of larger QP phrases or CDs) but do not label them explicitly. Tokens include: only, a mere, just, fully, somewhere, anywhere, further, possibly, perhaps, an additional, maybe another. This distinction of whether qualification is meaningful is then reflected in whether the token is assigned an NNE label.



(a) WSJ0095_3 **(b)** WSJ1528_15

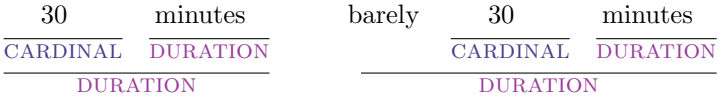
Figure 5.12: Phrase demonstrating expansion of PERCENT label to incorporate QUAL-like word only.

Various adjectives and adverbs which do not affect the cardinality of a phrase are nevertheless attached in the PTB analysis to the CARDINAL, or to an associated QUAL. We agree with these derivations, and allow these tokens to be included in our NNE spans, but do not explicitly mark them up.



(a) Growth of CARDINAL **(b)** Growth of DURATION

Figure 5.13: Adjectives and adverbs are allowed to grow NNE spans, but are not explicitely marked up.



(a) Original NNE annotation **(b)** Expanded DURATION span

Figure 5.14: Resulting named entity span incorporating adverbial barely in `WSJ0239_53`.

5.3.2.3 Include adjectives in NNE span

In addition to the QUAL-like tokens described above, other tokens also often occur within the required span. A frequent subset of adjectives, and occasional adverbs, to be included occur inside PER and ROLE spans. Examples are given below:

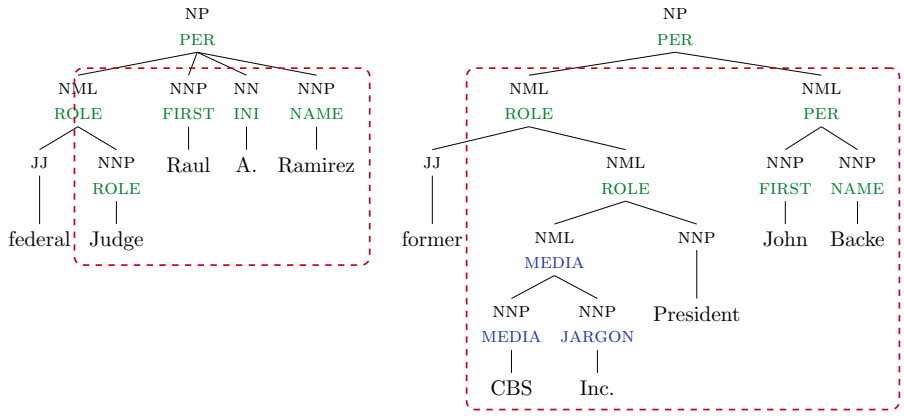


Figure 5.15: Phrases demonstrating the expansion of NNE PER spans to include additional tokens and spans.

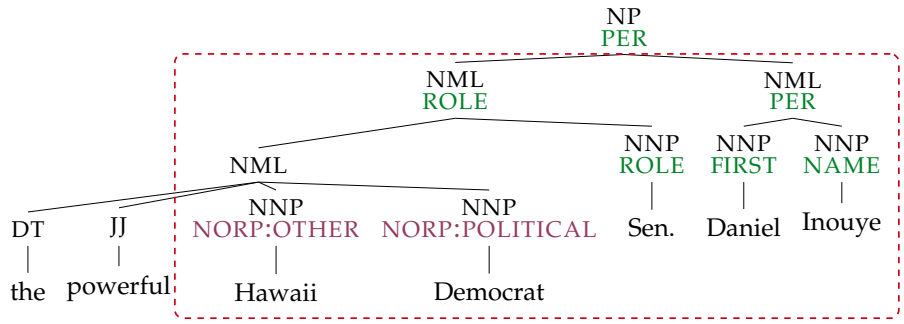


Figure 5.16: Phrase from `WSJ0101_14` demonstrating expansion of NNE PER to include NML

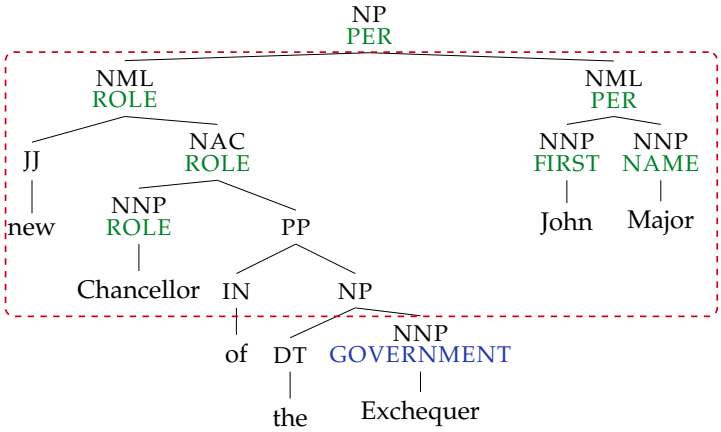
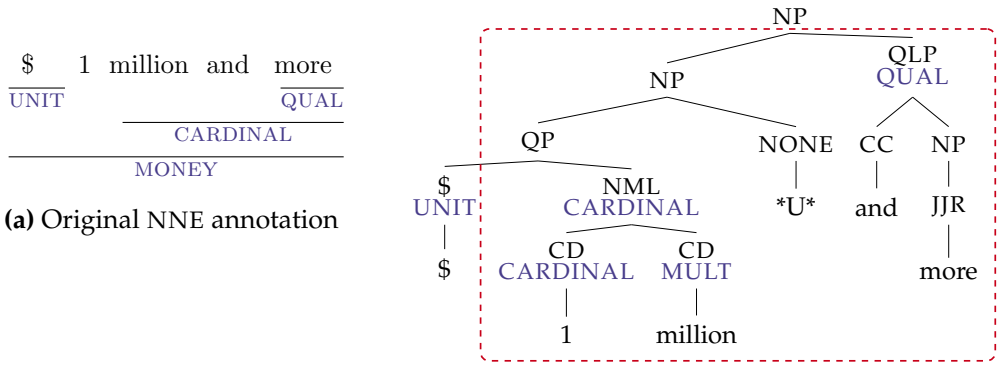


Figure 5.17: Phrase from `WSJ0231_41` demonstrating expansion of NNE PER to include NML

5.3.2.4 Include post-positional QUAL in NNE span

Following the annotation guidelines in attaching QUALs as close to the CARDINAL as possible, post-positional QUAL spans in our NNE annotations are

joined to the CARDINAL as early as possible. This, however, causes issues with the PTB analysis of MONEY structures, which binds the UNIT to the CARDINAL as early as possible. We follow the PTB analysis, and elect not to annotate the problematic larger CARDINAL span from our annotations.



(a) Original NNE annotation

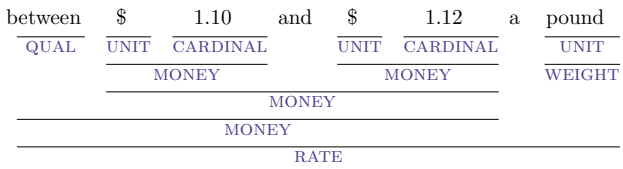
(b) Incompatible larger CARDINAL span

Figure 5.18: Phrase demonstrating post-positional QUAL in `WSJ0219_14`

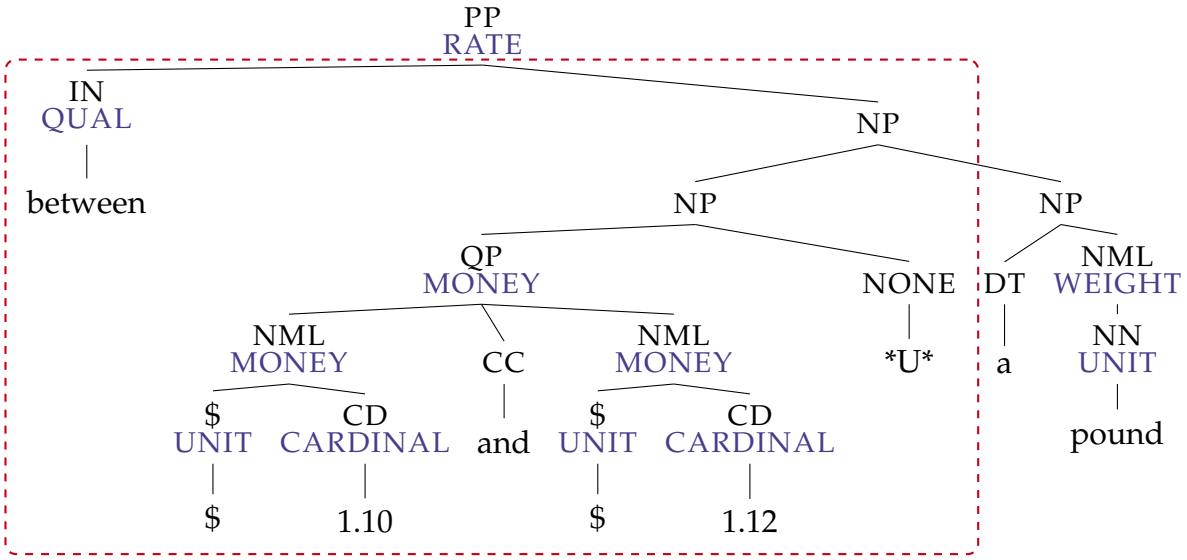
5.3.2.5 Between in RATE construction

The principle in our annotation guidelines of QUALs attaching as close to the CARDINAL as possible does not fit with the PTB analysis of phrases such as between $ 1.10 and $ 1.12 a pound. Specifically, our analysis of between as a QUAL which joins to CARDINAL (or MONEY spans) as closely as possible conflicts with the PTB analysis of such phrases.

We are unable to add in these larger NNE spans without substantially re-structuring these trees, and instead remove the spans.



(a) NNE analysis



(b) PTB derivation

Figure 5.19: Structured entity derivation and constituent tree for a RATE containing between, from `WSJ0664_51`. (a) NNE structure (b) Corresponding joint derivation with PTB analysis, showing lack of valid node for largest MONEY NNE span.

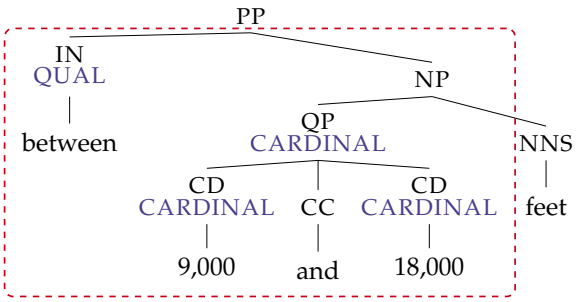


Figure 5.20: Phrase demonstrating no valid node for a larger CARDINAL span in `WSJ0550_11`.

5.3.2.6 Durations inside DATE constructions

Phrases of the form *ordinal cardinal duration of duration*, for example, first two weeks of June, have ambiguous bracketings possible.

```
( (first (two weeks) ) of June)
( (first ( two (weeks of June) ) )
```

We therefore elect to leave these structures as flat as possible. This check ensures that additional spans are not incorrectly added over either two weeks or weeks of June.

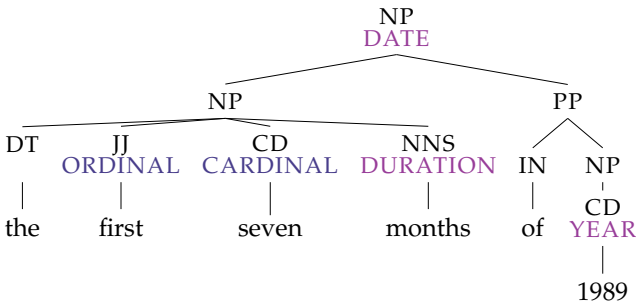


Figure 5.21: Phrase demonstrating correct flat structure of weeks of June style phrase in `WSJ0640_1`.

In our initial annotation, the smallest valid span capturing an NNE was annotated. Similar to expanding our NNE spans to incorporate determiners where required, we also include prepositions. These usually occur in TIMEX and NUMEX spans.

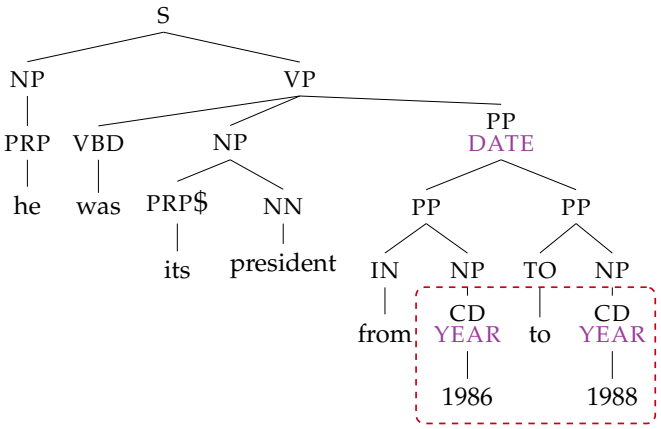


Figure 5.22: Phrase demonstrating preposition into DATE span in `WSJ0509_11`.

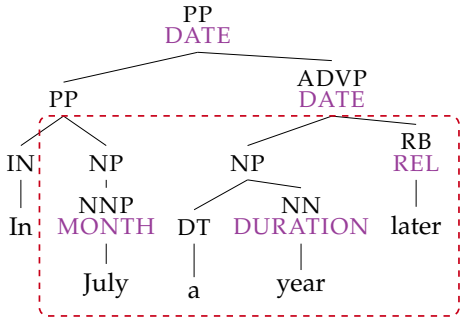


Figure 5.23: Phrase demonstrating inclusion of preposition into DATE span in `WSJ1634_98`.

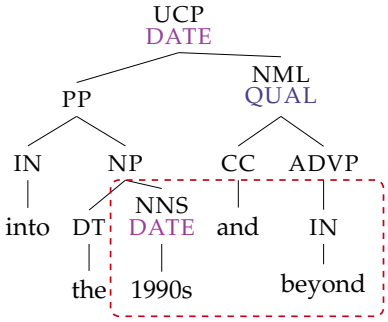


Figure 5.24: Phrase demonstrating inclusion of both determiner and preposition into DATE span in `WSJ1566_34`.

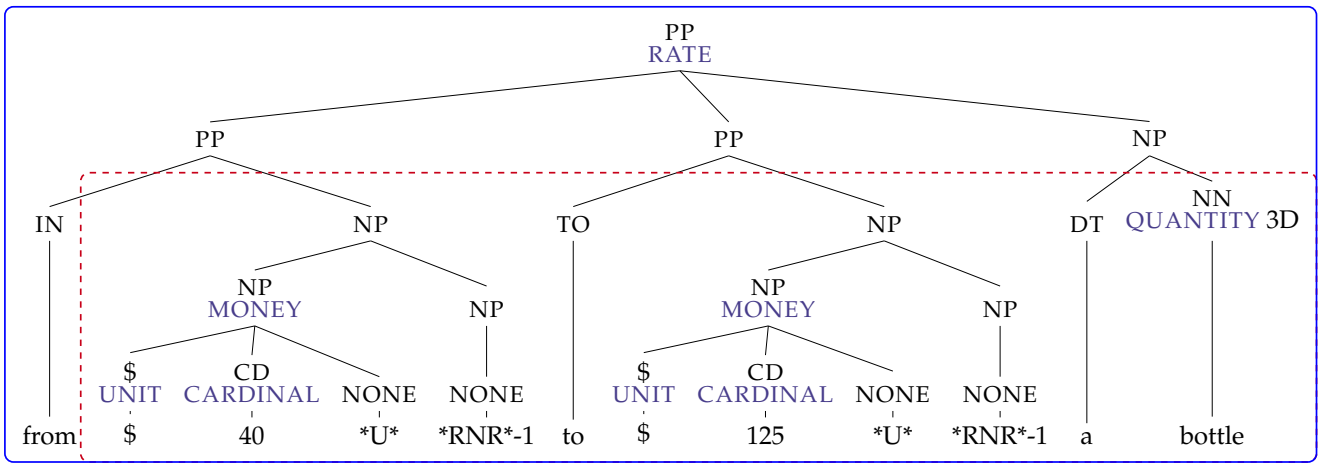
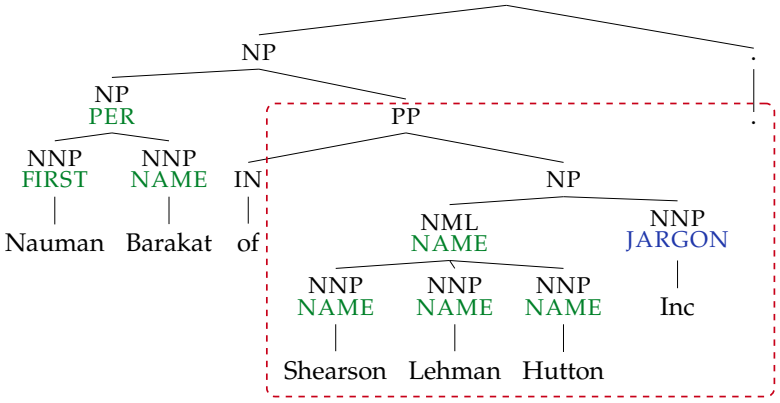


Figure 5.25: Phrase demonstrating addition of preposition from to RATE span in `WSJ0071_7`.

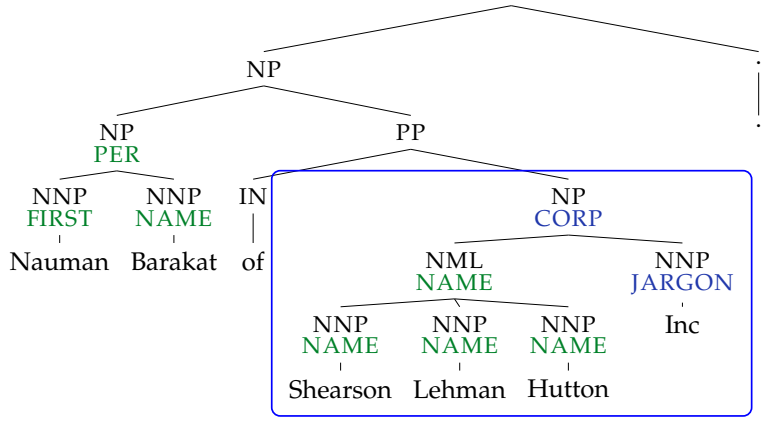
5.3.2.7 Exclude full stop in NNE span

In sentences where an entity ending in a full stop occurs at the end of a sentence which would also end in a full stop, only one full stop is used. This usually affects CORP annotations since JARGON terms such as Co. are usually mentioned

in abbreviated form. In some of these cases, our annotations inconsistently included the sentence final full stop as part of the entity. In these cases, we shrink the NNE span to exclude the sentence-final full stop, as shown in Figure 5.26, since including it in the NNE span would require substantial modifications to the syntactic tree, as sentence final full stops attach at the S level.



(a) Incorrect span including full stop



(b) Corrected span

Figure 5.26: Phrase demonstrating full stop annotation error in `WSJ1932_15`

Conversely, in two instances (`WSJ2007_22` and `WSJ2211_1`, seen in Figure 5.27), full stops that have been tokenised as separate tokens are nevertheless included in the bracketing of an entity. In these cases, in order to minimise changes to the PTB, we adjust the NNE spans to include these full stops.

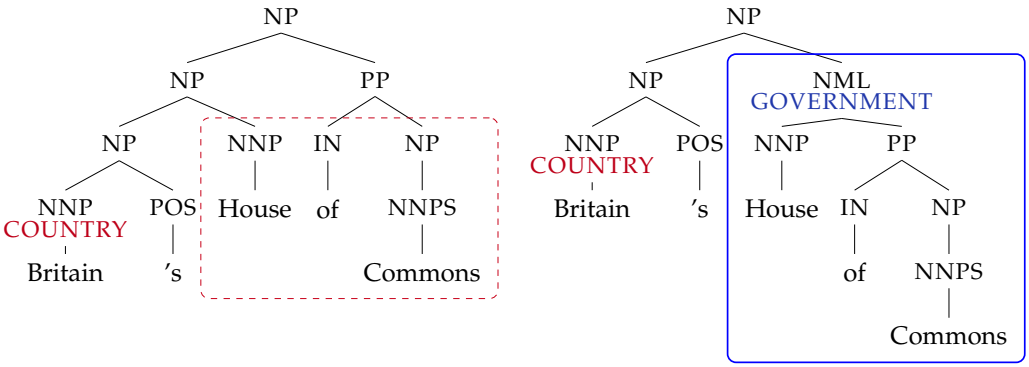


Figure 5.28: Phrase demonstrating PP attachment breaking NNP span in `WSJ0745_12` and required restructuring.

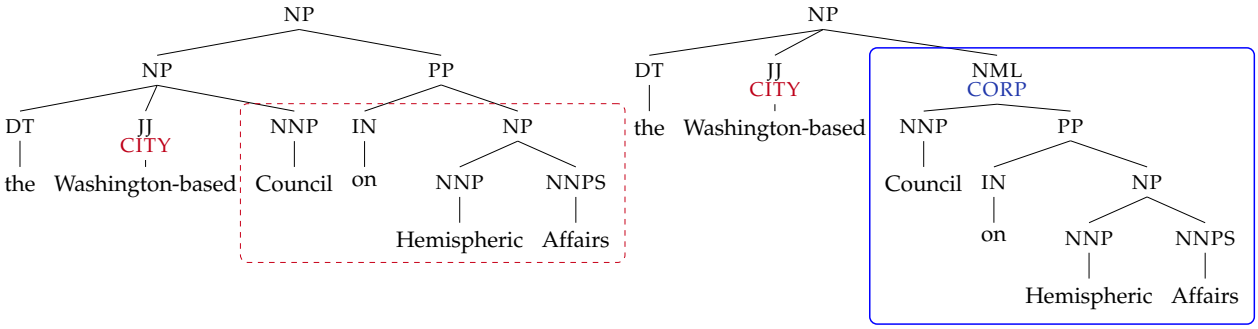


Figure 5.29: Phrase demonstrating PP attachment breaking NNP span in `WSJ0910_12` and required restructuring.

```
( (NP
    (NML (NNP J.P.) (NNP Morgan) )
    (CC &)
    (NML (NNP Co) (. .) )))
```

```
( (NP
    (NML (NNP Eli) (NNP Lilly) )
    (CC &)
    (NML (NNP Co) (. .) )))
```

Figure 5.27: Full stop PTB annotation inside NP bracket in `WSJ2007_22` and `WSJ2211_1`.

5.3.3 Tree Restructure Rules

5.3.3.1 Modified names with internal PPs

NNPs that include PPs and are preceded by JJs, POSs or other preceding terminals from the same parent node as the starting token(s) of the NNE span in the PTB are split across separate branches. We restructure these derivations, moving the PP and NNE-span initial NNP or NNPs into a new NML node, as shown in Figures 5.28, 5.29 and 5.30.

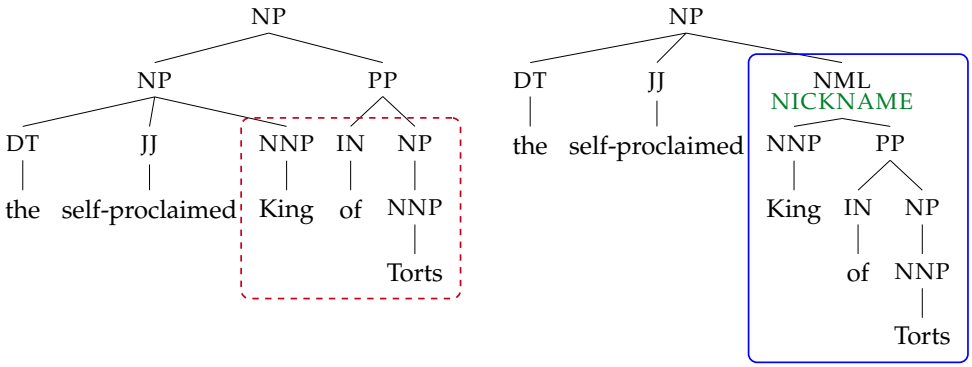
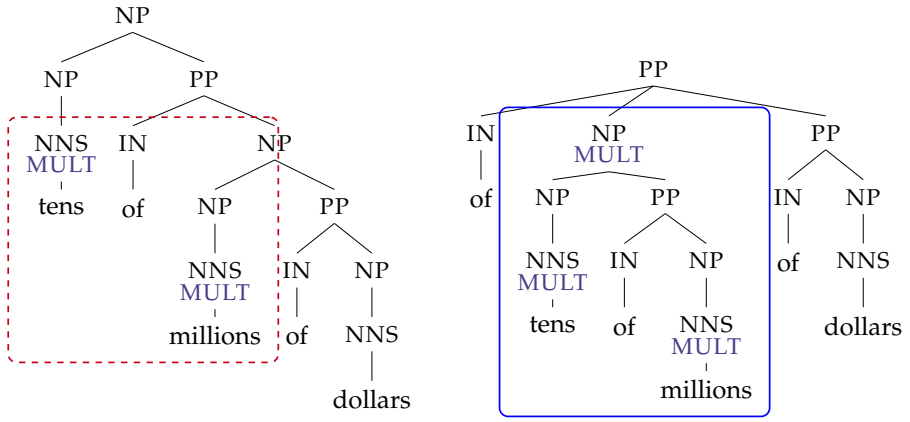


Figure 5.30: Phrase demonstrating restructuring PP attachment for NNP in `WSJ1688_1`.



(a) Original PTB bracketing **(b)** Restructured larger MULT span

Figure 5.31: Phrase demonstrating original bracketing in `WSJ0317_33` and restructuring to allow for a larger MULT span.

5.3.3.2 TIMEX/ NUMEX with internal PPs

We restructure trees with stacking MULT expressions such that they form one constituent. The larger MULT node now acts as one substitutable span.

5.3.3.3 As *x* as *y*

Phrases such as as much as and as early as are inconsistently annotated in the Treebank, sometimes occurring as part of a QP, and other times split into an NP and PP, structured in a fashion consistent with comparatives. Table 5.1 shows

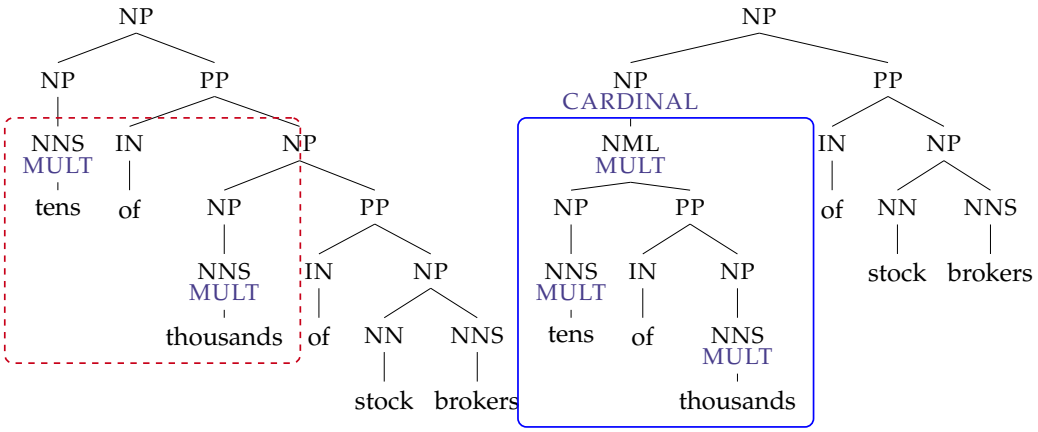


Figure 5.32: Phrase demonstrating MULT rebracketing in `WSJ0118_10`.

the comparative frequencies in the PTB for various bracketings with tokens as x as.

```
(NP (NP (QP as much as 15) %)
    (PP of
        (NP Jaguar shares)))
```

```
(NP (NP as much)
    (PP as
        (NP 15 %))
    (PP of
        (NP Jaguar shares)))
```

Indeed, the PTB guidelines describe this second analysis as an irregularity:

> There may be occasional irregularities in the treatment of as much as, where it appears with the bracketing shown below, which is consistent with the usual structure for comparatives but inconsistent with just about everything else. (Bies et al., 1995)

We restructure all phrases of the pattern as x as (e.g. as much as, as late as) that occur within TIMEX or NUMEX entities to align with the intended QP analysis, which allows for a QUAL span over the as x as tokens.

Structure	frequency
(QP as X as Y)	23
(? as X) (? as (NP Y))	64
(? as X) (? as (? Y))	168

Table 5.1: Table showing comparative frequency for as x as constructions.

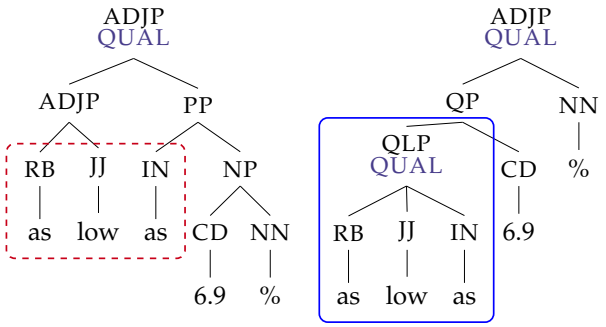


Figure 5.33: Phrase demonstrating as x as restructuring in `WSJ0451_15`.

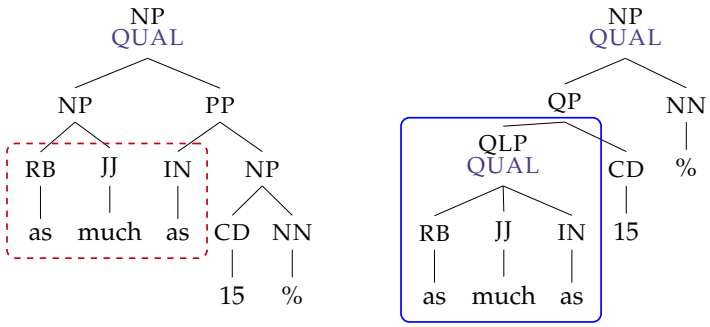


Figure 5.34: Phrase demonstrating as x as restructuring in `WSJ0688_0`.

This merge option requires us to add a new syntactic label, QLP, onto the node inserted for the QUAL NNE span. Additionally, we change ADVP with function -CLR (denoting *closely related*) to NP to label the phrase which is now correctly headed by a noun, and acting as an NP.

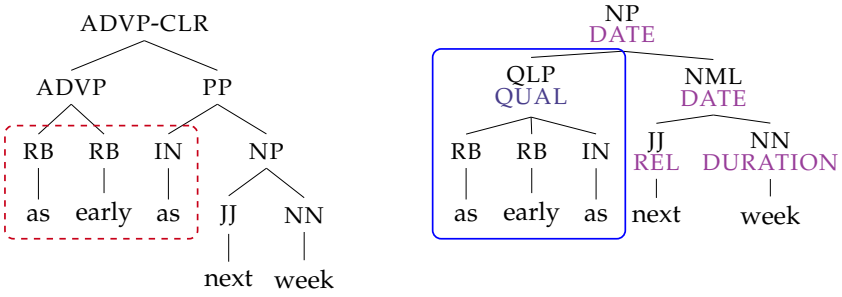


Figure 5.35: Phrase demonstrating as x as restructuring in `WSJ0142_55`

5.3.3.4 More than / less than

Similar to as x as, phrases such as more than and less than are analysed in an inconsistent fashion in the PTB. The guidelines give the following examples:

```
(NP (QP more than one)
```

```
      person)

(NP (QP more than three in five))

(NP (QP no more than 8)
      characters)
```

However, the following structure is frequently found, as in WSJ0203_15:

```
(NP
    (NP (JJR more) )
    (PP (IN than)
        (NP (DT a) (NN third) )))
```

We restructure all such cases which occur with phrases: bigger than, less than, order of, higher than, no more than, little more than, up to, much of, most of, greater than, more than, significantly lower than, longer than, slightly less than. This analysis is also much closer to the analysis of phrases like over 11 %, and brings it into line with other QUAL spans, with QUAL attaching as close to the CARDINAL as possible.

As with as much as spans, we use the new syntactic label, QLP, to span the node inserted for a QUAL NNE span label.

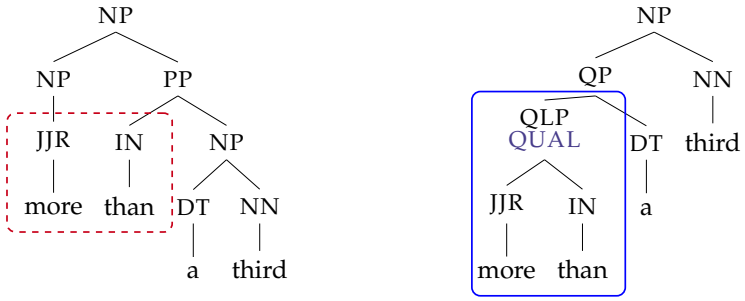


Figure 5.36: Phrase demonstrating more than restructuring in WSJ0203_15.

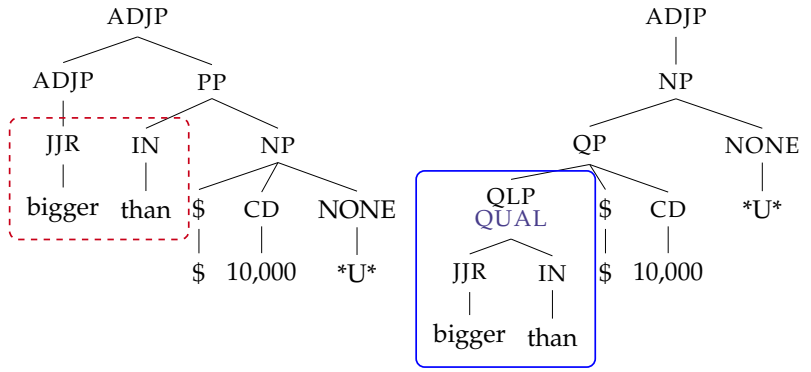


Figure 5.37: Phrase demonstrating more than restructuring in `WSJ0461_7`.

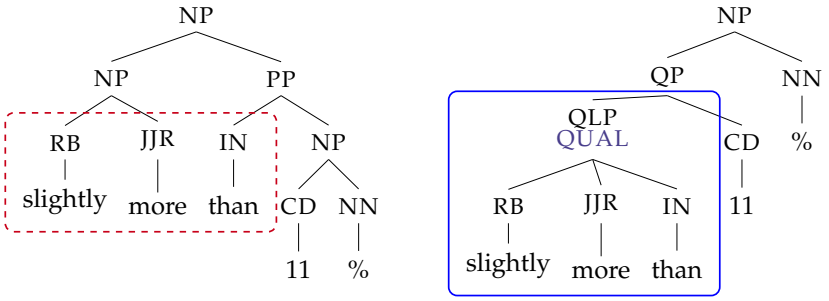


Figure 5.38: Phrase demonstrating more than restructuring in `WSJ0774_7`.

5.4 Remaining Cases

5.4.1 Manual fixes

In some cases, our NE analysis offers a linguistically motivated representation which more closely reflects the sentence's semantics than the analysis allowed by the restrictions of the PTB guidelines. In other cases, similar phrases had differing syntactic structures in different sentences. In applying our NE annotations, many of these sentences were identified, since they often caused node 'mismatches'. Structures were corrected and normalised as far as possible.

Consider the phrase 5 % to 10 %. Both structures in Figure 5.39 are found in the PTB.

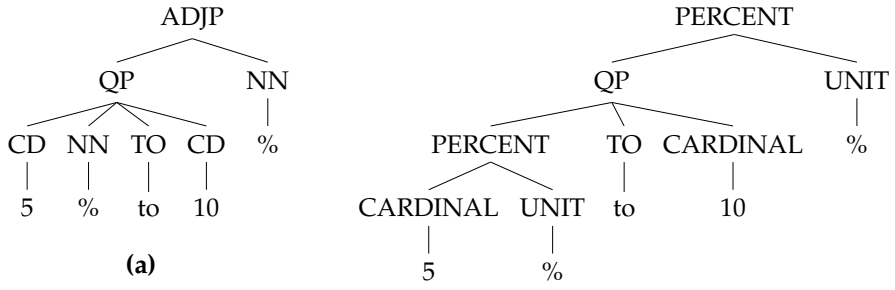


Figure 5.39: Suboptimal tree structures, (a) and (b), found in the PTB.

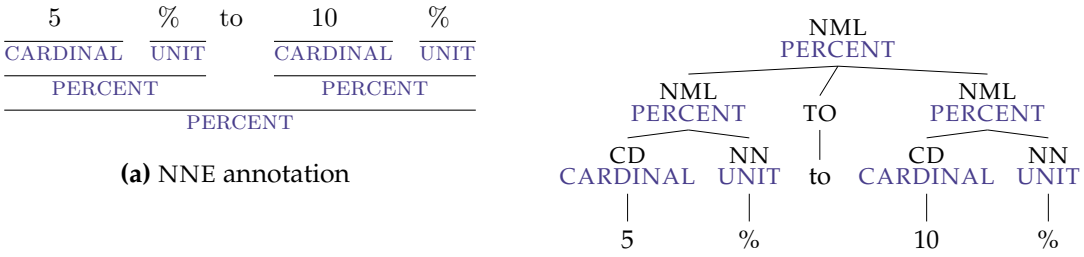


Figure 5.40: Proposed entity coordination in WSJ0666_28.

Our NE annotation includes two separate spans over each percentage, that are then coordinated to one larger span. This is inconsistent with some structures seen in the PTB for this phrase. We correct the syntactic derivation as shown in figure 5.40.

5.4.2 Cases where no merge is possible

In a small number of cases, we cannot accurately capture both NE semantic spans and syntactic spans correctly in the same tree. For example, we cannot get a clean span over the larger DATE span between 1983 and 1987 in Figure 5.41.

```
...
  (PP-TMP (IN between)
    (NP
      (NP (CD 1983) )
      (CC and)
      (NP
        (NP (CD 1987) )
        (, ,)
        (NP
          (NP (DT the) (JJ last) (NN year) )
          (SBAR
            (WHPP-1 (IN for)
              (WHNP (WDT which) ))
            (S
              (NP-SBJ (NNS figures) )
              (VP (VBP are)
                (ADJP-PRD (JJ available)
                  (PP (-NONE- *T*-1) ))))))))))
  (. .) ))
```

Figure 5.41: Fragment from sentence WSJ1556_22 demonstrating incompatible syntactic and semantic structure for DATE between 1983 and 1987.

In these cases, we annotate all smaller spans, (e.g. 1983 and 1987 both as DATE individually), and discard our larger, incompatible NE annotation.

5.4.2.1 Part of Speech tag and Label annotation errors

Some part of speech errors in the PTB cause anomalous syntactic structures. For example, the token No. (meaning number) is labelled with four different POS tags: NN 62 times, NNP 5 times, JJ twice and VB once. The particularly strange choice of VB, in sentence WSJ0678_21, forms the constituent structure: [South [Texas]$_{\text{STATE}}$ Project]$_{\text{CORP}}$ [Units [No. [[1]$_{\text{CARDINAL}}$ and [2]$_{\text{CARDINAL}}$]$_{\text{CARDINAL}}$]$_{\text{ORDINAL}}$]$_{\text{FACILITY}}$

```
        (NP
          (NP
            (NML (NNP South) (NNP Texas) )
```

```
    (NNP Project) (NNP Units) )
(ADJP (VB No.) (CD 1)
    (CC and) (CD 2) ))
```

When an incorrect POS is causing incorrect syntactic structures that interfere with our NE structure, we manually correct both the POS and syntactic bracketing.

5.5 Adding syntactic labels to additional nodes

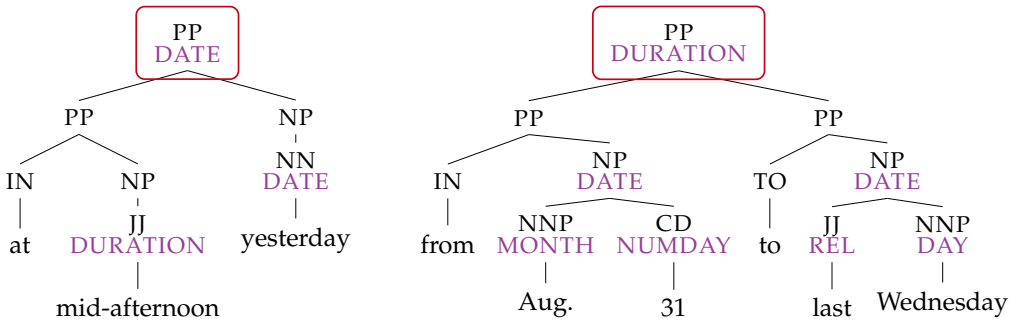
Once we have our combined tree, we need to add syntactic labels onto newly inserted nodes. If all of our NE labels applied directly to an existing node, no further action is required. If, however, our NE labels required the addition of a new node into the tree, for instance, in the case of a unary NE transformation, we need to add a syntactic label to that node.

We use the following rules for adding syntactic labels to nodes:

1) if NNE label is QUAL, add the syntactic label of QLP to the node.

2) else if first child is a PP, and the NNE label is DATE, add the syntactic label of PP. Examples of this can be seen in Figure 5.42

3) else, add a NML syntactic label to the node.



(a) PP added to DATE node **(b)** PP added to DURATION node

Figure 5.42: Examples of adding syntactic label of PP to DATE entities.

5.6 Discussion

In total, just over 700 constituency trees were modified as a result of the merging process, and the boundaries of 2276 entities were adjusted. A summary of the frequency of these adjustments is shown in Table 5.2.

In the process of combining the nested named entity spans and syntactic constituents of the PTB, though we have tried to minimise changes to both corpora, some changes are necessary. The changes outlined in this chapter represent a compromise between the integrity of the annotation spans and the constituency structure from the PTB.

An advantage of the merging process is that by checking for compatibility with constituency spans, a small number of annotation errors were identified and corrected. Thus, the consistency of annotations in the corpus, even in the version of the corpus not merged with the PTB, has been improved by syntactic validating performed using the PTB.

When considering the numbers of inconsistencies reported in this table, it is worth keeping in mind that in cases of genuine ambiguity in possible annotation spans, such as (first (two weeks) of June) or ((first two) weeks of June)), where we were unsure of the bracketing to use, we referred to the PTB and selected an analysis that was compatible with the syntactic bracketing.

5.6.1 Impact on NNE annotations

The most frequent change was a modification of the NNE boundary to include a determiner, accounting for two thirds of all changes. The majority of these included the determiner the, amounting to 1377 of the 2000 total determiner changes, followed by a further 688 cases involving a. The remaining issues involved tokens such as some, an, another, but these were comparatively infrequent. These determiner changes were most frequently needed for DATE (1324

Rule	Frequency
Modify NNE annotation bounds	
Include DT in NNE span	2096
Include quasi-QUAL in NNE span	321
Include adjective in NNE span	116
Post-positional QUAL	41
Exclude full stop in NNE span	22
Durations inside DATE constructions	12
Between in RATE construction	11
Tree Restructure Rules	
TIMEX/ NUMEX with internal PPs	113
Modified names with internal PPs	100
As *x* as *y*	48
More than / less than	31
Remaining manual decisions	76
Total count of span mis-matches	2995

Table 5.2: Table showing frequency of changes made to fit NNE annotations to PTB. Note that the sum of specific rule applications is more than the total number of nodes with mismatches, as multiple rules can apply on one span.

instances) entities, and ORGs (around 500, primarily of type GOVERNMENT, CORP and ORG:EDU).

While including determiners into the annotation span in only these cases does introduce inconsistency in their analysis as an NE, the other alternative of modifying the PTB analysis of determiners is a large task, and outside the scope of this work.

The other frequent change to NNE boundaries is the inclusion of *quasi-*QUALs in the NNE span. The most frequent of these are only (around half of all instances), and just (around a quarter of instances), with the most common entities affected being PERCENT, and MONEY. We expect the inclusion of these spans inside the NNE boundary to be less problematic than other changes, since the distinction on which adjectives should be included as a QUAL and those that shouldn't, namely if the adjective affects the numerical value or only offers a *journalistic comment*, was already quite subtle, and therefore likely to be problematic for parsers to learn already.

5.6.2 Impact on PTB annotations

Most of the changes we make to form a single, compatible corpus involve changes to the NNE annotations. In some cases, however, we believe the NNE annotation span to be a higher fidelity analysis, especially in the case that our NNE spans identify inconsistencies in PTB analyses. In these cases, we modify the constituency structure of the PTB. The majority of changes to the syntactic constituents concern prepositional phrases. These bulk of these changes are from the rules dealing with TIMEX and NUMEX with internal PPs, (often MULT issues) and modified named entities which include internal PPs, such as Britain 's House of Commons.

While we have endeavoured to keep the number of changes to PTB structures to a minimum, some were deemed necessary. The total changes to PTB structure due to rules and manual decisions amounts to around 300, compared to some 2500 changes.

5.6.3 Consistency in PTB after merging process

These changes made to the PTB and to NNE, while necessary in order to create a compatible corpus, introduce inconsistencies into the corpus.

Any modifications to the structures in the PTB will render results on parsing this new PTB version incompatible with results on the standard PTB. However, we found it necessary to ensure compatibility between the constituency structure and our named entity annotations, and to ensure the integrity of our NNE annotations..

The largest concern is the changes made to PPs in the PTB. Of specific concern is the fact that these changes affect only those PPs affected by our NNE spans; that is, only PPs in or containing NEs. Adding this inconsistency in the analysis of PPs is not ideal. The alternative would be a consistent reanalysis of

all PPs in the PTB, changing the level of attachment to the noun level, rather than the noun phrase level. While possible, this would be a huge process, and is outside the scope of this thesis.

Additionally, we believe that the internal constituents of these PPs are sufficiently distinct such that, if a parser is equipped with grandparent or other features that allow it to look inside one level of the children or the parent node, it should be able to learn to distinguish between these PPs and those not changed by our process. Features such as these are very common in state of the art parsers, and while we do acknowledge that these PP changes we introduce do constitute an inconsistency, it is an inconsistency we expect parsers to be able to predict reasonably well.

Further, we believe that the analysis we suggest is a higher fidelity analysis with respect to NE spans. The attachment of PPs at the N or NP level is the source of ongoing debate. This attachment level can be used to introduce an additional semantic distinction. For instance, Honnibal et al. (2010) modified a resource derived from the PTB, moving prepositional phrases and relative clauses to the N level by default.

Fundamentally, we believe that NEs should correspond to constituents in the phrase structure grammar. Not adjusting these constituent spans would result in an inconsistency in the NE scheme, which is the core contribution of this thesis. If, however, the original PP analysis would preferable for a specific task, it is a simple, mechanical transformation which can be reversed, and the original attachments recovered.

5.7 Summary

In this chapter we have described the key changes made both to the NNE spans and to the PTB constituents in order to form a single, compatible corpus of

syntax and NNE semantics. This process involved aligning the two corpora, identifying or creating nodes which shared both syntactic and NNE spans, or modifying the NNE span or PTB analysis such that a common node existed. The majority of changes made were to the boundaries of NNE annotations, with the largest group being that of including determiners (some 2000 of a total 3000 changes). While the introduction of these inconsistencies, both to NNE annotations and the PTB, were necessary, we are also conscious of the difficulties they will pose in further analyses. Specifically, we see the inclusion or exclusion of determiners in entity spans likely to be a problem case for NER systems, due in large part to the inconsistency introduced by this merging phase. The necessary modification to the PTB also poses issues for comparing results with those reported by others.

Nevertheless, ensuring the compatibility of annotation spans between the PTB and NNE spans is key both for the work done in the remainder of this thesis and for the wider use of the corpus in general. It will allow the NNE corpus to be used not just directly in PTB parsing, but also in all corpora derived from the PTB, such as different grammar formalisms (e.g. CCGbank) and semantic resources (e.g. NomBank and PropBank).

Now that we have completed this corpus, we are ready to train an existing near state of the art Penn Treebank parser to benchmark the difficulty of this combined syntactic and semantic corpus.

6 Parsing Nested Named Entities

In the previous chapter, we described the augmentation of the Penn Treebank with NNE structure. We will now use this extended corpus as the dataset for parsing experiments.

We would like to use our corpus as training data for a system to label NNE structures on unseen text. We anticipate that the best way to do this is to adapt an existing parser to encapsulate both standard syntactic structure and our newly constructed nested named entity structures. We expect that a constituency parser will be quite robust to learning various complex NNE sub-structures, mostly within noun and prepositional phrases.

To evaluate this system, we first need to determine the best way to represent NNE structures as node labels for the parser.

6.1 Parsing Background

High quality parsing has been achieved in a large number of languages, domains and formalisms. In English, the creation of the Penn Treebank corpus (Marcus et al., 1993) was instrumental in facilitating the development of high quality statistical parsing models in the newswire domain, reflected in the corpus being the de facto standard parsing corpus for English.

The parsing of named entities in English newswire text has not been the focus of research efforts, primarily due to the absence of a large corpus with

the requisite nested named entity annotations. The annotation of internal NP structure, absent in the original Penn Treebank, was added by Vadas and Curran (2007). This did not extend to the internal structure of named entities, however. As such, parsers trained on the Penn Treebank are not able to recover nested named entity structure.

6.1.1 Hand-written Grammars

Before the release of a large corpus of gold-standard constituency parses such as the Penn Treebank, parsers were constructed using hand-written grammars based on rules specified by grammarians.

> Parser development [was] generally viewed as a primarily linguistic enterprise. A grammarian examines sentences, skillfully extracts the linguistic generalizations evident in the data, and writes grammar rules which cover the language. The grammarian then evaluates the performance of the grammar, and upon analysis of the errors made by the grammar-based parser, carefully refines the rules, repeating this process, typically over a period of several years. (Jelinek et al., 1994)

Hand writing these grammars proved expensive and time-consuming, and the resulting grammars generalised poorly.

6.1.2 Penn Treebank Parsing

Parsing models (e.g. Magerman (1994), described below) trained on the Penn Treebank demonstrated the power of training with statistical models, albeit with some initial linguistic input, showing that a decision tree parser can significantly outperform a grammar-based parser developed by a grammarian over a ten-year period.

While the final Penn Treebank was still in development, Black et al. (1992) introduced the key innovation of head annotations. Namely, for each constituent, a specific subconstituent is deemed representative of the node. This *head* of each

phrasal node is determined by recursively selecting head subconstituents until a leaf node is reached. By using history-based modeling with the chain rule this head information can be percolated up through each individual probability decision.

Jelinek et al. (1994) and Magerman (1994) improved on the work of Black et al.'s (1992) work by removing the reliance on handwritten grammars. Magerman's model builds a candidate parse tree and uses breadth-first search to prune partial parses if their probability is less than the probability of the best found so far. The model works from the leaf nodes up, with leaf probabilities being used to calculate the probabilities of further potential parses. Parameters were estimated using decision trees with relative-frequency estimates at the leaves, with the probability of the final parse tree being the product of each of the probabilities assigned by the decision tree. Magerman's model achieved 78% accuracy rate (on sentences of 25 words or fewer) on Section 23 of the Penn Treebank, the section which has since become the standard test set.

Collins (1996) implemented a statistical generative model that estimated probabilities using relative frequency counts in the Penn Treebank. The model contains both rules for the probability of individual base NPs, and the probability of dependencies between constituents. The specific NP modelling rule is to account for base noun phrases, which are the most common constituent in the treebank, and allows the NPs to be represented by a single head when calculating further external probabilities. Constituents are generated by the model top-down, with the first inference producing the head constituent, and subsequent inferences generating the sibling constituents. The parse tree is built bottom-up, using the CKY chart parsing algorithm (Kasami, 1965), which has since been used in a number of other parsers.

Collins incorporates a lexicalised Probabilistic Context Free Grammar (PCFG) in a second generative model, which addresses data sparsity issues by making

independence assumptions. All modifiers are conditioned only on the head, not other additional modifiers, and the inclusion of modelling the complement/adjunct distinction and subcategorisation frames further improves the efficiency and accuracy of this model. In a third model, Collins further extends the parser to incorporate traces and *wh*-movement. Collins' best performing model achieves 88.6% precision and 88.1% recall on sentences in section 23 with fewer than 40 words.

Charniak (1997) proposed a probabilistic model that uses a chart to build candidate trees, using the probability of the head and the probability of the grammar rule being applied. Charniak (2000) builds on this with improvements to generating the lexical head's pre-terminal node before the head itself. At a similar time, Collins (2000) also improved on his previous result by using reranking information from a second model that included additional features. Both these models performed strongly, achieving slightly over 90% for precision and recall.

Building on work demonstrating that high quality PCFGs can be learned from a treebank by manual annotation (Klein and Manning, 2003) or automatic state splitting (Matsuzaki et al., 2005), Petrov et al. (2006) introduced a hierarchically split PCFG that could exceed the accuracy of lexicalised PCFGs. Starting with a simple Xbar grammar, Petrov et al. (2006) learn a new grammar whose nonterminals are sub-symbols of the original non-terminals.

More recent research efforts have refocused on reformulations of the parsing problem, such as dependency parsing (Yamada and Matsumoto, 2003) or specific issues, such as domain adaptation (Roark and Bacchiani, 2003; McClosky et al., 2006), or the parsing of NPs (Vadas and Curran, 2007), described further in Section 2.2.1.

6.1.3 Parser evaluation

The release of the Penn Treebank also heralded a substantial improvement in the evaluation of parsers. Treebanks allowed competing parsers to be trained and evaluated under identical conditions. Black et al. (1991) proposed the PARSEVAL metrics for evaluating consistency parsers, based on the number of constituents in a system's proposed parse that match the gold-standard parse tree in the Treebank. These measures are labelled precision, labelled recall and labelled F_1-score.

Evaluation of parser output is conducted by comparing the brackets produced by a parser, which delimit constituent boundaries, to those prescribed in the gold-standard bracket data. The most widely used evaluation measures for constituency parsers are the PARSEVAL metrics (Black et al., 1991; Grishman et al., 1992). For a sentence or subsentence to be correct, each set of brackets should begin and end at the same token as in the gold standard, and share the same label. Matched bracket evaluation can then determine the precision, recall and F_1-score.

Precision (P) shows what percentage of the predicted brackets are correct:

$$precision = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{6.1}$$

Recall (R) reflects what proportion of gold brackets were correctly identified:

$$recall = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{6.2}$$

The F_1-score is the harmonic mean of precision and recall:

$$F_1 = \frac{2PR}{P + R} \tag{6.3}$$

Parser development often involves an iterative training-evaluation cycle. Evaluation data is usually split between a *development* and *test* set. The development (dev) set is used in many cases for model optimisation and model selection, while the held-out test set is used for model assessment.

6.2 Variants of merging nested named entity Structure into the Penn Treebank

We find ourselves with a single tree containing the annotations from our two original corpora, one being the PTB with syntactic bracketing, and the other being the nested named entity annotations described in Chapter 4.

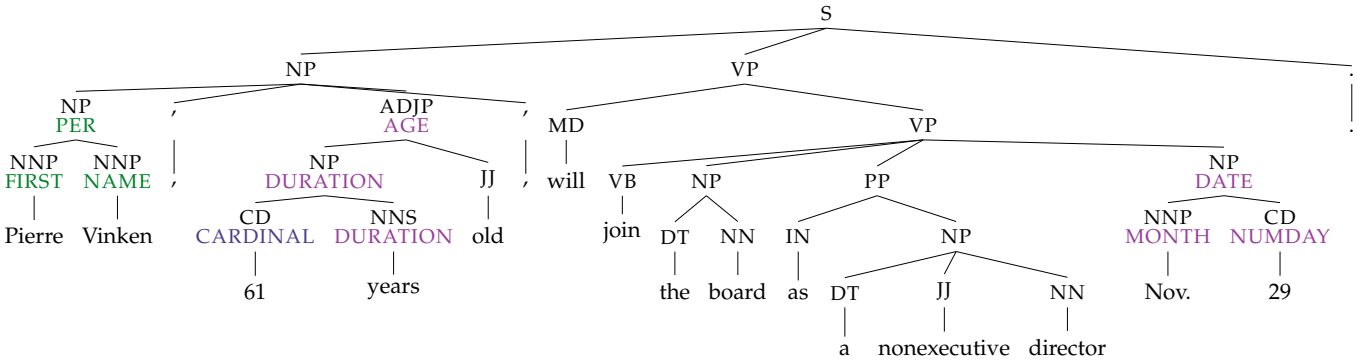


Figure 6.1: Underlying sentence `WSJ0001_0` with PTB and NNE annotations.

We present six alternative strategies which vary in the surface forms produced in combining the syntactic and NNE labels. These experiments will demonstrate both the difficulty of the combined parsing NER task, and to what extent a parser can learn a combined model. Further, the different variants of combined syntactic and NNE annotations will explore to what extent the existing syntactic structures already capture the NNE structure, or conversely, how easily the syntactic structures can be expanded to do so.

6.2.1 'Joint' variant: concatenated POS and NNE label

The simplest combination is achieved by concatenating the POS tags or syntactic node labels with the nested named entity structure. Thus, each node is annotated with a combination of each token's POS tag or syntactic label and the entity span that the token or node belongs to.

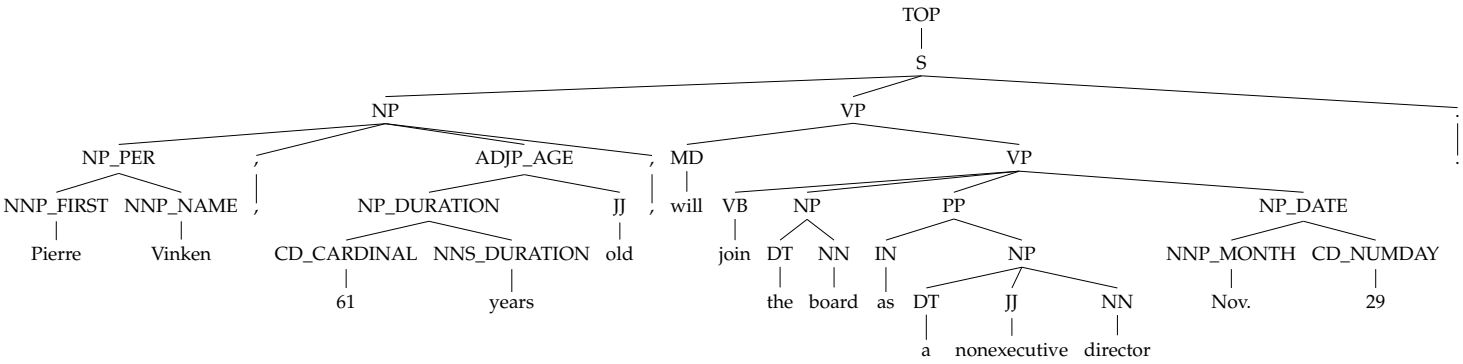


Figure 6.2: Sentence `WSJ0001_0` with JOINT variant annotations.

The JOINT variant has the benefit of being very straightforward to produce. NE labels fit semantically directly onto existing nodes in the tree, and their combined structure is directly preserved. However, due to the labels being produced by combining the existing syntactic labels with our new NE annotations, this variant has a substantially larger number of labels, more sparsely applied over the corpus.

The JOINT variant results in a total of 872 unique labels over sections 00, 02-21, and 23 of the WSJ. Furthermore, some labels occur in the development and testing sections of the corpus that do not occur in the training section. Specifically, two labels occurred in section 00 that did not occur in the training data: NN _QUANTITY:3D and JJ _WEAPON (see Figure 6.3), and a further 13 labels occurred in section 23 that did not occur in the training data. These cannot be correctly predicted, since they never occur in the training data.

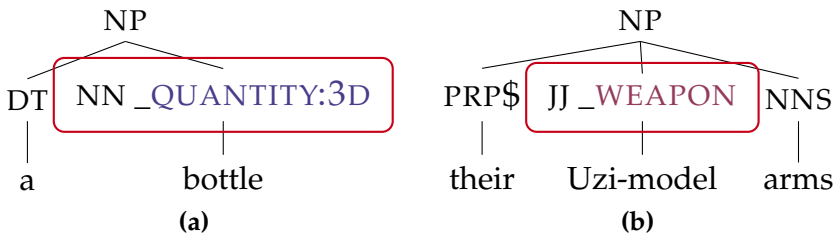


Figure 6.3: Two examples of labels occurring in section 00 that do not occur in the training data.

6.2.2 HIGH variant

The HIGH variant inserts additional nodes into the syntactic tree above the existing POS or syntactic node label. That is, NE labels are added above the corresponding syntactic label as a unary parent which covers the required tokens.

In this variant, all NE nodes occur as nodes with a single child of a syntactic label.

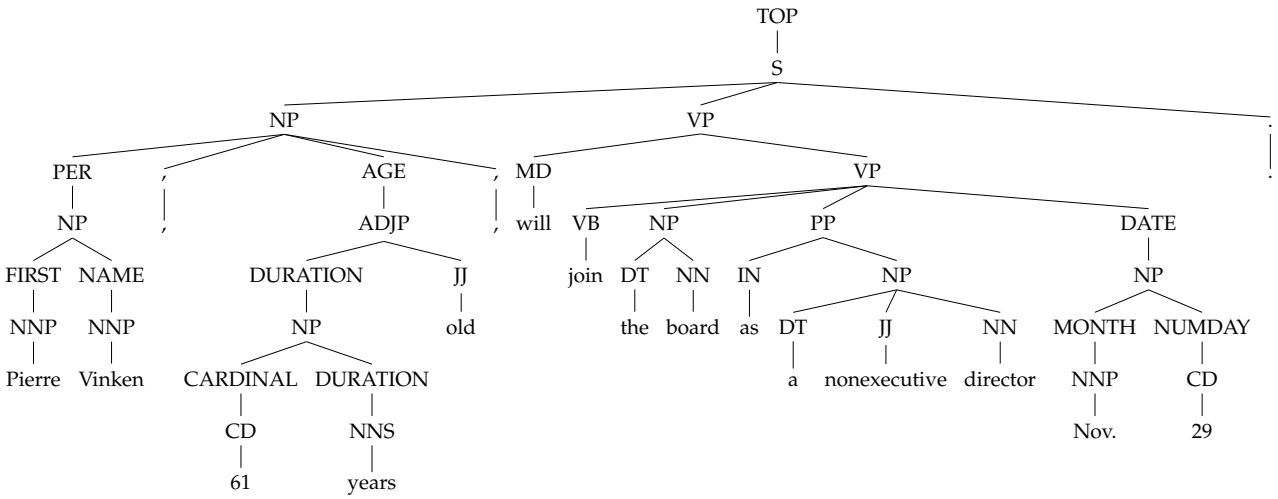


Figure 6.4: Sentence `WSJ0001_0` with HIGH variant annotations.

The HIGH variant resulted in a total of 211 unique labels used over the train/test/dev sections. Due to the reduced size of the tagset, all tags in test and dev occur in the training data. The smaller number of unique labels shifts much of the difficulty of the task from having very sparse data to learn from, to resulting in more complex nesting structure.

We expect the HIGH variant to suffer from difficulty in reproducing the NNE structure using a standard constituent parser due to the distance between the tokens and our first level of NE annotation. If the constituency parsers does not include grandparent or grandchild features, it would not be able to see tokens when making decisions on even the first layer of named entity structure.

6.2.3 LOW variant

The LOW variant adds nodes with NE labels into the tree below the corresponding nodes with syntactic labels. All NE nodes have a single parent, which is a syntactic label. Token level NE tags apply directly to the token, with the POS tags attaching to the direct parent node.

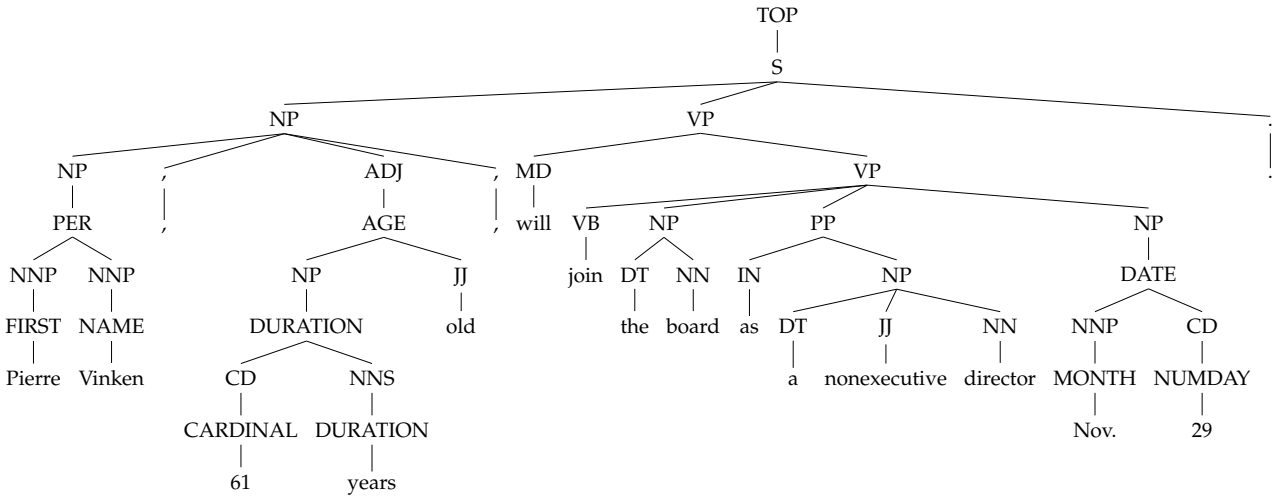


Figure 6.5: Sentence `WSJ0001_0` with LOW variant annotations.

As with the HIGH variant, the LOW variant resulted in a total of 211 unique labels used over the train/test/dev sections, with all tags in test and dev occurring in the training data.

6.2.4 POSLOW variant

The POSLOW variant is similar to the LOW variant, with the exception that POS tags are kept at token level. That is, POS tags apply directly to tokens in the tree, and all NEs occur either as a direct parent of a POS or as an only child of a syntactic label.

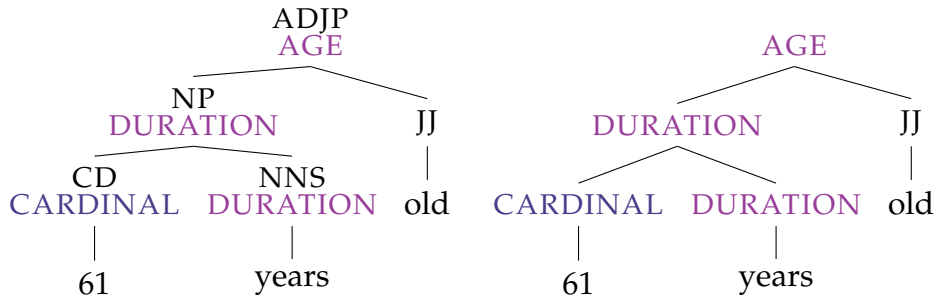


Figure 6.7: Phrase from sentence `WSJ0001_0` with combined PTB and NNE annotations, and corresponding phrase with SUB variant annotations.

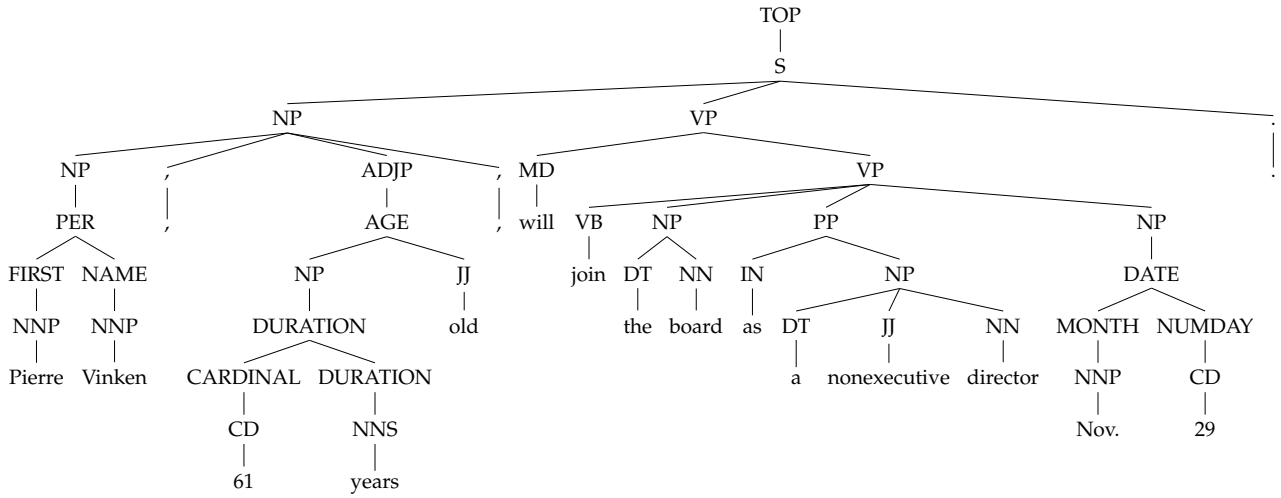


Figure 6.6: Sentence `WSJ0001_0` with POSLOW variant annotations.

As with the HIGH and LOW variants, the POSLOW variant resulted in a total of 211 unique labels used over the train/test/dev sections.

6.2.5 'Substitution' (SUB) variant

The SUB variant takes the highest node with an NE label, and for all nodes below it, uses the NE label if it exists. For example, consider the branch in figure 6.7. The root node of this example, with labels ADJP and AGE, has both NE and syntactic label. In the SUB variant, only the NE label (AGE) is used. Similarly, for its first child, which has syntactic label NP and NE label DURATION, we use DURATION to label the node. The other child of our example's root only has a syntactic label (JJ) so we use that label for that node.

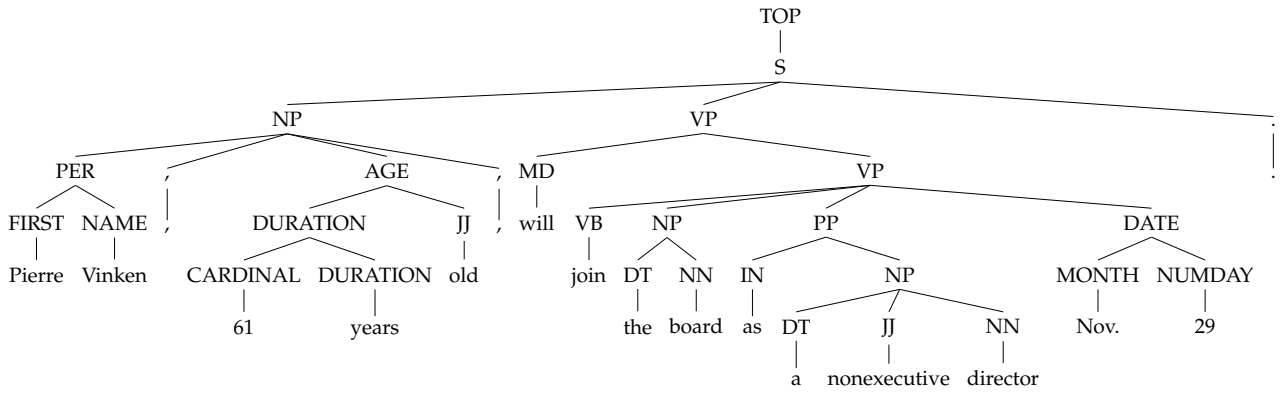


Figure 6.8: Sentence `WSJ0001_0` with SUB variant annotations.

The SUB variant resulted in a total of 209 unique labels used over the train/test/dev sections, with all tags in test and dev occurring in the training data.

The performance of the SUB variant will be of particular interest because it best encapsulates the linguistic principles behind the inclusion of nested NER into syntactic structure. Specifically, it is important to see how well our constituent parser can learn the large number of NP subcategorisations required to capture our NE types. The SUB variant is the most direct method of including more semantic information directly into the grammatical structures without creating additional layers of syntactic obfuscation. As such, we are particularly interested in the performance of models trained on this variant.

6.2.6 'Substitution' under parent label (SUB LAYER) variant

The SUB LAYER variant is similar to the SUB variant (6.2.5) where nodes which have both syntactic and NE annotations in the underlying combined tree are expressed as only the NE annotations. In this variant, however, the topmost node of each entity (or nested entity structure) has its syntactic label added as a direct parent. That is, the AGE node, being the top-most entity in that branch, has as parent its syntactic label, ADJP.

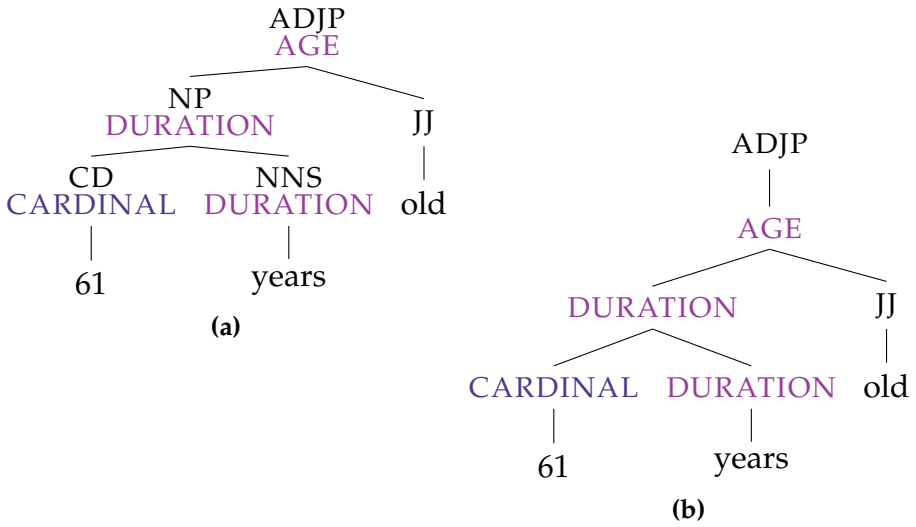


Figure 6.9: Phrase from sentence `WSJ0001_0` with PTB and NNE annotations, and corresponding phrase with SUB variant annotations.

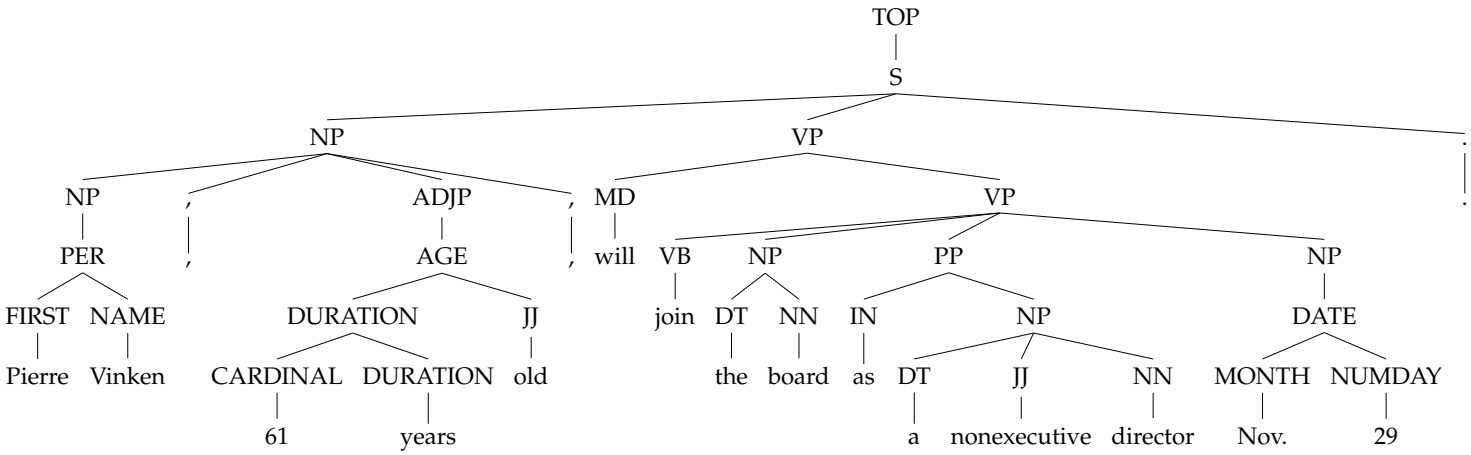


Figure 6.10: Sentence `WSJ0001_0` with SUB LAYER variant annotations.

The SUB LAYER variant resulted in a total of 210 unique labels used over the train/test/dev sections, with all tags in test and dev occurring in the training data.

Variant	No. labels
Pure PTB	62
JOINT	872
HIGH	211
LOW	211
POSLOW	211
SUB	209
SUB LAYER	210

Table 6.1: Number of unique labels in each variant.

6.3 Parsing syntactic and nested named entity structure with the Berkeley Parser

In selecting a parser, we elected to use a constituency parser that was widely used on the Penn Treebank. The Berkeley Parser (Petrov et al., 2006) satisfied these constraints as well as being likely to be robust to the large number of noun subcategories used in our variants to capture NNE structure.

The Berkeley parser performs split-merge cycles on our training data and automatically induces a PCFG with optimised syntactic categories. The split-and-merge approach allocates subsymbols adaptively where they are most effective. Expectation-maximasation (EM) is used to learn a set of rule probabilities on latent annotations that maximise the likelihood of the training trees. In the training phases, each label is split in two, trained, and then the loss in likelihood incurred when it is removed is measured. If this is only a small loss, the new annotation does not represent an important distinction carrying useful information, so that split label is removed. The split-merge cycle allows the parser to progressively increase the complexity of the grammar, prioritising the most useful category extensions.

It is interesting to note the similarities between some automatically learnt subsymbols, or subcategories, of POS tags, especially NNP, as described in Petrov et al. (2006). Specifically, when analysing the three most frequent word in each subcategory of the POS tags reported, many align closely with our proposed NE categories. For example, NNP-2 contains initials (INI) J., E., L.; NNP-12 contains first names (FIRST) John, Robert, James; NNP-13 contains honorifics and roles (HON, ROLE) Mr., Ms., President; NNP-9 contains months (MONTH) September, August, and days of the week (DAY) Friday; while NNP-14 contains abbreviated month names, Oct., Nov., Sept.. Other POS subcategories

also align closely to our own NE categories, including CD-11, which contains MULT tokens such as million, billion and trillion.

Other NNP subcategories have a less well delineated set of contents. NNP-5 contains tokens we would annotate as JARGON (Inc., Corp., Co.), but these overlap considerably with NNP-7 (Corp., Inc., Group). Our approach entails combining the semantic information and linguistic intuition into the grammar, and using the split-merge approach, with a view to mitigate some of the increased complexity of the task.

6.3.1 Learning combined syntax and NNE structure is difficult

As training data we used sections 02 to 21 from the Wall Street Journal (WSJ) of the Penn Treebank, augmented with our NNE structures.

Sentences		Bracketing			Tagging
Variant	Valid	P	R	F	Accuracy
Pure PTB	1916	90.86	90.36	90.61	96.89
syntax only	1918	86.00	89.47	87.70	96.80
HIGH	1915	85.08	84.71	84.89	96.72
LOW	1914	87.21	87.68	87.45	95.59
POSLOW	1918	84.59	84.95	84.77	96.67
JOINT	1918	83.52	87.16	85.30	94.93
SUB	1915	84.23	87.87	86.01	95.46
SUB LAYER	1918	85.38	87.47	86.41	95.73

Table 6.2: Eval-B Analysis of the 1920 sentences in section 00 for each of the variants. Shown are number of valid sentences parsed, Precision, Recall, F_1-score and Tagging Accuracy.

To gauge the difficulty of learning each of the variants described in this section, the Berkeley Parser (Petrov et al., 2006) was trained on each variant,

as well as the original Noun Phrase augmented version of the Penn Treebank (Vadas and Curran, 2007).

Since EVAL-B is calculated based on the number of correct constituency claims, the results shown in Table 6.2, are not directly comparable, as each variant results in a different number of brackets and labels. For the purpose of ensuring the task has not become substantially harder with the additional nested named entity nodes, we do compare performance, since it offers us assurance of the similar overall complexity of the task. We find that learning the more complex annotations does seem to be more difficult, but not substantially so.

Furthermore, since the number of unique labels is approximately steady between variants (other than JOINT), we cannot draw any conclusions about the impact of label sparsity on parser performance.

To judge the added complexity of the syntactic changes introduced into the corpus discussed in Section 5, a merged corpus was created (using the LOW variant), then stripped of all NE labels, thereby creating a corpus that only included syntactic labels. This too was trained and tested using the Berkeley Parser.

We expect to see a difference between the HIGH and LOW variants with respect to whether the syntactic structure or nested named entity structure is able to be learnt better. Notably, the absence of grandparent features in the Berkeley Parser means that the addition of intermediate nodes between the NE labels and tokens in the HIGH variant, and between the syntactic node labels and tokens in the LOW variant, would make NNE and syntax difficult to learn, respectively.

The LOW variant appears to be a good strategy for learning the combined NNE and syntactic structures. It offers a good balance between maintaining a small label set and ensuring that difficult NE label decisions are made with

direct token features visible. It is interesting to note the difference between the LOW and POSLOW variant performance, even though the numbers are not directly comparable. The performance discrepancy between LOW and POSLOW gives credence to the expectation that the token itself, rather than the POS, is vital in making correct NE decisions.

The tagging accuracy of JOINT is lower than the other variants, but performance is still comparable, which is of particular interest if a more straightforward implementation is desirable.

6.3.2 How well does a combined model learn syntax?

In order to test the effects of different variants on learning the syntactic structure only, and thereby obtain directly comparable numbers, we conducted the same experiments from Section 6.3.1 again, recalculated using only the syntactic or nested named entity labels. These experiments were run to test only the model's results on the syntactic or nested named entity component of our combined corpus.

In one experiment, all NE labels were removed from the resulting dev and test data, and results when only considering the remaining syntactic structure were calculated. These are seen in Table 6.3.

These 'syntax only' results were straightforward to obtain for the HIGH, LOW, POSLOW and JOINT variants, as the entity labels could simply be excluded. The SUB and SUB LAYER variants, however, had already substituted out some of these syntactic labels for the named entity ones. For each NE label in these variants, instead of removing that node, we use the most common syntactic label for that combination of token, NE label and parent's label, or purely NE as a backoff.

The process for reinserting these NE labels has quite good coverage over the tokens needed in the dev and test set, but there are some ambiguous or

Variant	Valid Sent.	P	R	F
pure ptb	1916	90.39	89.86	90.12
syntax only	1917	90.27	88.98	89.62
HIGH	1915	89.95	88.74	89.34
LOW	1914	90.63	89.29	89.96
POSLOW	1918	89.41	88.88	89.14
JOINT	1918	89.26	88.82	89.04
SUB	1915	89.40	88.89	89.14
SUB LAYER	1918	90.00	88.78	89.38

Table 6.3: Eval-B Analysis of the 1920 sentences in section 00 for each of the variants when evaluated only on syntactic components of their output.

unknown nodes for which we cannot recover a correct label. Between the dev and test sets, there were 160 unique tokens which were found to be ambiguous given their NE and parent context. Slightly more than half of these (82) were found to be only rarely ambiguous. That is, they either occur fewer than 3 times in total, or more than 90% of occurrences in a given context are predictable. These cases represent a combination of annotation errors in syntactic labels, or infrequent occurrences that would not substantially reduce our performance. The remaining 78 tokens were found to be genuinely ambiguous and frequently occurring, and thus will affect the performance in this syntactic metric, though not in the actual combined parsing task. Examples of these ambiguous tokens include three, which is consistently annotated as a CARDINAL, but has ambiguous syntactic labels CD (48 times) and NNP (11 times). Similarly, yen is consistently annotated as UNIT, but should have syntactic labels NN (173 times) or NNS (196 times). later is labelled REL, but should be given syntactic label RB, JJ, RBR or JJR . Adding additional parent information does not disambiguate the instances; when occurring with a parent with label DATE, the node should

have as POS: RB (70 times), RBR (17 times), JJ (5 times), and JJR (1 time). Unfortunately, there is no way to correctly make these decisions given only the output of the SUB or SUB LAYER variant trained model, and so we must consider the results of syntax only analysis with the knowledge that they do not represent how well the SUB and SUB LAYER models have learnt the structure. They do, however, offer a lower bound on performance that can be directly compared to other variants' models.

A parsing model was also trained on a syntactic-only version of the LOW variant. The difference between this result and our other syntax only results demonstrates the additional difficulty involved in learning a variant with both syntactic and nested named entity labels.

Looking at the LOW and POSLOW variants in more detail, we see that on the syntactic component only, POSLOW outperforms the model trained on LOW. This is in line with our prediction that by not deciding on NE labels with token information visible, the models cannot predict NE labels and structure accurately. This would, however, not affect the syntactic component specifically, and indeed we see that the syntactic component of POSLOW is higher than LOW.

The results for the SUB variant are lower than the syntax only variant and the HIGH variant, reflecting the fact that these model did not learn the syntactic equivalent of the NE substructure, since it had been substituted out by NE labels. As expected, the SUB LAYER model performed slightly stronger, due in part to the fact that more of the original syntactic labels, (e.g. NP layers) are preserved and learnt by models trained on this variant.

To investigate to what extent the bracketing changes in our merged corpus, specifically changes to PP attachment, make the corpus more difficult to learn, we trained and tested the syntactic only component of a version of the LOW variant which excluded any PP changes, the results of which are in Table 6.4.

We found that recall and precision improved only a small amount, and scores were still below pure PTB scores though higher than the syntax only variant. It is, therefore, not the potential inconsistency in changing the analysis of PPs that include nested named entities, but not changing other PPs that is impacting on our performance, but rather, that the combined task of NNE and syntactic parsing is a more challenging task.

Variant	Valid	P	R	F
pure PTB	1916	90.86	90.36	90.61
syntax only	1917	90.27	88.98	89.62
JOINT	1918	89.26	88.82	89.04
JOINT no PP	1918	90.29	89.46	89.87

Table 6.4: Eval-B Analysis of Syntax only versions from JOINT in Section 00 for a model trained only on syntactic output of the JOINT variant, a model trained on JOINT and evaluated only on the syntactic component of the output, and a model trained and tested only on the syntactic component of the JOINT variant that excludes all PP changes.

6.3.3 How well does a combined model learn nested named entity structure?

Similar to the 'syntax only' analysis in Table 6.3, in order to assess how well the models learn the nested named entity structure, we also evaluate only over these labels by removing all syntactic structure. We ran two sets of experiments here, one where all syntactic labels and POS tags were removed from the test data, replacing all POS tags with the label 'O'. These results are shown in the first three columns of Table 6.5

Variant	'O' and Entities				POS and Entities			
	P	R	F	Tag Acc	P	R	F	Tag Acc
Entity only	91.48	77.50	83.91	96.84	91.35	77.62	83.93	93.38
HIGH	78.85	70.86	74.64	96.82	78.90	70.88	74.68	93.87
LOW	90.56	79.73	84.80	98.38	90.57	79.74	84.81	95.59
POSLOW	79.44	73.72	76.47	96.61	79.45	73.72	76.48	93.67
JOINT	89.83	86.47	88.12	98.25	89.85	86.47	88.13	95.38
SUB	89.56	86.83	88.17	98.24	89.59	86.88	88.21	95.46
SUB LAYER	89.51	84.53	86.95	98.44	89.54	84.53	86.96	95.73

Table 6.5: Eval-B Analysis (Precision, Recall and F_1-score) and Tagging Accuracy of Entities only over section 00 using the label 'O' all for non-entities, or using POS tags for non-entities, and tagging accuracy for each model.

The other experiment had a similar setup, with all syntactic labels other than POS tags being removed. The results for this nested named entity and POS analysis are shown in the last three columns of Table 6.5.

6.3.4 Discussion

When considering the results outlined in Section 6.3, it is important to keep in mind what can and cannot be directly compared. In many respects, the Gold Standard of each variant is a moving target. The HIGH and LOW variants have, as a raw count, many more nodes than the SUB variant.

The results of these experiments remain largely inconclusive when considered in the context of selecting one substantially superior variant. They do, however, validate our hypothesis that the HIGH variant will perform better that LOW on a purely syntactic evaluation, and that the LOW variant will outperform HIGH when evaluating on NNEs.

As expected, the HIGH variant outperforms the LOW variant when evaluating purely on syntactic structure. By adding NNE structure between the token and its POS, or additional syntactic label structure, we are making those syntactic decisions more difficult.

Likewise, the HIGH variant does not perform well when evaluated only on the NE structure. In this variant, the POS and syntactic labels interleave between the token and the NE layer to be labelled. Since the parser used has no 'grandparent features', this interleaving means that the token itself cannot be seen when making a named entity label decision.

Corroborating this hypothesis is the performance of our POSLOW variant, which outperforms the LOW variant on syntactic evaluation but is beaten by HIGH, and which similarly outperforms HIGH on NNE only evaluation but is beaten by LOW.

Of particular note are the substantial improvement in NNE only evaluation in our JOINT, SUB and SUB LAYER variants when compared to the HIGH, LOW and POSLOW. These imply that these variants, which either replace POS tags or augment them, learn a much more accurate model of the structure of the nested named entities.

6.3.5 Error Analysis: a more meaningful metric

Although F_1-score is the standard metric through which parser performance is measured, it does not offer much insight into the linguistic nature of parser errors. Further, as discussed in Section 6.3.1, the different numbers of brackets in the gold standards for each of our variants mean we cannot directly compare F_1-score results. A more useful metric for our experiments is a detailed analysis of errors.

We follow the error analysis of Kummerfeld et al. (2012), who propose specific classifications of linguistically meaningful types of errors. This error

analysis method uses tree transformations to classify different linguistic types of parser errors. The system identifies the shortest path from the system output to the gold-standard tree using individual tree transformations as each step. The resulting metric is a measure of the amount of subtree movement, node creation and node deletion that is required to fix each parse tree error. These transformations are then classified into one of several specific error types.

This analysis allows us to further analyse the causes of the drop in performance in our models, and allow us to compare errors caused by each different variant. This error analysis augments the precision, recall and F_1-score statistics in Tables 6.2, 6.3 and 6.5, which do not provide linguistically meaningful intuition for the source of the errors.

Much like the analysis in Table 6.3, in order to evaluate only on comparable output, we train our models (HIGH, LOW, POSLOW, JOINT, SUB, SUB LAYER) and then remove all non-syntactic nodes and labels from our data, leaving only POS tags and syntactic node labels. We follow the same process described in Section 6.3.2 to create a syntax only version of the SUB and SUB LAYER variants. We compare these to two base variants: one trained and tested directly on the Penn Treebank, ('pure PTB'), and another trained on only the syntactic component of output from our LOW merged variant.

Kummerfeld et al. (2012) split the errors into the following categories, and report both the number of individual errors of each type as well as the number of nodes affected by each type of error.

PP Attachment in which the transformation involved in correcting an error included moving a Prepositional Phrase, or the incorrect bracket is a PP.

NP Attachment in which NPs had to be moved to correct an error.

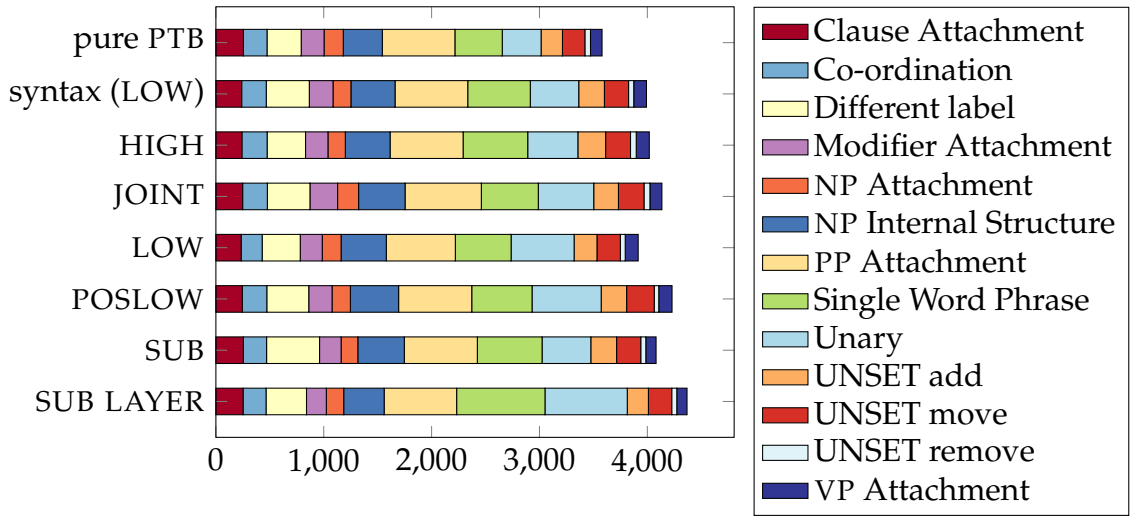


Figure 6.11: Error type breakdown over section 00 for pure PTB, a model trained only on the syntactic output of the LOW variant, and each of our variants. The most frequent error types are discussed below.

Modifier Attachment in which adjectives and adverbs are incorrectly placed. This also includes errors corrected by subtree movement or by creation of a node.

Clause Attachment in which an S node must be moved.

Unary in which unary productions are not linked to a nearby error such as a matching additional node, or a missing node.

Coordination in which a conjunction is an immediate sibling of a node that is moved, or is the left- or rightmost node that is moved.

NP Internal Structure in which NP internals such as ADJP, NX, NAC or QP is incorrect, or our nested named entity structure, including added NML.

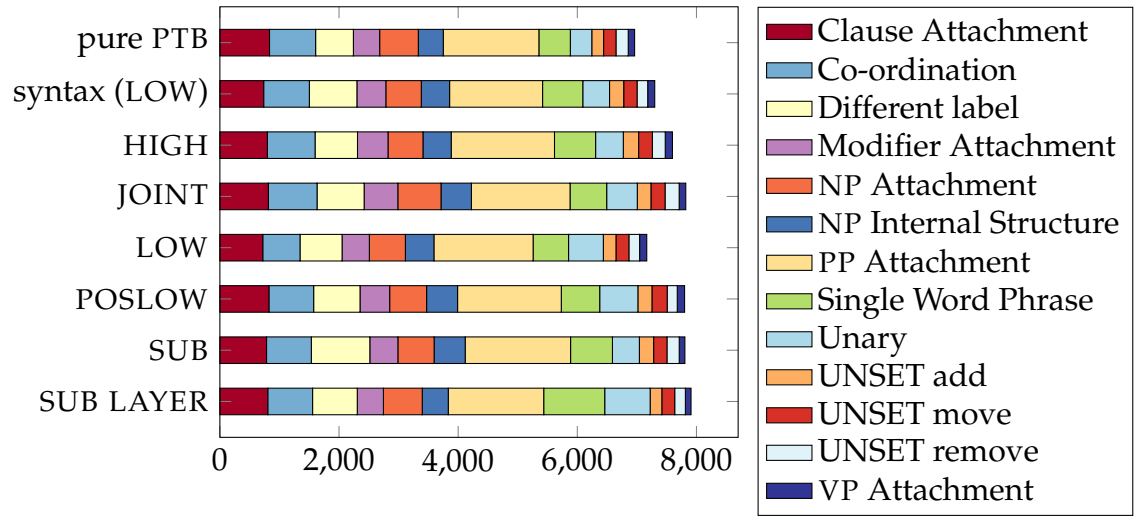


Figure 6.12: Total number of nodes affected by errors, by type, in section 00, for pure PTB, a model trained only on the syntactic output of the LOW variant, and each of our variants.

Different label in which a node has the correct span of children, but the incorrect label.

This prevents having this label error classed as two separate errors: one extra label and one missing label. Note that Different label applies to cases where a non-terminal span exists in both the gold and parsed data, but with different labels, and as such, does not include POS errors.

Figures 6.11 and 6.12 show the numbers of errors of different types in each of our models. Comparing to a model trained directly on the PTB, ('pure PTB' in figures 6.11 and 6.12), our models do have more errors, but not unexpectedly or disproportionately so. We further found that, overall, our models are more or less comparable to one another, with no single model performing substantially worse than others.

In Figure 6.11, comparing the pure PTB variant to a model trained and tested only on the syntactic component of our merged LOW variant, we can see that more errors occur in our processed model, especially errors of a Different Label,

Single Word Phrases and Unary rule types (317 vs. 399, 439 vs. 580, and 360 vs. 449 respectively).

When looking to minimise errors, the LOW variant outperforms our other variants, including the 'syntax only' variant, producing more than 200 fewer errors compared to the other models trained with nested named entity information. The SUB variant also performs strongly, but is hampered by a substantial increase in labelling errors. This is in large part due to the fact that we are not evaluating directly on the produced labels, since in creating the SUB and SUB LAYER variants we substitute our NE labels over the syntactic node labels, and these are statistically regenerated for this evaluation, as described in 6.3.2.

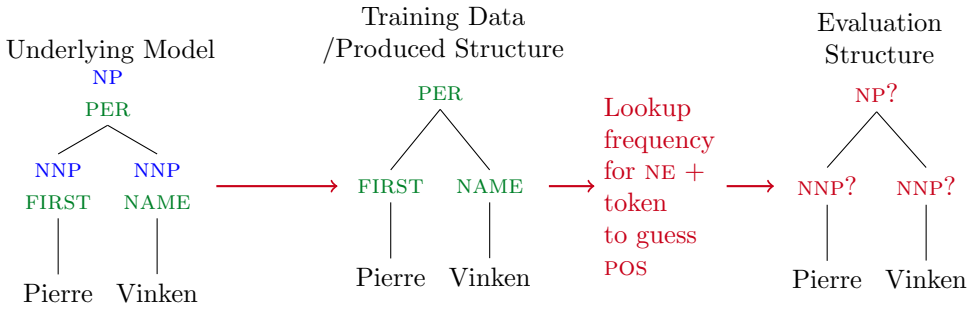


Figure 6.13: Process for evaluating SUB and SUB LAYER variants.

Specifically, when adding syntactic node labels back into the tree for evaluation, the only information that we have to make these decisions is the nested named entity label that the original label was substituted by. That means that all PER nodes are replaced, for example, with NP nodes. In more complicated structures however, for example TIMEX or NUMEX structures, the substitution isn't always as clear cut. Take the example of DATE, which is frequently applied to nodes with label PP (e.g. 'from 1986 to 1988'), ADVP (e.g. 'a year earlier'), NP (e.g. 'the first week of March') and NML. All of these would be given the syntactic category that was most frequent in the training corpus. With this in mind, the larger numbers of error on Different Label is quite understandable,

and does not necessarily represent a decrease in label quality in the combined syntactic NNE model.

We also look in more detail at POS and syntactic label confusion in each model, the results of which can be seen in Figure 6.14. The chart shows the number of POS or syntactic label confusion for each of the models. The 30 most frequently occurring types of confusion occurring in the 'pure PTB' model are shown, as well as the 'long tail' of errors, amalgamated into one 'Other' category.

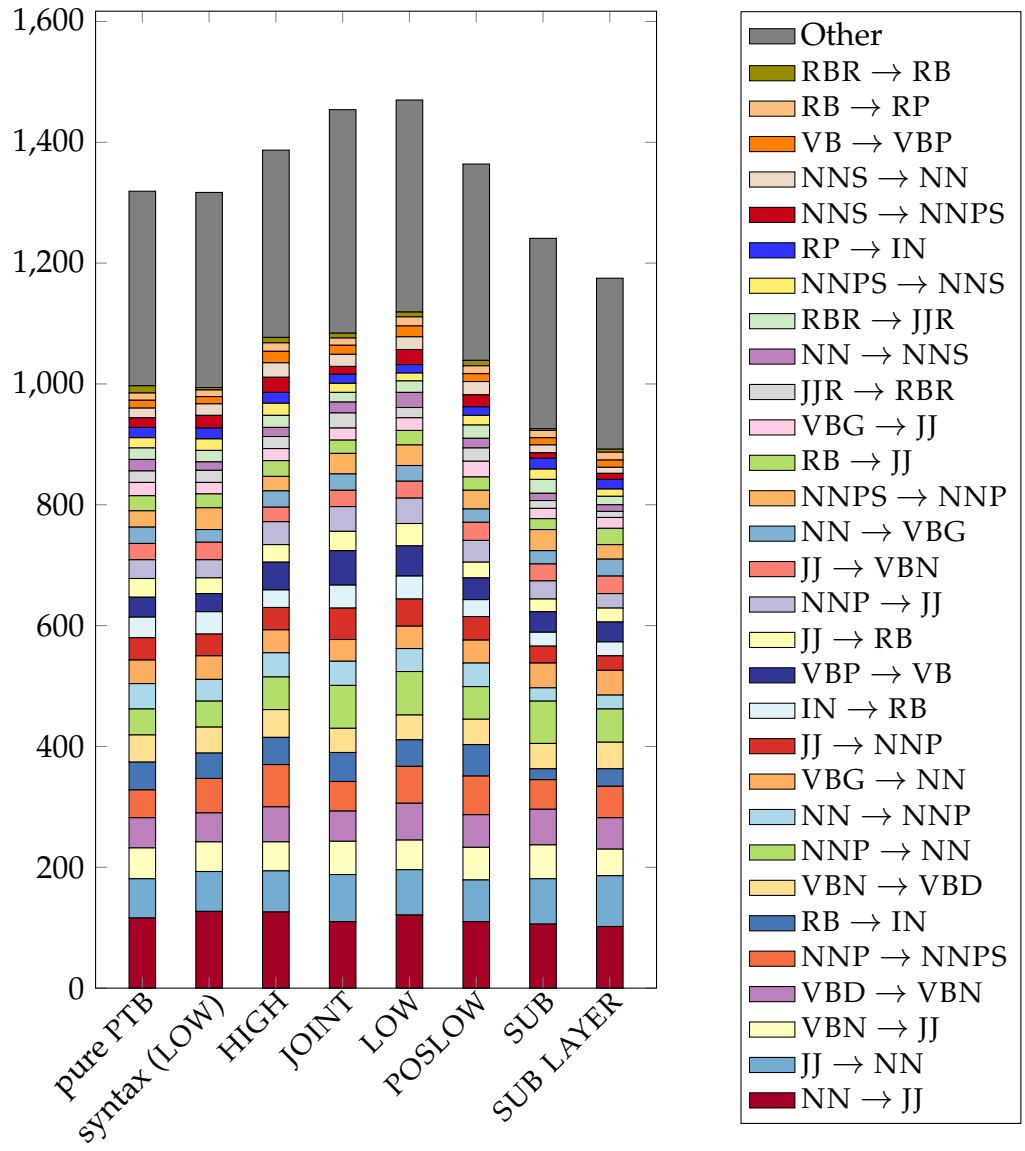


Figure 6.14: POS and phrase label error confusion for each model in section 00. Shows the individual 30 most frequent incorrect (gold to 'found') POS or phrase label errors in section 00, and 'other' category showing length of long tail.

Interestingly, despite the additional step of adding POS and syntactic labels back into the SUB and SUB LAYER variants, they outperform all other variants, and the 'pure PTB', in minimising label errors. Overall, the specific category results are not particularly surprising. SUB and SUB LAYER reduce the numbers of errors in most categories by a small margin, with more substantial improvements in correctly labelling nodes as RB rather than IN or JJ . Some of

Sentences		Bracketing			Compl.	Avg	No	2 or less	Tagging
Len	Valid	P	R	F	Match	cross.	cross.	cross.	accuracy
SUB	2415	83.73	87.84	85.73	0.00	0.95	64.60	87.12	95.64

Table 6.6: Eval-B Analysis of SUB LAYER variant over 2415 sentences in Section 23.

these specific category differences can, however, be explained in part by the frequency-based method used to reinsert missing syntactic labels and POS tags. For instance, the SUB variant makes far fewer errors of NN incorrectly labelled as NNP than other models. It does, however, make a considerable number of errors where NNP is mislabelled as NN . That is, it is favouring NN over NNP in a number of cases, which reduces some but increases other errors.

We conclude that the SUB variant best encapsulates the data. It offers the best balance of strong performance both on syntactic and nested named entity evaluation. From a linguistic point of view, the SUB variant best represents the underlying grammar of named entities. Specifically, named entities, in many cases especially NUMEX and TIMEX expressions, do not always follow standard grammatical rules, but instead have different, systematic and regular structures. By encoding these entity specific grammars directly into the one grammatical model, we can improve the linguistic integrity of our data without facing a large penalty on performance.

The results of the SUB variant on section 23 can be seen in Table 6.6.

6.4 Summary

This chapter introduced a number of different ways to combine compatible nested named entity annotations to syntactic constituents, and evaluated the impacts of each of these variants on the task of parsing. We used the Berkeley Parser (Petrov et al., 2006), a standard syntactic parser, to learn the combined

task of syntactic and semantic named entity parsing, without the introduction of any specialised NER features, and found very promising results.

While we found that the changes made to the syntactic constituents as part of the merging procedure described in Chapter 5 did increase the difficulty of the corpus, as did the introduction of these additional NER nodes, the Berkeley parser was able to reliably learn these models. By evaluating further on only the syntactic or NER component of the joint model, we found that the syntactic structure improved the accuracy of predicting named entity structure, showing that syntax is useful in NER. Further error analysis explored the impact different variants had on both learning NER components and various syntactic components, and we found that a number of different variants perform well on both subtasks.

We have thus shown that combined syntactic constituents and named entity structure can be reliably learnt using existing systems. The next concept to explore is how well NER systems, with features specific to named entity recognition, perform on the task of structured named entity recognition.

7 Recognising Nested Entities

In the previous chapter, we explored how well a standard constituency parser can learn nested named entity structures, finding that these structures can be reliably learnt. In this chapter, we consider the inverse proposition: how well can a standard named entity recognition system learn structured entities. To do this, we must first devise a number of different projections of these structured entities into flat individual labels.

7.1 NER background

We first give a background of NER methods used to learn nested NE structure from our dataset, with specific note of evaluation metrics, and methods used by the C&C NER tagger (Curran and Clark, 2003b) and the state of the art LIBSCHWA NER system (Dawborn, 2015) which we use in our experiments.

Extensive literature on the subject of NER exists (see Sekine and Ranchhod (2009); Nadeau and Sekine (2007); Tjong Kim Sang (2002); Chinchor (1998) for a review). The main approaches fall into three categories: hand-crafted, machine learning and hybrid systems. Hand-crafted approaches involve manually created rules, and use gazetteers. These approaches are very labour intensive, requiring experts in the target domain. Machine learning methods predominantly train supervised models on annotated training corpora to infer the lexical, orthographic, syntactic and contextual features associated with named entities.

For English, a number of datasets exist which can be used for this purpose, discussed in Section 2.3.

Particularly challenging aspects of the task include semantic ambiguities such as abbreviations, nicknames and nested expressions. These can be alleviated by world knowledge sources such as gazetteers, but these are prohibitively expensive to continually update to ensure coverage.

Various methods of detecting named entities have been developed, including semantic, syntactic and statistical approaches. McDonald (1996) first introduced the concept of *internal evidence* (word-level features such as InitialCaps, or "Ltd." within an ORG entity) and *external evidence* (evidence gathered from context, e.g. titles such as "Dr." or "Mrs." before a PER element), and various early systems utilised these linguistic cues in manually-constructed rules. As the area developed, particularly with the release of large annotated corpora, statistical machine learning tools gained more popularity. In these approaches, a system learns patterns from an annotated training corpus, allowing it to predict the most likely NE in a given context. Given appropriate training texts, a single machine-learning system may easily be applied to varying languages, domains and classification schemes.

In 1999, Bikel et al. suggested that each entity class can be described by its own language, and constructed an NER system using class-specific Hidden Markov Models, which are dependent on having previously seen patterns.

7.1.1 Machine Learning Approaches to NER

By CoNLL 2002 and 2003, the focus of NER was shifting to cross-domain modelling which required more complex detection than manually-constructed pattern-matching could allow for. Various machine learning techniques were applied, such as AdaBoost (Carreras et al., 2003), Maximum Entropy Modelling (Tjong Kim Sang and De Meulder, 2003) and Conditional Random Fields (CRFs)

(McCallum and Li, 2003). The most successful ensemble learner at CoNLL-2003 combined the classification decisions of a number of machine learning techniques (robust linear classifier, maximum entropy, transformation-based learning and Hidden Markov Model). Florian et al. (2003), attained strong performance without using gazetteers or other additional training resources, reporting 91.6% F-score on the development set (no result reported for test).

The results of the CoNLL 2002 shared task showed that whilst choosing an appropriate machine learning technique affected performance, "the choice of features is at least as important." (Tjong Kim Sang, 2002) It became clear that the success of machine learning for NER was not simply dependent on the strategy used, but rather, on the training data and feature sets which were incorporated into system design. The most common features that each NE system in CoNLL 2003 used were the creation of neighbouring n-grams, POS-tags, affixes, capitalisation patterns and gazetteers (Tjong Kim Sang and De Meulder, 2003). More contextual features were also used, including using features such as capitalisation patterns, possible expansions of acronyms and NE classes assigned to previous occurrences of terms seen elsewhere in the training data. One high-scoring system distinguished itself by using character-based as well as lexical models (Klein et al., 2003); another utilised the global context of the document as well as local features of a particular word (Chieu and Ng, 2003). The overall results of the task showed that a number of systems could achieve good results (see Table 7.1).

Curran and Clark (2003b) introduced a maximum entropy NER tagger. The NER tagger is built on previous work from Curran and Clark (2003a) on training CCG supertaggers, adding specific NER features. The system uses a large variety of features, and uses Gaussian smoothing, which allows a large number of sparse but informative features to be used without overfitting.

statistic	English	German
baseline	59.6	30.3
median	84.2	67.9
maximum	88.8	72.4

Table 7.1: Baseline, median and maximum F_1-score for the 16 entrants in the CoNLL 2003 shared tasks in English and German (Tjong Kim Sang and De Meulder, 2003).

The near state of the art results of Collobert et al. (2011) suggest that deep learning approaches such as semi-supervised representation learning can help remove the need for extensive feature engineering.

7.1.2 State of the Art performance in English NER

Three main publicly available NER systems are currently state of the art: the Stanford NER system (Finkel et al., 2005), the University of Illinois' Named Entity Tagger (Ratinov and Roth, 2009), and the LIBSCHWA NER system (Dawborn and Curran, 2014; Dawborn, 2015) from the University of Sydney. The performance of these three systems on the OntoNotes 5 corpus is shown in Table 7.2.

The Stanford NER system (Finkel et al., 2005), also known as CRFClassifier, is distributed as part of the CoreNLP suite of NLP tools. It uses a conditional random field with L-BFGS (Nocedal and Wright, 1999) for numerical optimisation. The Illinois Named Entity Tagger (Ratinov and Roth, 2009) uses regularised averaged perceptron (Freund and Schapire, 1999) and beam search for decoding.

The libschwa NER system utilises document structure annotations using DOCREP (Dawborn, 2015). Similar to the Stanford NER system, it also uses a linear chain CRF and L-BFGS (Nocedal and Wright, 1999) for numerical optimisation. The libschwa NER system is discussed in more detail in Chapter 7.4.1.

System	dev	test
Illinois 2.8.2	82.32	84.00
Stanford 2015-01-30	81.93	84.51
LIBSCHWA	84.12	85.98

Table 7.2: Performance of NER systems outlined in Section 7.1.2 on the OntoNotes 5 English splits proposed by Passos et al. (2014). LIBSCHWA NER numbers from Dawborn (2015).

7.1.3 Unsupervised and Distantly Supervised Approaches

Rössler (2004) uses a form of lexical bootstrapping, statistically deriving words to be used as cues from a small annotated corpus. They describe three levels for detecting NEs.

Firstly, on the local level they observe a single occurrence of a word form in context and the semantic label assigned to it. The deliberate meaning of a word form (i.e. the semantic label) is unambiguous, excluding intended ambiguity aiming at comedic or poetic effect. Some of the word forms occur in predictive contexts and can be tagged with NE labels with high reliability.

Secondly, on the discourse level all occurrences of a word form with a text unit are observed, along with the semantic labels assigned to them (Gale et al., 1992). A word sense located on the discourse level can be seen to have a strong one-sense-per-discourse tendency whereby various occurrences of a polysemous word will tend to belong to the same semantic class within one discourse. Rössler found that in the complete CoNLL 2003 the 'one-sense-per-discourse' was accurate on 93.5% of the data. They further found that word forms tagged with different labels within the single discourse were most often found to consist partially of locations (e.g. "Deutsche Bank"), people (e.g. "Phillip Morris") or regular nouns (e.g. "Sport Factory"), indicating that this token level ambiguity would potentially constitute a considerable source of error. This is strong evidence for the importance of nested named entities.

Thirdly, on the corpus level all the occurrences of a word form within all the texts available for the application were observed. The larger the corpus, the more likely a particular word form was seen as a member of two or more semantic classes.

Another unsupervised approach by Etzioni et al. (2005) combines NE lists with disambiguation rules. An advanced system of generating lists of named entities for a class 'X' searches the web for phrases like "X , such as [Y]" and attempts to find lists of items 'Y'. Nadeau et al. (2006) use similar automatically-acquired lists for marking entities in texts, along with unsupervised means of disambiguating entity-noun ambiguity, entity-entity ambiguity, and entity boundaries. Using only their web-derived lists and some language-independent algorithms, their system outperformed the MUC-7 baseline, but could not compete with its top entrants.

Chiticariu et al. (2013) and others continue to argue in support of rule-based approaches, in large part due to precision, introspection and the fast selective adaptation of rules for new domains and types. While rule-based information extraction systems are widely regarded as dead-end technology by academia, they remain used in the commercial world. Chiticariu et al. (2013) argue that this disconnect stems from the discrepancy between how the two communities measure the benefits and costs of IE, as well as academia's perception that rulebased IE is 'devoid of research challenges'.

7.1.4 Gazetteers

External resources such as gazetteers have been used extensively in NER systems to varying extents (Tjong Kim Sang, 2002; Florian et al., 2003; Tjong Kim Sang and De Meulder, 2003), specifically aiming to provide robustness against unseen entities. Gazetteers are, however, costly to produce and maintain, and many quickly become obsolete.

Krupka and Hausman (1998) show that there is very little performance loss when reducing 25,000 gazetteer entries to 9,000, and conversely show a dramatic improvement with a selection of 42. Mikheev et al.'s (1999) system was also tested with and without supplementary gazetteers, finding that not using the gazetteer gave only a small reduction in ORG and PER class accuracy, but significantly worsened performance for LOC (from 6% to 40%). Mikheev et al. (1999) argue against the necessity of gazetteers, pointing out that using list-lookup techniques require gazetteers to be enormous and constantly updated to cover naming variations, and cannot avoid ambiguity with common nouns and between entities. These issues with the use of gazetteers seem to stem from two main problems: gazetteer coverage and entity-entity ambiguity.

Rössler (2004) avoid limitations associated with the use of gazetteers and other handcrafted rules by using a knowledge-poor approach, refraining from using any additional linguistic tools such as a morphological analyser or POS-tagger, any handcrafted linguistic resources such as dictionaries, or any handcrafted knowledge providing lists, such as gazetteers, lists of NEs or lists of trigger words. They use a second order Markov model trained on a comparatively small annotated corpus (100,000 tokens annotated by a single student) and statistically generate a list of words providing evidence for NNEs.

More recently, Lin and Wu (2009) and Tkachenko and Simanovsky (2012) explored the use of word and phrase clusters as a substitute for a gazetteer.

7.1.5 Evaluating NER

Evaluating various NER methods in detail poses various problems. Many categories of ambiguity make it difficult to establish an appropriate evaluation metric (Nadeau and Sekine, 2007). The output of NER methods may differ from the annotation of various corpora it can be tested against in several crucial aspects including granularity, extent and markup, making it difficult to com-

pare different methods. Firstly, the use of different classes of named entities, or different granularities (e.g., organisation is often split into 'company', 'university', 'government' etc.) can effect results. Different annotations schemes may mark-up the extent of a NE differently: e.g., a person name may or may not include a function and a title ('President George Bush' vs. 'George Bush'). Furthermore, the markup of the corpus may be textually oriented (e.g., as XML tags) while the output of other grammars may be in the form of semantic structures. Differences such as these often arise because existing corpora and annotation schemes are developed for different purposes and are re-used, or the output structure of the grammar may be changed after corpora have been annotated. These differences pose various challenges for testing and evaluating NER methods with respect to a corpus, since a NE may be recognised correctly according to the intentions of one grammar, but may be annotated differently in the corpus.

Some NER evaluation metrics convert entities into a sequence tagging problem, using a specific encoding. BIO encoding, Ramshaw and Marcus (1995), adds a label to each tag that indicates whether it is at the beginning (B), inside (I) or outside (O) of an entity. In this way, consecutive entities of the same type are identified as separate entities. Other encodings exist, such as BMEWO (beginning, middle, end, single word, outside). Sang and Veenstra (1999) experiment with using various different encodings within the NE tagger.

MUC (Chinchor, 1998) equally awarded achieving a match on TYPE, where an entity's class is identified with at least one boundary matching, and TEXT, where an entity's boundaries are precisely delimited, irrespective of the classification assigned. This equal weighting is unrealistic, as some boundary errors are highly significant, while others are relatively arbitrary.

CoNLL 2003 only awarded exact phrasal matches, ignoring boundary issues entirely. Manning (2006) claims that this evaluation method is biased

towards systems which leave entries with ambiguous boundaries untagged, since boundary errors amount to both false positives (where an entity is tagged which is not marked up in the gold-standard annotation) and false negatives (where an entity tagged in the gold-standard data is not matched).

Tsai et al. (2006) explore a number of approaches to evaluating NER, including relaxing entity boundary requirements by matching only the left or right boundary, having any tag overlap, or incorporating a more semantically based matching method or per-token measures.

In 2003, Bering et al. developed a diagnostic and evaluation tool (jTaCo) which allows user-defined mappings between different NE classes, for controlled partial overlap between recognised and annotated NEs, and supports user-defined mappings between text-based and semantically-based annotations and output structures. This is not widely used however, due to various limitations and low up-take.

The difficulties in evaluating NER are unresolved, and results reported using different evaluation methods are incomparable. In order to evaluate NER accurately, annotation schemes must be closely evaluated and, in many instances, remapped for recall, precision and F_1-score.

7.2 Nested Named Entity Recognition

An extensive body of work exists on named entity recognition, but comparatively little of it focuses on nested structures. This is primarily due to the relative scarcity of corpora annotated with nested NE structure, limiting most work in the field to the biomedical domain (GENIA), or languages other than English (Carreras et al. (2003) in Spanish and Catalan, and Fu and Fu (2012) in Chinese).

Section 2.4 outlined a number of corpora which contain nested structures, work on which will be discussed below.

7.2.1 Nested NER in non-English, and other domains

In the biomedical domain, early work on the GENIA corpus only focused on the innermost entities, rather than the full nested structures. In 2004, Zhang et al. (2004) and Zhou et al. (2004) developed methods of capturing the nested, or 'cascading', structure of GENIA entities. Zhou et al. (2004) develop a pattern-based post-processing for cascaded entity name resolution, identifying six patterns, extracted from the cascaded entity names in the GENIA training data. The pattern-based post-processing was found to be successful for identifying cascaded entities, improving overall F_1-score by 3.9 %.

Zhang et al. (2004) build on this post-processing rule-based cascading recognition approach, comparing it to a a Hidden Markov Models (HMM) with back-off modeling and cascaded recognition, which used one HMM to identify short, embedded entities, and another HMM model to iteratively extend these short entities, capturing the nesting. They found the post-processing rule-based approach performed slightly better than the HMM-based approach, achieving 66.5% F_1-score, compared to 64.2% F_1-score.

Gu (2006) approach the task of recognising nested named entities as a binary classification problem, and solve it using Support Vector Machines (SVM). Though the PKU corpus (Fu and Luke, 2005) (see Section 2.4.5) contains entities with up to four layers of nesting, Gu elect to simplify the task to only consider a single level of nested entities. This allows them to reformulate the task as a dual-layer cascaded chunking task on a sequence of words. For each token in a nested NE, two schemes are used for classification. For the outer layer of entities, they use what they describe as the "traditional BIO tagset", though expand on this with explanation that 'B' indicates the token is at the beginning of a multi-token entity, 'I' denoting the middle or the end of a multi-token named entity, and 'O' denoting "that the token is an independent NE by itself", which

is inconsistent with the more commonly used 'O' being outside an entity. They also present a new encoding method for embedded entities: BIO-E, adding 'E' for the end of a multi-token embedded NE, 'M' for the middle of a multi-token NE and modifying 'I' to indicate the second token of a multi-token embedded entity. Gu (2006) report separate P, R and F_1-score for the outer layer of entities and the single layer of embedded entities.

Alex et al. (2007) develop a variety of techniques for identifying nested NEs on the GENIA corpus, comparing a layering conditional random field (CRF) approach, a cascading approach and a joined label tagging approach, with all approaches aiming to reduce the nested NER problem down to one or more flat 'BIO' problems that can be solved with existing NER tools. The layering method involves each level of nesting being modelled as a separate BIO problem, and the output being combined. This can work either with an inside-out or outside-in direction of layering, with the former identifying the innermost entities, then second-level entities, and so on. For outside-in layering, the outermost entities would be identified by the first CRF, with subsequent layers identifying increasingly embedded entities.

The joined labeling approach reduced the nested labels to one tagging problem by concatenating all BIO tags of all levels of nesting. This method involves a substantially expanded label set, which led to data sparsity issues. The cascading method similarly splits the task into several BIO problems, specifically by grouping entity types and training separate models for each group. Each CRF is applied in a specific order, allowing each to use features from the previously identified entities. One limitation of the cascading method is that it cannot capture nested entities of the same type since each entity-specific model is run only once, and these nested entities of the same type are not an infrequent occurrence in the data. Nevertheless, the cascading method performed most strongly on the data. A Dual-layer CRF approach has more recently been used

in Chinese nested named entity recognition (Fu and Fu, 2012) with promising results.

Byrne (2007) investigate nested named entities in historical archive text, drawing from a dataset of 9,768 written notes, of which approximately 30% are grammatical English sentences (see Section 2.4.6). 9.4% of approximately 27,500 entities were found to have nesting. The method proposed in Byrne (2007) uses a specific length of token window, and concatenates tokens such that each nested entity string has its own separate label. That is, given the example when Edinburgh University Library was ..., each individual token would be considered, in addition to concatenations of consecutive tokens of length two (when_Edinburgh, Edinburgh_University ...), three (when_Edinburgh_University, Edinburgh_University_was, ...) and so on, up to a maximum entity length, decided in advance. This presents a novel way of representing structured data without needing to expand the tagset, though it does substantially increases the number of tokens, thereby increasing the time taken for training the classifier, and also removes the ability to capture entities of more than a specific length. (In the experiments, token length of 6 was chosen, which captured 97.1% of entities in the data.) The now-flattened data was trained with the C&C NER tagger (Curran and Clark, 2003c).

The SemEval 2007 Task 9, (Màrquez et al., 2007), Multilevel Semantic Annotation of Catalan and Spanish, included a nested NER subtask in addition to noun sense disambiguation and semantic role labeling, using data from AnCora. Only two teams participated, one of which did not specifically attempt the nested NER task, ignoring all weak entities, which are those that would have contained nesting. The other team, Màrquez et al. (2007), presented a system that used a pipeline of two classifiers trained with a multiclass AdaBoost algorithm, running the second over phrases of the parse tree which,

syntactically, could be an entity, thus identifying a maximum of only two levels of nesting.

Finkel and Manning (2009a) present a method of parsing nested named entities using a discriminative constituency parser. By including the nested structure of embedded entities, their model allows entities to be influenced not just by the labels of surrounding tokens, as in a standard CRF, but additionally by embedded entities. They represent each sentence as a constituency tree where named entities correspond to a phrase level node, all joined by an S node which joins the sentence components. Finkel and Manning also include POS tags as preterminal nodes, and the tokens themselves as leaf nodes, though no other syntactic structure is included. Each node is labelled with both its parent and grandparent labels, allowing the learnt model to capture the structural nesting of entities. Trees are binarised in a right-branching manner before features are generated, which has the disadvantage of removing the distinction between genuinely right-branching structures and actual flat structures, similar to that discussed in Section 2.2.1.

Building on the technique outlined in Finkel et al. (2008), Finkel and Manning (2009a) train the nested NER model using a discriminatively trained, conditional random field-based, CRF-CFG parser, similar to a chart-based PCFG parser. They add local named entity features (e.g. word, label, shape combinations), pairwise named entity features (over labels for adjacent words), embedded named entity features, as well as whole entity, local POS and joint NE and POS features. Their model outperformed a flat semi-CRF parser on both top-level entities and all entities over the GENIA corpus. They also run similar experiments on the AnCora corpora, achieving promising results, and finding that modeling nested entities does not, on average, reduce performance when evaluating solely on outermost entities.

7.2.2 Joint parsing and Named Entity Recognition

More recently, research efforts have been directed at the task of joint parsing and named entity recognition.

> For named entities, the joint model should help with boundaries. The internal structure of the named entity, and the structural context in which it appears, can also help with determining the type of entity. Finding the best parse for a sentence can be helped by the named entity information in similar ways. Because named entities should correspond to phrases, information about them should lead to better bracketing. Also, knowing that a phrase is a named entity, and the type of entity, may help in getting the structural context, and internal structure, of that entity correct. (Finkel and Manning, 2009b)

Finkel and Manning (2009b) present a joint, discriminative model of parsing and named entity recognition, using a feature-based CRF-CFG parser operating over tree structures augmented with NER information. This joint model of parsing and NER achieves small gains on parser performance and moderate gains on named entity performance when compared with single-task models trained on the same data. The joint representation allows for information from named entities to inform constituency decisions, and conversely, parse information to improve NER decisions. In experiments on the OntoNotes corpus (Hovy et al., 2006), (see Section 2.3.5), they report improvements of up to up to 1.36% F_1-score for parsing, and up to 9.0% F_1-score for named entity recognition, over 4 entity types (Person, Organization, GPE and Misc, reduced from the original 18 categories in OntoNotes). They note, however, the small comparative size of the OntoNotes corpus (200,000 annotated English words) compared to the Penn Treebank. Specifically, the performance of their model trained using the OntoNotes corpus, fell short of separate parsing and named entity models trained on larger corpora annotated with only one type of information (Finkel and Manning, 2010).

Finkel and Manning (2010) builds on Finkel and Manning (2009b), which trains on only jointly-annotated data, to incorporate larger amounts of single-task annotated data, in order to produce a hierarchical joint model. This produces substantial gains over a joint model trained on only the jointly annotated data. A hierarchical prior is used to link feature weights for shared features in both single-task models and the joint model. By ensuring that the joint model has features in common with each single-task model, even if it has additional features which are only present in the joint model, the single-task models and the joint model are able to influence one another via a hierarchical prior.

The hierarchical model performed better than the joint model overall, over various sections of the OntoNotes corpus. Experiments on the smaller corpora show the largest gains, with performance improving up to about 8% F_1-score. Other sections saw a 1% gain on both subtasks, while one section saw an improvement in the parsing subtask, but a small performance decrease in the NER subtask. As a general trend, they note that the hierarchical model helps smaller datasets more than the large ones, which they credit both to lower baselines being easier to improve upon, and due to the experiment setup, the fact that the larger corpora had comparatively less additional singly-annotated data to provide improvements, since that additional data was the remaining, smaller sections of OntoNotes.

7.2.3 Evaluation of Nested Named Entity Recognition

Nested named entity recognition has been evaluated, to various extents, in a number of different ways. Some coreference-like tasks have required the identification of referential units at the nested named entity level, including the KBP EDL task (Section 2.4.7), though this evaluation did not explicitly model nested NEs and their types.

As outlined in Section 7.2.1, many approaches to nested NER evaluation rely on separating the evaluation into two tasks: one on the outermost entities, and a second evaluation on embedded entities. This places an artificial limit on the number of layers entities can be annotated with, and also impacts the evaluation of these entities. If a system only finds the nested named entity, and not the corresponding outermost entity, it is counted as a mistake twice - once for a top-level entity which does not appear in the 'correct' top level entities, and again for missing an embedded entity.

A better approach is for scoring to not be limited to a particular layer of entities (e.g. only outermost layer), but to include all levels of nesting. During scoring, all entities and their start/end offsets from the system output are analysed. If an entity is correct, it should match the type and start/end offset in the gold-standard data. From there, precision, recall and F_1-score can be calculated in the standard fashion, along with the numbers of true positives and false negatives. The parsing results from the previous chapter use essentially this approach, with the EVAL-B metric scoring each layer. In this chapter, we use the standard CoNLL NER evaluation because we are deliberately treating it as a flattened NER task.

7.3 Mapping structured named entities into flat tags

We explore a number of different variants of representing nested named entity structure as a flat label. These variants capture the structure of entities to different degrees, from only learning a flat model of either the top (TOP) or bottom (BOTTOM) layer of annotations, to a complete full structure (STACK) similar to the labels used in calculating inter-annotator agreement (see Figure 4.5 in Section 4.1.5). The variants are outlined below, and then evaluated using

both a state of the art NER system (LIBSCHWA NER and DOCREP) introduced in Section 7.1.2, and a faster system, the C&C (Curran and Clark, 2003b).

7.3.1 TOP

The TOP variant uses the highest NE label for all tokens under that node. In Figure 7.1, Pierre and Vinken are both labelled PER, and 61, years and old are all labelled as AGE.

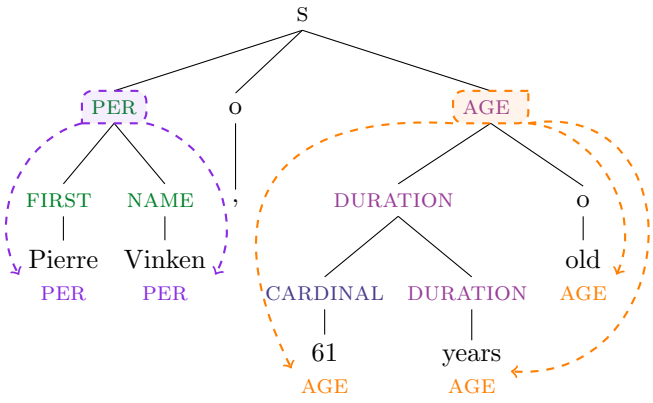


Figure 7.1: Figure showing TOP variant of NER labels

The TOP variant has fewer entities than other variants, as shown in Table 7.3, having some 60,000 fewer entities than any other variant. Since in the TOP variant, all tokens below the topmost entity are included in that entity, it has both fewer entities in total and fewer entity types. For instance, MULT or INI do not occur in the TOP variant, since they are always contained within larger entities.

Figure 7.1 results in two entities: [Pierre Vinken]$_{\text{PER}}$ and [61 years old]$_{\text{AGE}}$.

The TOP variant follows a similar motivation to the standard NER task, although in this case using fine-grained NER categories. It also differs from the standard NER task in that many of the nesting decisions described in Chapter 3 have resulted in different spans. The largest PER span, for instance, includes ROLE and HON tokens that are not usually included in PER entities.

7.3.2 TOP$_2$

We introduce the TOP$_2$ as a compromise between the standard NER task, and that of learning the full structure of an entity. TOP$_2$ is one step towards full modeling of the structure of an entity.

In the TOP$_2$ variant, the topmost two labels (or one, if only one exists) are projected down on each token. Pierre and Vinken are analysed as in the STACK variant, since only two layers of entities exist between the token and the sentence root. For the other tokens, both 61 and years have the same analysis, as shown in Figure 7.2. They therefore end up with the same label, DURATION_AGE, and are combined to form one entity. Since old only has one entity layer above it, it keeps the label AGE.

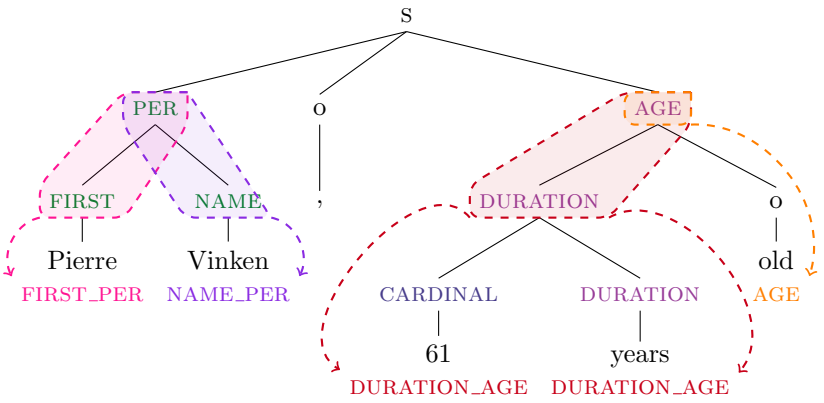


Figure 7.2: Figure showing TOP$_2$ variant of NER labels

The TOP$_2$ variant, therefore, has four separate entities for the example sentence fragment: [Pierre]$_{\text{FIRST\_PER}}$ [Vinken]$_{\text{NAME\_PER}}$ [61 years]$_{\text{DURATION\_AGE}}$ [old]$_{\text{AGE}}$

We expect the TOP$_2$ variant to perform slightly worse than the TOP variant, since the number of categories, 906, is substantially higher than the 109 categories in the TOP model. Nevertheless, it represents a step towards learning the complete nested structure, and we anticipate it will offer a good balance of information learnt, performance and training time.

7.3.3 BOTTOM

The BOTTOM variant uses the closest NE labels to the token to label each token. In Figure 7.3, we can see that Pierre is marked as FIRST, and Vinken is labelled NAME. Similarly, 61 is marked as a CARDINAL, years is a DURATION and old, which forms part of the larger AGE span, is marked directly as AGE. In total, the BOTTOM variant has five separate entities for the sentence fragment in Figure 7.3: [Pierre]$_{\text{FIRST}}$ [Vinken]$_{\text{NAME}}$ [61]$_{\text{CARDINAL}}$ [years]$_{\text{DURATION}}$ [old]$_{\text{AGE}}$

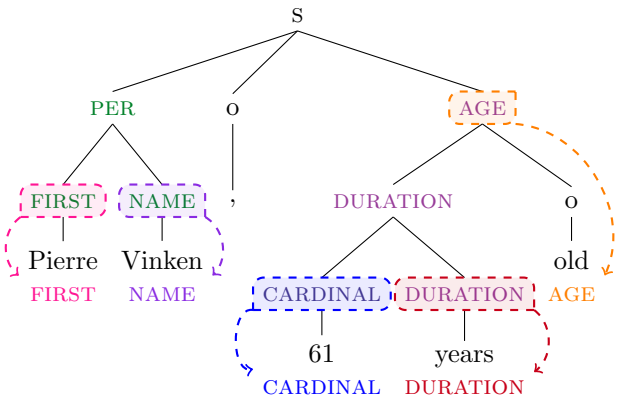


Figure 7.3: Figure showing BOTTOM variant of NER labels

BOTTOM does not capture nesting information, and consequently has very high level of token consistency. We consequently expect the BOTTOM variant to be the easiest to learn, since the high token consistency also reduces the issue of span determination.

Consider the annotation:

During the [[[quarter]$_{\text{CARDINAL}}$]$_{\text{DURATION}}$]$_{\text{DATE}}$, [[Delta]$_{\text{NAME}}$]$_{\text{CORP}}$ issued. . . .

In the BOTTOM variant, the two entities would be [quarter]$_{\text{CARDINAL}}$ and [Delta]$_{\text{NAME}}$, which does not capture TIMEX or CORP information, but instead follows a very token-consistent model where any referential spans or metonymy are lost.

7.3.4 BOTTOM$_2$

The BOTTOM$_2$ variant is similar to the TOP$_2$ variant, with the distinction whereby nested entity labels are chosen from those closes to the token, rather than furthest away. Figure 7.4 demonstrates this, specifically the tokens 61 and years.

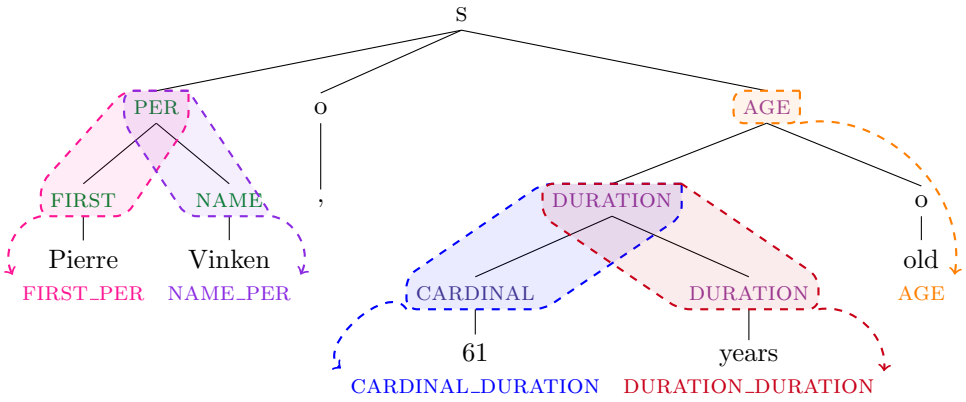


Figure 7.4: Figure showing BOTTOM$_2$ variant of NER labels

The BOTTOM$_2$ variant has five separate entities for the example sentence fragment: [Pierre]$_{\text{FIRST\_PER}}$ [Vinken]$_{\text{NAME\_PER}}$ [61]$_{\text{CARDINAL\_DURATION}}$ [years]$_{\text{DURATION\_DURATION}}$ [old]$_{\text{AGE}}$

As with the TOP$_2$ variant, we expect the BOTTOM$_2$ variant to be harder to learn than the BOTTOM, but will offer one step towards full modelling of the nested structure of entities.

7.3.5 TOP_BOTTOM

The TOP_BOTTOM variant acts as a compromise between the benefits of top-down approaches, which limit the number of separate entities, and bottom-up approaches, which we postulate to better represent the token. In this variant, we use the topmost and bottommost labels for each token, as shown in Figure 7.5.

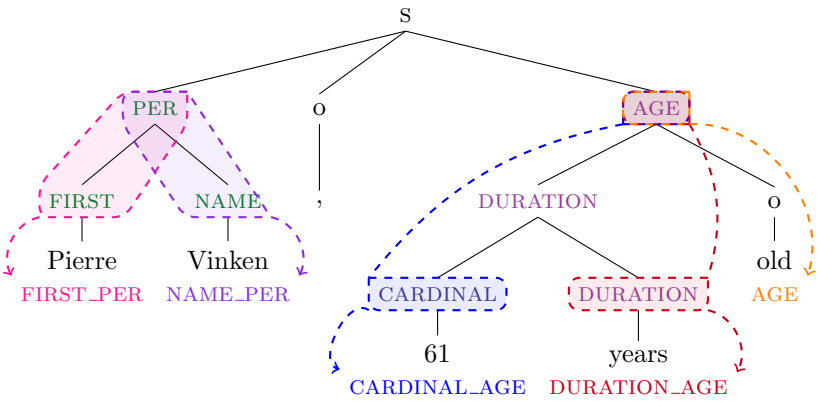


Figure 7.5: Figure showing TOP_BOTTOM variant of NER labels

The TOP_BOTTOM variant has five separate entities for the example sentence fragment: [Pierre]$_{\text{FIRST\_PER}}$ [Vinken]$_{\text{NAME\_PER}}$ [61]$_{\text{CARDINAL\_AGE}}$ [years]$_{\text{DURATION\_AGE}}$ [old]$_{\text{AGE}}$

7.3.6 STACK

The STACK variant creates a new NE label that is a concatenation of all labels above each token. For example, Pierre has a NAME and a PER label above it, and is labelled as FIRST_PER. Similarly, 61 is under a CARDINAL label, nested in a DURATION label, in turn nested in an AGE label, so its STACK label is CARDINAL_DURATION_AGE.

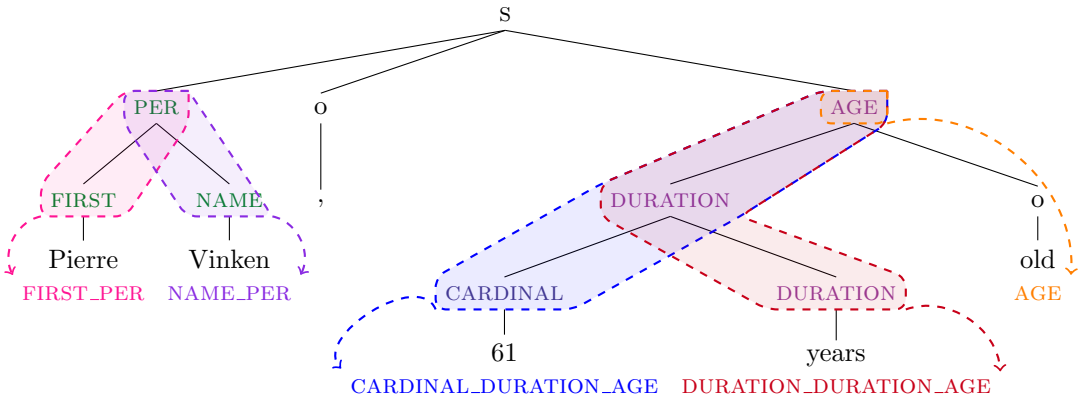


Figure 7.6: Figure showing STACK variant of NER labels

The STACK variant for the sentence fragment shown in figure 7.6 results in five entities: [Pierre]$_{\text{FIRST\_PER}}$ [Vinken]$_{\text{NAME\_PER}}$ [61]$_{\text{CARDINAL\_DURATION\_AGE}}$ [years]$_{\text{DURATION\_DURATION\_AGE}}$ [old]$_{\text{AGE}}$

The STACK variant is similar in form and motivation to the category tags used for interannotator agreement. It is most faithful to the nested named entity task, but is also the most difficult task, with a total of 1738 categories, resulting in very sparse data.

We consider the STACK variant to be similar to supertagger categories in CCG, where each category gives information on how it participates in a larger span. Though the STACK categories do not give direct information on which other categories it collects, it does specify how the token will contribute in the larger entity span.

While we do not expect very strong performance on the STACK variant directly, it is nevertheless an interesting variant to consider. Because the accuracy of supertagging is not very high, many supertaggers supply multiple categories with an adaptive beam approach, or similar, to counteract this and allow for good performance in the CCG parsing task. So too could a larger number of STACK categories be supplied for a bespoke nested NER tool.

7.3.7 COMPRESS

The COMPRESS variant is similar to the STACK variant, with the modification that if two adjacent labels are identical, only one is included in the concatenated tag. Consider years, which in STACK has the label DURATION_DURATION_AGE. In the COMPRESS variant, since DURATION is adjacent (in the label, and nested directly, in the tree structure) to another DURATION label, it is omitted, and the final label is DURATION_AGE. Thus, the five entities from the COMPRESS variant for the example sentence fragment are: [Pierre]$_{\text{FIRST\_PER}}$ [Vinken]$_{\text{NAME\_PER}}$ [61]$_{\text{CARDINAL\_DURATION\_AGE}}$ [years]$_{\text{DURATION\_AGE}}$ [old]$_{\text{AGE}}$

The COMPRESS variant acts as a compromise between the structural complexity of STACK and the smaller category tagsets of other variants. With a total of 1314 categories, roughly 400 more than TOP_2, BOTTOM_2 and TOP_BOTTOM, and 400 fewer than STACK, we expect the COMPRESS to train faster than the STACK model, but still substantially slower than the less sparse variants.

7.3.8 Variant Discussion

Due to the differences in how adjacent tokens are analysed, each variant has a different number of entities, calculated by adjacent non-matched labels, as shown in Table 7.3. Notably, the TOP variant has substantially fewer entities (around 111,000 compared to 170,000 - 190,000), since adjacent tokens under the same entity node all form part of the same entity. Compare this to the BOTTOM variant, where the bounds of each entity are determined by the closes entity labels to the tokens.

The different numbers of entities and entity bounds complicates evaluation, since we cannot directly compare between variants. We can, however, compare different NER methods, and compare them to the output of a parser.

		# Entities		
NE Variant	# Categories	Train	Test	Total
TOP	109	100,523	10,646	111,169
BOTTOM	113	169,811	17,874	187,685
TOP_2	906	155,752	16,467	172,219
BOTTOM_2	909	172,392	18,171	190,563
TOP_BOT	967	170,319	17,937	188,256
COMPRESS	1314	169,787	17,867	187,654
STACK	1738	172,840	18,213	191,053

Table 7.3: Number of categories and entities across different NE variants.

In the NER task, we expect the best performance on BOTTOM, due to token level consistency. We expect the STACK variant to have the lowest performance, as it is the hardest of the tasks, with the most categories.

NE Variant	# Categories with Frequency						
	All	>1	%	>5	%	>10	%
TOP	109	109	(100%)	101	(93%)	97	(89%)
BOTTOM	113	113	(100%)	106	(94%)	105	(93%)
TOP_2	906	707	(78%)	416	(46%)	327	(36%)
$BOTTOM_2$	909	668	(74%)	406	(45%)	323	(36%)
TOP_BOT	967	711	(74%)	424	(44%)	340	(35%)
COMPRESS	1314	868	(66%)	451	(41%)	345	(26%)
STACK	1738	1161	(67%)	605	(35%)	453	(26%)

Table 7.4: Number of categories in each NE variant occurring more than 1, 5 and 10 times, and the percentage of the total categories for that variant.

With the substantially increased number of categories in our more complex variants, shown in Table 7.4, we also introduce data sparsity issues. Table 7.4 also shows the numbers of categories if we impose thresholds of each category appearing more than 1, 5 and 10 times, respectively. We can see that the number of categories which occur only once across the 2+ variants is quite high. In the TOP_2 variant, 199 categories occur only once. That number increases to 240 for the $BOTTOM_2$ variant, and 255 for TOP_BOTTOM. In the COMPRESS variant, the number of category tags occurring only once further increases to 445 (34% of a total 1314 categories), and in the STACK variant, 576 of the 1738 categories (33%) occur only once, and a further 285 categories (16%) occur only twice, creating an exceptionally long tail of categories.

When we look at the numbers of categories for each variant which occur more than 5 and 10 times, we see the gap between COMPRESS and the *two* variants (TOP_2, $BOTTOM_2$ and TOP_BOTTOM) virtually disappears. The number of categories in the STACK variant, however, remains higher even when considering only categories which occur more than 10 times.

Table 7.5 shows the 10 most frequent entity labels for each of the NE variants, and the number of entities of that type in each variant. In the TOP variant, we see that CORP, DATE and PER are the three most frequent entities, compared

to CD, NAME and CORP for the BOTTOM variant. Since NAME is always embedded in another entity, it does not occur in the TOP variant. Similarly, PER does not occur in the BOTTOM variant.

We can see that CORP occurs frequently across all variants. In the 2+ variants (the TOP_2, BOTTOM_2, TOP_BOTTOM, STACK and COMPRESS variants, which combine the labels on two or more nodes to form categories), we see that CORP exists frequently both with and without nested structure. In all 2+ variants, the category NAME_CORP occurs frequently, as do the combinations NAME_PER and UNIT_MONEY. MULT_CD_MONEY is a combination of three layers of nesting, and occurs in the 10 most frequent categories in both the COMPRESS and STACK variants, further supporting the premise that multiple layers of nesting are prevalent throughout the corpus.

Freq.	Token Label	Freq.	Token Label	Freq.	Token Label	Freq.	Token Label
TOP		BOTTOM		STACK		COMPRESS	
45863	CORP	33348	CD	22465	CORP	22520	CORP
31490	DATE	25730	NAME	11705	NAME_PER	13257	NAME_PER
29600	PER	25557	CORP	7838	UNIT_MONEY	11848	NAME_CORP
27329	MONEY	18902	UNIT	7596	NAME_CORP	9271	CD_MONEY
14598	PERCENT	10377	DURATION	7020	GOVERNMENT	9177	UNIT_MONEY
10282	CD	7832	MULT	6548	CD	8104	CD
9094	RATE	7795	GOVERNMENT	6163	CITY	7133	GOVERNMENT
8201	GOVERNMENT	7697	CITY	5839	UNIT_PERCENT	6756	MULT_CD_MONEY
6239	CITY	6754	FIRST	5755	DATE	6726	CD_PERCENT
6168	DURATION	6696	DATE	5712	MULT_CD_MONEY	6257	UNIT_PERCENT
TOP_2		$BOTTOM_2$		TOP_BOTTOM			
22465	CORP	22465	CORP	22465	CORP		
14016	CD_MONEY	13170	NAME_PER	13300	NAME_PER		
12378	NAME_CORP	10958	UNIT_MONEY	11912	NAME_CORP		
11816	NAME_PER	10113	CD_CD	9283	CD_MONEY		
8119	UNIT_MONEY	7798	NAME_CORP	9182	UNIT_MONEY		
7223	CD_PERCENT	7738	MULT_CD	7020	GOVERNMENT		
7078	DURATION_DATE	7020	GOVERNMENT	6792	MULT_MONEY		
7020	GOVERNMENT	6547	CD	6741	CD_PERCENT		
6547	CD	6303	UNIT_PERCENT	6548	CD		
6159	CITY	6218	FIRST_PER	6266	UNIT_PERCENT		

Table 7.5: 10 most frequent entity labels for tokens in each different NE variant. Here, CD is short for CARDINAL.

Variant	NE only	NE POS	JOINT	HIGH	LOW	P'LOW	SUB	SUB L
TOP	66.19	66.02	**83.20**	70.18	79.87	74.79	82.63	80.87
BOTTOM	75.99	75.56	85.44	72.74	**85.66**	72.75	85.37	83.58
TOP_2	61.78	61.57	**78.37**	62.97	74.06	66.49	77.22	74.97
$BOTTOM_2$	66.20	65.44	**79.93**	64.84	75.97	67.69	78.50	76.32
TOP_BOT	63.73	63.19	**80.13**	64.87	76.36	68.03	78.40	76.49
STACK	60.13	59.99	**77.01**	61.42	72.23	64.43	75.63	73.29
COMPRESS	63.44	62.88	**79.80**	64.33	75.87	67.63	78.10	76.29

Table 7.6: Result of parsing models in NER; P'LOW stands for POSLOW variant; SUB L stands for SUB LAYER variant

We expect the NER models to outperform our parsing models on the TOP and BOTTOM variants, since we are comparing a state of the art NER system to a standard parser with no additional NER features. We also expect the NER models to be highly competitive on the TOP_2, $BOTTOM_2$ and TOP_BOTTOM variants, though the LIBSCHWA NER system has not been evaluated on a category set of this size before, so we predict that training time will be substantially slower for these models than for those trained on the TOP and BOTTOM variants.

We expect that training time will be a much greater concern with the STACK and COMPRESS variants. Due to the complexity with respect to tagset size in these variants, training time is a substantial concern. We therefore also train the C&C NER tagger (Curran and Clark, 2003b) (see Section 7.4.2) which uses a maximum entropy tagger, and is designed to train CCG supertagger models, which have a much larger category set, and sparser data, than standard NER.

7.3.9 Using Parsing Models for NER Variant Experiments

In order to compare the results of the LIBSCHWA NER system and the C&C output to our parsing models, we take the output of our parsing experiments from Chapter 6 and convert them into the output expected for each of the variants. That is, we reformat each output tree from the output of our parsing models, and output the resultant parse tree in each variant, which we then evaluate using the standard CoNLL script. The F_1-scores of each of these experiments are shown in Section 7.6.

We found that although the SUB and SUB LAYER parsing variants performed strongly on the combined parsing task evaluations, the JOINT variant outperformed it in all NER variants, indicating that the parser was robust to the larger node label space of the JOINT model, and that it learnt better representation of the structure of these entities with access to syntactic structure. The performance of models trained without syntactic information also adds further support to this. The comparatively low performance of the NE only and NE POS variants indicate that syntactic structure is useful in learning NER structure. Further, almost all of the models trained with variants which included syntactic structure also outperformed these NE only versions when evaluating only on BOTTOM, with the strongest achieving $10F_1$-score higher than these models, indicating that syntactic structure improves the learning of even the closest NE to the token.

We also see that the model trained on the HIGH variant does not perform well on any NE variant, indicating that although we found that syntax us useful for learning NE structure, it should be as close to the token as possible, and not act as an intervening node between the token and NE layer.

Interestingly, the model trained on the JOINT variant performed strongest across all variants with the exception of BOTTOM, for which the LOW parsing

variant outperformed it by a small amount. This indicates that the model trained on LOW had the best model for labels closest to the token. When we compare this to the model trained from the POSLOW variant, which has these NE labels interleaved with POS tags, we see a sharp decrease in performance, indicating that having the labels only immediately above the token (which also occurs in SUB and SUB LAYER) results in the best performance when evaluating only on the closest NE labels to each token, as occurs in BOTTOM.

We will use the parsing model trained on the JOINT variant for all future experiments other than BOTTOM, for which we will use LOW.

Our parsing models which were trained on NE only, or on only NE and POS tags were also strongly outperformed by other variants which indicates that syntactic information helps in learning NE representations. Interestingly, this difference is seen both in our more complex NER variants which combine multiple layers of NE as well as the flat NER variants of TOP and BOTTOM.

7.4 Experimental Setup

We compare the state of the art, fast NER with the parsing results from Chapter 6. The goal of this comparison is to identify which is the best way to learn representations of fine-grained nested NER.

7.4.1 LIBSCHWA NER

We use the LIBSCHWA NER system, a state of the art NER system which utilises document structure information provided by DOCREP (Dawborn and Curran, 2014; Dawborn, 2015). The system utilises a linear chain CRF backed by CRFsuite (Okazaki, 2007) and L-BFGS for numerical optimisation.

Preprocessing As a preprocessing step, LIBSCHWA NER attempts to perform truecasing (Lita et al., 2003) on sentences which appear in all-caps using capitalisation frequencies from both in- and out of document. All digits and ordinals are normalised to 9 and 9th respectively, which reduces the sparsity of numerical quantities.

Morphosyntactic features of NER system The system uses various morphosyntactic features including prefix (of length 2 to 5), suffix (of length 2 to 5), word shape, and boolean features indicating whether the word contains a digit, hyphen, uppercase letter, roman numeral, or whether it looks like an acronym. Capitalisation pattern for a window of 1 token around the current token, and another with a window of 2 tokens around the current token are also used. Brown cluster path (Ratinov and Roth, 2009), Clark cluster generated from the Reuters 1 corpus[1] and HLBL word embedding features (Mnih and Hinton, 2009) are also used.

Contextual features of NER system As a contextual feature, LIBSCHWA NER uses multi-word gazetteer matching using the gazetteers distributed with the Illinois tagger (Ratinov and Roth, 2009). It also uses extended prediction history (Ratinov and Roth, 2009) with memory restricted to the current document, rather than the previous 1000 tokens.

Document level features Block-ordered iteration enables the LIBSCHWA NER system to annotate sentences which occur in paragraphs before annotating tokens in headings or lists. This allows the system to make initial classification decisions with more context before classifying entities in headings, which are often hard to classify without first reading the document.

[1] http://www.cs.rhul.ac.uk/home/alexc/

7.4.1.1 Tuning LIBSCHWA NER

In our experiments we use sections 02 to 22 as training data, and sections 00 and 23 as test, utilising the same split as our parsing experiments. To establish the best configuration for the LIBSCHWA NER system, we evaluated the effect of different CRF smoothing parameters (see Table 7.7) on the BOTTOM variant, finding a small decrease in performance with a smoothing parameter of 0, but no significant difference between 0.2 and 1.0. A very small decrease was also seen between 1.0 and 2.0, alongside a large decrease in training time.

The BOTTOM and TOP variants have substantially fewer categories than other variants, on the order of 100, compared to 900 to 1700 (see Table 7.3), so have the least sparse label space, and train the fastest. We use BOTTOM to tune training parameters from a practical standpoint, given that other variants take upwards of two months to train, even using conservative feature sets and training parameters.

Smoothing	Precision	Recall	F_1-score	Accuracy	Training time
0.0	88.88	88.00	88.44	97.09	4 days, 7h 37m
0.2	91.20	89.86	90.53	97.52	3 days, 10h 48m
0.4	91.27	89.81	90.53	97.54	2 days, 14h 32m
0.6	91.40	89.82	90.60	97.55	2 days, 12h 58m
0.8	91.47	89.80	90.63	97.54	2 days, 9h 33m
1.0	91.49	89.70	90.59	97.54	2 days, 5h 21m
1.5	91.38	89.43	90.39	97.50	20h 37m
2.0	91.36	89.25	90.29	97.45	20h 48m

Table 7.7: Results of training with various CRF values.

We elect to use the BMEWO encoding, which was found to be the best on the standard NER task (Dawborn, 2015). We verify this setting by analysing

the different entity encodings during training (see Table 7.8), finding negligible difference between BIO1, BIO2 and BMEWO when training with the BOTTOM variant.

	Precision	Recall	F_1-score	Accuracy
BIO1	91.11	89.21	90.15	97.48
BIO2	91.45	89.77	90.60	97.55
BMEWO	91.27	89.81	90.53	97.54

Table 7.8: Results of varying the encoding during training.

7.4.2 C&C NER Tagger

We also use the C&C NER system (Curran and Clark, 2003b) as a comparison point. The system uses a maximum entropy tagger employing Gaussian smoothing, which allows a large number of sparse but informative features to be used. We use the default orthographic, contextual, in-document and personal name gazetteer features. The C&C NER system has not been actively developed for over 10 years, and it is not competitive with state of the art. It does, however, provide a powerful model, capable of learning from very sparse data. We therefore use it as both a comparison to LIBSCHWA NER and especially as a fallback for very sparse data variants.

7.4.3 Results of LIBSCHWA NER, C&C, Parsing

In Table 7.9, we compare the results of each variant trained using the LIBSCHWA NER system, the C&C NER tagger, and the JOINT or LOW Parsing models. As expected, we find that when evaluating on the two variants which use a small set of categories, TOP and BOTTOM, the LIBSCHWA NER DOCREP model outperforms both C&C and our parsing models. That is, the state of the

			LIBSCHWA NER			C&C			Parsing		
	#NE	# cats	*P*	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
TOP	109	100,523	85.7	84.6	85.1	73.8	68.1	70.8	84.8	81.7	83.2
BOTTOM	113	169,811	91.4	89.4	90.4	83.9	80.0	81.9	86.4†	85.0†	85.7†
TOP_2	906	155,752	85.8	83.8	84.8	65.5	61.8	63.6	79.8	77.0	78.4
$BOTTOM_2$	909	172,392	88.0	85.2	86.6	70.8	67.1	68.9	81.4	78.5	79.9
TOP_BOT	967	170,319	88.1	85.6	86.8	68.8	64.5	66.6	81.6	78.7	80.1
STACK	1738	172,840	$-^{*}$	$-^{*}$	$-^{*}$	63.8	60.2	62.0	78.5	75.6	77.0
COMPRESS	1314	169,787	$-^{*}$	$-^{*}$	$-^{*}$	70.5	66.7	68.6	81.3	78.4	79.8

Table 7.9: Comparison of NER results on different NE variants with CRF 1.5, using LIBSCHWA NER, C&C and our parsing models. *DOCREP STACK and COMPRESS did not finish within a reasonable timeframe using these settings, with estimated experiment length was over 40 days.†Note these were LOW parse variant, rather than JOINT, which is used for all other parsing numbers.

art LIBSCHWA NER model better learned the variants which involved on the order of 100 categories, and which did not have a structural component.

LIBSCHWA NER also outperformed both C&C and our Parsing models on the *combine two* variants: TOP_2, BOTTOM_2 and TOP_BOTTOM. This indicates that LIBSCHWA NER can still learn highly accurate models with particularly sparse labels (on the order of 900 categories). However, these models were slow to train, as discussed more in Table 7.10. The *combine two* variants TOP_2, BOTTOM_2 and TOP_BOTTOM proved substantially harder for the C&C models to learn, performing around 20F_1-score lower than the corresponding DOCREP models for the same tasks, - compared to around 9 and 15 F_1-score lower for the BOTTOM and TOP variants respectively.

As expected, the BOTTOM variant proved easier to learn than any other variant, due to the very high consistency of labels per token. Interestingly, while the Parsing model trained on the TOP variant did not outperform the corresponding LIBSCHWA NER model, the performance difference between the two (1.9F_1-score) is substantially lower than the performance difference for the BOTTOM variant (4.7F_1-score), indicating that the Parsing model learnt this harder task quite well.

Our initial experiments used the parameters that performed strongest on the standard NER task on CoNLL data, that is, a CRF smoothing parameter of 0.4, no minimum count for a feature to be included in training, and a maximum of 500 iterations. The BOTTOM_2 model achieved an F_1-score of 86.6, substantially outperforming the C&C F_1-score of 68.9 and our Parsing F_1-score of 79.9. However, it took 31 days, 12 hours and 52 minutes to train. The BOTTOM_2 model was still trained on data that was substantially less sparse than the STACK and COMPRESS variants, which took substantially longer to train.

The STACK and COMPRESS variants have an even larger and more sparse category space, and the training time of the LIBSCHWA NER models increases

further. Using those settings, the COMPRESS variant finished training after 64 days, 21 hours and 40 minutes, achieving 88.1 precision, 86.2 recall and an F_1-score of 87.1. This is significantly higher than the F_1-score of the parsing model (79.8) or of the C&C model (68.6), though its training was more than 100 and 1000 times faster respectively.

In another configuration which limited the number of iterations to 25, the STACK model trained in 8 days and 4 hours and 48 minutes, and achieved an F_1-score of 74.4, performing slightly behind the Parsing model which achieved 77.0 F_1-score in 12 hours and 39 minutes. Using the same configuration, the COMPRESS achieved an F_1-score of 79.2, again, slightly behind the Parsing F_1-score of 79.8. It trained in 6 days, 4 hours and 58 minutes.

Given an extremely large amount of training time, the LIBSCHWA NER models perform very strongly. In a different configuration, we ran the COMPRESS variant for 500 iterations. This ran for 64 days, 21 hours and 39 minutes, and achieved an F_1-score of 87.2, significantly outperforming all other results for the COMPRESS variant. It is clear that the LIBSCHWA NER model is highly accurate, and such high performance on a very complex task is promising, but such a long training time is not always practical; indeed a number of other experiments were abandoned due to machine failures or forced restarts.

Given the results of these experiments, it seems the best combination of variant and system to use depends highly on the specific task, and whether training time is a limiting factor.

7.4.4 Training Time Comparison of LIBSCHWA NER, Parsing

The performance of the LIBSCHWA NER models is very strong, but this comes with a substantially longer training time than the C&C and Parsing models, as shown in Table 7.10. It is clear that the C&C models are the fastest to train, with

even the most complex and sparse model finishing training in just over two hours, some 9 hours faster than any other LIBSCHWA NER or Parsing model.

The Parsing model is trained on the JOINT variant of NE Parsing, or the LOW variant for the BOTTOM NER variant. Those models took 12:39 and 11:03 hours to train, respectively, making them slower than the C&C models, but substantially faster than the LIBSCHWA NER models. These Parsing models do, however, have additional time for testing, with the annotation of sections 00 and 23 taking an additional 13 hours and 21 minutes, meaning that the overall time taken to train and test TOP and BOTTOM for the Parsing and LIBSCHWA NER models were comparable. This additional testing time is, however, linear with the amount of data to be tagged, and the increased time is dwarfed by the substantial increase in training time used by LIBSCHWA NER for the more sparse and complex models. Though the LIBSCHWA NER models do represent a gain in performance and accuracy, in many situations the training time would result in these models being impractical.

	LIBSCHWA NER		C&C		Parsing	
	F	Training	*F*	Training	*F*	Training
TOP	85.1	20h 37m	70.8	10m	83.2	12h 39m
BOTTOM	90.4	20h 37m	81.9	7m	85.7§	11h 03m
TOP_2	84.8	12d 22h 35m	63.6	1h 2m	78.4	12h 39m
$BOTTOM_2$	86.6	16d 12h 19m	68.9	1h 0m	79.9	12h 39m
TOP_BOT	86.8	19d 11h 8m	66.6	1h 24m	80.1	12h 39m
STACK	74.4*	8d 4h 48m	62.0	2h 3m	77.0	12h 39m
COMPRESS	79.2*	6d 4h 58m	68.6	1h 25m	79.8	12h 39m

Table 7.10: Comparison of F_1-score and training times for different NE variants with CRF 1.5, using LIBSCHWA NER, C&C and our Parsing models. These models were trained using multiple runs of LIBSCHWA NER, as outlined in subsection 7.4.6. * Note that when DOCREP STACK and DOCREP COMPRESS were restricted to 25 iterations, they achieved these F_1-score. § Note this result uses the LOW parse variant, rather than JOINT, which is used for all other parsing numbers.

7.4.5 Error analysis across models and variants

To get a sense of what types of errors the systems were making with each model and each variant, we conducted some manual error analysis. The five most frequent misclassifications for each system are shown in Table 7.11 for the TOP, BOTTOM, TOP_2, $BOTTOM_2$ and TOP_BOTTOM variants. We can see that across each variant, the LIBSCHWA NER models consistently overpredict CORP entities, particularly PRODUCT:OTHER (e.g. Comprehensive Test of Basic Skills) and NAME (e.g. Bozell from Bozell Inc.). We can see this more clearly in the TOP_2, $BOTTOM_2$ and TOP_BOTTOM variants, with NAMEs within CORP entities (i.e. NAME_CORP) being mislabelled as CORP frequently in each. The MEDIA CORP confusion in LIBSCHWA NER was boosted by CNN being misclassified.

The CARDINAL O confusion in LIBSCHWA NER is caused predominantly by several, few and hyphenated tokens (e.g. nine-member).

The overclassification of CORP appears to also happen in the Parsing models, but comparatively less frequently in the C&C models, which are instead spread more widely between categories. The Parsing models trained on each variant show similar errors to that of the LIBSCHWA NER models, but also appear to learn the GOVERNMENT CORP distinction less clearly than LIBSCHWA NER models, predicting entities such as the Department of Health and Human Services substantially in favour of CORP. This GOVERNMENT CORP error is seen in each variant.

The DATE O confusion across all three systems and all variants is primarily due to determiners and prepositions being included in the entity span, with tokens the, of, in, its, this, and over being frequently missed. The C&C TOP variant was particularly bad at determining this, with this and the each being incorrectly excluded from the DATE span more than 150 times.

Not shown in Table 7.11 are the confusion matrices for COMPRESS and STACK for the C&C and Parsing models. Other than having a substantially longer tail to the errors, they are similar to other variants, with NAME_CORP, CORP and DATE errors featuring prominently.

	LIBSCHWA NER			C&C			Parsing		
	# E	Gold	Predicted	# E	Gold	Predicted	# E	Gold	Predicted
TOP	106	DATE	O	1021	DATE	O	177	CORP	O
	96	PROD:OTH	CORP	485	CORP	PER	147	DATE	O
	80	MEDIA	CORP	445	RATE	O	115	PER	CORP
	63	CD	O	398	MONEY	PERCENT	101	PROD:OTH	CORP
	57	O	DATE	364	PER	CORP	97	GOV	CORP
BOT	134	NAME	CORP	600	DATE	O	210	NAME	CORP
	123	DATE	O	400	RATE	O	110	O	DATE
	74	PROD:OTH	CORP	395	O	REL	108	CORP	O
	67	QUAL	O	283	CORP	O	84	O	CORP
	63	MEDIA	CORP	250	CORP	NAME	84	CD	O
TOP2	115	DATE	O	487	DATE	O	144	NAME_CORP	CORP
	112	NAME_CORP	CORP	394	O	REL_DATE	133	CORP	O
	68	MEDIA	CORP	322	RATE	O	89	CORP	NAME_CORP
	65	PROD:OTH	CORP	315	NAME_CORP	NAME_PER	83	DATE	O
	61	CORP	O	297	CD_MONEY	CD_PERCENT	70	GOV	CORP
BOT2	136	NAME_CORP	CORP	487	DATE	O	137	NAME_CORP	CORP
	111	DATE	O	457	O	REL_DATE	133	CORP	O
	73	PROD:OTH	CORP	359	QUAL_CD	O	85	CORP	NAME_CORP
	70	MEDIA	CORP	322	RATE	O	83	DATE	O
	59	CORP	O	300	CD_PERCENT	CD_CD	70	GOV	CORP
TOPBOT	109	NAME_CORP	CORP	487	DATE	O	142	NAME_CORP	CORP
	107	DATE	O	454	O	REL_DATE	133	CORP	O
	70	MEDIA	CORP	381	CD_MONEY	CD_PERCENT	89	CORP	NAME_CORP
	66	PROD:OTH	CORP	322	RATE	O	83	DATE	O
	65	CORP	NAME_CORP	313	NAME_CORP	NAME_PER	70	GOV	CORP

Table 7.11: Comparison of frequent labelling errors between each variant and system. The five most frequent incorrect predictions for each variant (other than STACK and COMPRESS) are shown, along with the number of entities (# E) affected. PROD:OTH stands for PRODUCT:OTHER, GOV for GOVERNMENT, and CD for CARDINAL.

7.4.6 Sparse model vs. multiple CRF

We also compare the impact of sparse training data compared to running multiple LIBSCHWA NER models each trained for a distinct *level* of NER category labels. To do this, we modify the LIBSCHWA NER model to accept NE labels as a feature, and run two separate models sequentially. The first model is a standard model, trained on the BOTTOM variant to predict the lowest layer of NE labels. We then train a model to predict the additional layer of NEs, either the second bottom layer or the top layer, to match the BOTTOM_2 and TOP_BOTTOM variants, respectively. This model is trained on gold-standard NE tags, and tested using the NE labels from the output of the BOTTOM model. Ideally we would perform jackknifing during training, but the length of training time for each of the models that would be required for this was prohibitive.

The results of these multi-run LIBSCHWA stack experiments, and the comparable LIBSCHWA NER experiment, along with training times, are shown in Table 7.12. The performance of the multi-run LIBSCHWA stack experiments is promising. While the LIBSCHWA NER models still outperform these multi-run classifiers, they in turn outperform the Parsing and C&C models for the same tasks. The multi-run stack models are substantially faster to train than the LIBSCHWA NER models, with the total training time for both the BOTTOM and the required second stack model layer taking less than two days. While this is still slower than training times for both the C&C models and the Parsing models, it does offer a balance between training time and performance.

	LIBSCHWA NER				Multi-run LIBSCHWA stack			
	P	*R*	*F*	Train	*P*	*R*	*F*	Train
BOTTOM_2	88.0	85.2	86.6	16d 12m 19	86.3	82.2	84.2	1d 5h 32m
TOP_BOT	88.1	85.6	86.8	19d 11h 8m	85.0	81.2	82.1	1d 10h 11m

Table 7.12: Comparison of label sparsity vs. multiple taggers for BOTTOM_2 and TOP_BOTTOM variants.

7.5 NER Summary

In this chapter we have explored the question of how well existing NER systems can learn structured entities. We have presented a number of NER variants, and compare these across systems, and reinterpret our parsing results from Chapter 6 on these variants.

We have defined seven NER variants which capture differing extents of the nested structure of entities in the NNE corpus. Each variant yields a different number of categories, ranging from 109 to 1738, enabling downstream systems to control the degree of nested structure they make use of. The TOP variant is the closest to current general domain NER, and STACK captures the full nested structure of each entity.

We have evaluated these projections for two NER systems and reevaluated parser results for direct comparison. We found that, in general, the performance of all evaluated systems degraded as the degree of nesting represented in each variant increased. We also found that the TOP_2 variant was particularly hard to learn, and suggested that this is in part due to the comparative ease of predicting the BOTTOM layer (a component of all non-TOP variants), because of the high level of token consistency which is a result of nested named entities.

In this chapter, we have evaluated three different alternatives for NER. When training the state of the art LIBSCHWA NER system, we achieved the

highest performance, but also found that these models were the slowest to train, especially on more complex variants. This potentially precludes application downstream. In cases where only a few layers of nesting are required, a multi-run setup provides a good compromise between performance and training time, but would be increasingly impractical for capturing the extensive layers of nesting in the NNE corpus.

The other two systems discussed in this chapter (C&C and Parsing) represent further options. We find that the C&C NER system is the fastest of systems we evaluated, and had fair performance. The performance of the parser, in both training time and accuracy, was encouraging. Specifically, the Parsing model for the TOP variant performs only 1.9% F_1-score below LIBSCHWA NER and trains in just half the time. Furthermore, the parser performance is strong despite having no NER specific features. We therefore see a move towards a parsing framework for NER as promising, and that the introduction of NER specific features has strong potential to substantially boost performance.

8 Conclusion

> One never notices what has been done; one can only see what remains to be done.
>
> Marie Curie

In this thesis, we have addressed the two key shortcomings of the current NER task: coarse granularity and non-structured entities. In particular, we have addressed the task of identifying and classifying structured, fine-grained named entities, numerical and temporal expressions. In so doing, we have presented a thorough examination of nested named entities.

The core contribution of this work is a corpus of nested named entities that brings the same level of detailed semantic analysis to the structure of named entities in the Penn Treebank, that has previously only been available for syntactic analysis. We have presented the first results exploring how well existing parsers and NER systems perform on this complex, nested named entity corpus. The promising results in both parsing and NER indicate that this corpus is feasible to learn, but indicate that substantial progress will be made by combining the best of both approaches in the future.

8.1 Future Work

We have presented the first large-scale, fine-grained, nested named entity corpus of English newswire text, annotating the full Wall Street Journal section of the

Penn Treebank. Our results, both for parsing the combined syntactic and nested named entity structure, and for the NER task on flat projections of the NNE corpus, have set the benchmark on how much can be achieved with existing parsers and NER systems. We have demonstrated that these structures can be learnt reasonably well with supervised methods, even without modifying existing parsing and NER systems. The key limitation of current NER systems is training time, and this could be addressed in a number of ways. However, there are many avenues for future work beyond NER itself that are also now possible.

8.1.1 Extend annotation for use in other corpora

OntoNotes (Hovy et al., 2006; Weischedel et al., 2010, 2013) is a recent annotation effort which combines multiple layers of annotation, including syntactic annotations, PropBank and NomBank. While OntoNotes does include named entity information over a subset of its articles, it does not annotate nested structure.

Since 590 articles used in OntoNotes are from the Wall Street Journal section of the Penn Treebank, we can use the same sections of our NNE corpus directly and add structured named entity information to those sentences. The obvious next step would be to extend our NE annotations to a larger set of OntoNotes articles, using the detailed annotation guidelines produced as part of this thesis.

8.1.2 Modify systems for better structured NER learning

Our preliminary results of training a parser to predict nested named entities have proven successful even without modifications to the parser. However, we should expect substantial performance improvements for approaches that model and/or represent the specific properties of this task.

A starting point would be to take the existing work of Finkel and Manning (2009c), whose joint parser and named entity recognition system achieved good performance on nested named entities in biomedical and newspaper text, using NER sections of AnCora Taulé et al. (2008) in Spanish and Catalan. Another modification of interest would be adapting a parser to take in both POS tags and NE suggestions, using an NER model trained from the BOTTOM or STACK variant. These NE labels could act in a similar way to supertags, adding information as to how the token will be used in the larger context.

The other logical extension to this work is to modify NER models to make them better suited to learning nested named entities. The strong performance of parsers that do not have any NER specific features indicates that we should be moving away from flat NER models. The Berkeley parser (Petrov et al., 2006), with no NER modifications and no attempt to account for the substantial splitting of the NP category, achieved an F_1-score of 83.2, outperformed by only 1.9 F_1-score by the state of the art LIBSCHWA NER system, which achieved 85.1 on the TOP variant, the closest to the more standard 4 category NER task. This strongly suggests that parsing these nested NER structures is worth further research. This is likely to further inflate training time so simultaneous improvements to engineering are also needed, particularly optimising the algorithm for many categories.

8.1.3 Extend the new NNE resource onto other resources

Linguistically rich formalisms

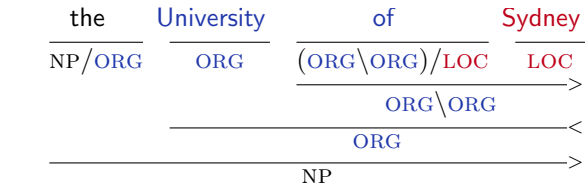
Another direction now made possible is the application of our NNE resource onto linguistically rich formalisms which are derived from the PTB, such as CCG, HPSG, or LFG. Many of these formalisms provide a theory of the relationship between syntax and semantics. With the NNE corpus, we are providing

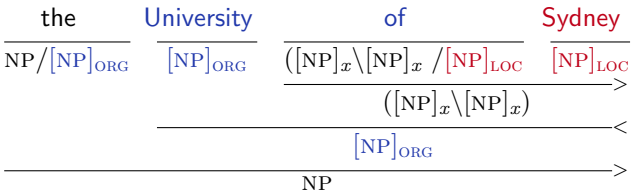
both more syntactic and semantic information about NEs than was previously available. We hope this flows through to those formalism specific corpora.

Just like the work in Chapter 5, where we discuss different ways of merging NEs and the syntactic information, there are also a range of ways that these formalisms could encode what we have identified as semantic grammar rules.

We have a particular interest in how NNE categories would work in CCG, either directly subdividing the atomic N or NP categories, as features on N and NP categories, or as a form of hat categories, as introduced in Honnibal and Curran (2009).

In some sense, the STACK projection of NNE variants is already quite similar to the supertags used in CCG and other lexicalised grammar formalisms, because the stack describes all of the nested entities a token is expected to fall within – and the result is a similar number of stack categories and CCG supertags.





In this example, University of behaves as a function that collects LOCs, or NPs with LOC as an annotation, and turns them into ORGs, and we can couple the syntactic and semantic behaviour of University of. In the second example, the CCG category for of further abstracts its semantics, taking any entity type and returning a larger entity with a locative modifier.

Other aligned resources: PropBank and NomBank

The NNE annotations are also valuable additions to other linguistic resources built on the same text as the PTB, including NomBank and PropBank, and can be used in downstream applications such as coreference resolution and relation extraction. Many entities, including those nested in other entities, can form part of a relation. Until now, we haven't been able to model the relationship of the entities within nested entity structure. We will now be better able to identify both that [William Boeing]$_{\text{PER}}$ is the likely founder of the [Boeing Company]$_{\text{ORG}}$ and that both [Bill Gates]$_{\text{PER}}$ and [Melinda Gates]$_{\text{PER}}$ are founders of the Bill and Melinda Gates Foundation.

We are also now able to analyse metonymy coercions when they exist. Until now, when using semantic compatibility tests, one has been forced to assume only the information that we have access to from the top layer of NE structure. For example, consider the arguments of the ditransitive verb *buy*. If we know, from PropBank, that the first argument should be an agent, but find an example where that argument is marked as a LOC, this is likely to be problematic. If, on the other hand, we find not a LOC, but a CITY embedded within a SPORTS-TEAM as the agent, perhaps [[Toronto]$_{\text{CITY}}$]$_{\text{TEAM}}$ bought star pitcher. . . , we have satisfied the semantic compatibility constraints.

8.1.4 Analysis in a practical task, and error analysis

A further avenue of research has to do with the number of categories in our entity scheme. Our motivation was to produce all annotations, not all of which would be required for all tasks. It is not clear what level of detail of each of the broader NER categories are needed for real-world applications is needed. On a similar track, it would be interesting to see the effects of downmapping some of

our 118 entities to the broader categories, and whether this would substantially impact performance.

More detailed error analysis of the NER task would also be a productive avenue for further work. Notably, facilitating the more linguistically motivated error analysis that Kummerfeld et al. (2012) allow us to perform for the task of syntactic parsing would be of use for better analysis at how systems compare, and what the particular challenges of fine-grained, nested named entities are.

8.1.5 Corpus improvements

Over the course of the annotation of nested named entities and their subsequent merging with the syntactic structures of the PTB, we identified some analyses that proved problematic. The Penn Treebank has now had three versions, with additional modifications in the form of NP structure (Vadas and Curran, 2007), yet some open questions remain.

The analysis of prepositional phrases is one such area. We have taken a fairly lean approach, changing only the analysis of PPs that interact with NEs, but a more consistent approach would be far more beneficial. The analysis of determiners and their relationship with NEs has also caused issues in our work. A large component of errors uncovered in our error analysis for the NER task were caused by the incorrect inclusion or exclusion of determiners.

In this work, we have taken a conservative approach to the distinction between proper and common nouns. The analysis of common nouns and NPs which contain NEs is a promising avenue for further corpus extension; for example, [U.S.]$_{\text{COUNTRY}}$ allies could qualify as a GRP:LOC.

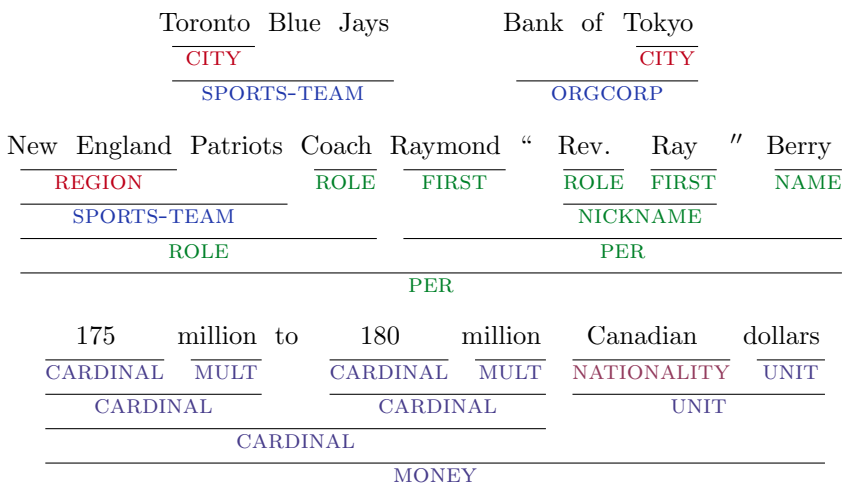
Despite taking great care to maintain consistency across annotators and the use of a sophisticated annotation tool that showed likely structural analyses from previously annotated entities, there are still areas of inconsistencies in the corpus that another annotation pass would substantially improve, both in

individual annotations and in granularity distinctions of the annotation scheme. Ideally, we would like to see a mechanism whereby if other researchers come across an annotation error, they can suggest corrections. In this way, we can iteratively improve the corpus. The usefulness of this is not limited to our NNE corpus, but would also be a valuable tool in reducing errors in all corpora.

8.2 Conclusion

This thesis contributes a substantial body of work on fine-grained nested named entities, the core contribution of which is a corpus that brings detailed semantic analysis to the structure of named entities. NNE is the first corpus of nested named entity structure in English newswire text. It comprises 50,000 sentences annotated with over 279,000 fine-grained, structured entity annotations over the Wall Street Journal section of the Penn Treebank. Additionally, we have addressed the question of the feasibility of this more complex task, with strong evidence indicating that it is.

In Chapter 3 we introduced a robust set of annotation principles governing the annotation task. This includes the addition of substructures to avoid spurious ambiguity on a per-token level ([Toronto]$_{\text{CITY}}$ should still be labelled a CITY in the [[Toronto]$_{\text{CITY}}$ Blue Jays]$_{\text{TEAM}}$), unary stacking principles to capture metonymy (The [[White House]$_{\text{BUILDING}}$]$_{\text{GOV}}$ said . . .), the embedding of structural sub elements such as numerical MULT tokens combining with CARDINAL tokens to form larger CARDINAL spans ([[180]$_{\text{CD}}$ [million]$_{\text{MULT}}$]$_{\text{CD}}$), and the addition of new categories to improve difficult annotation categorisation decisions (the name of a hotel is easy to identify as a HOTEL, but harder to classify into either CORP or FACILITY, as it displays both organisational and locational qualities). Other principles govern annotation boundaries and nesting structures, and could be applied to other annotation tasks.



In Chapter 3 we summarised the detailed nested named entity annotation scheme developed, including 118 fine-grained entity types, arranged in a hierarchy of 10 broad entity types: PER, LOC, ORG, FACILITY, NORP, EVENT, WOA, MISC, TIMEX and NUMEX These guidelines have high coverage of edge cases, and were continuously updated during the annotation process to reflect additional edge case decisions, resulting in a valuable annotation reference. This level of detail in annotation guidelines is important both for the quality of annotation and its reproducibility, in the case of extending the corpus, or annotating other corpora with the same scheme. These extensive annotation guidelines bring fine-grained, structural named entities to the same level of detail as the annotation guidelines for the Penn Treebank.

As described in Chapter 4, we present an annotation tool that displays previous annotation decisions within this document, and in the entire corpus, to the annotator, and allows annotators to make decisions on a per-document and corpus level. In the tool, annotators are shown their previous annotation decisions for entities and sub-structures that match a currently selected span, prompting the annotator and allowing them to easily apply consistent annotations. This aided in the creation of a highly consistent corpus. The annotation tool will be open sourced for use in the research community.

Our annotation principles, highly detailed annotation guidelines and innovative annotation tool allowed for a highly consistent annotation process. The nested named entity corpus has high inter-annotator agreement, achieving a Fleiss' kappa of 0.834. This high level of agreement on such a complex task is encouraging in that it suggests the task is feasible.

We present an empirical analysis of named entities in the WSJ PTB corpus, finding a high number of structural entities with multiple layers of annotation. We find that more than half of all entities form a structural part of an entity, with entities having up to 6 layers of nesting. This means that substantially more layers exist that are not possible to capture in a flat annotation scheme.

We find consistency in structural components that form these nested structures. For example, MONEY is frequently constructed by an adjacent UNIT and CARDINAL span. We find that 47 rules make up more than 80% of all nesting rules in the corpus. This further suggests the feasibility of learning the task.

We align this NNE corpus to the syntactic constituents of the WSJ Penn Treebank, which allows this resource to be used with all other resources which build from it, including PropBank, NomBank and syntactic corpora in other grammatical formalisms such a CCGbank (Section 2.2.2), enabling rich semantic modelling across annotation layers.

We find the majority of discrepancies between NNE and PTB were caused by tokenisation issues caused by full stops, and prepositional phrases, the PTB analysis for which is often incorrect. We use 4 rules which modify the bounds of our NNE annotations, 6 rules which modify PTB tree structure, and a number of specific, per-sentence fixes to ensure consistency with the PTB.

In Chapters 6 and 7, we explore the feasibility of the task empirically, by presenting present a novel analysis of combining syntactic and named entity information into a consistent structure, and present an analysis of the effect of different merging algorithms on both parsing and named entity recognition.

We present the first results of recovering nested named entity structure in English newswire text. We analyse the performance of models which have learnt combined syntax and NER, evaluating their performance on both the combined task and each individual component: syntactic parsing and structured named entity recognition.

We experiment with a range of methods for merging syntactic and named entity annotations into formats suitable for parsers and NER systems. We find, while the variants aren't directly comparable, there appear to be highly substantial differences in how well models can be learnt.

In particular, in Chapter 6, we use the Berkeley parser to learn combined constituent structure and named entity structure. We found that when we learnt a combined NE and syntactic structure, using the LOW variant, we achieve an F_1-score of 89.96 when evaluating on syntactic component alone (compared to 90.12 when learning the syntactic information only). We achieve an F_1-score of 88.17 when using the SUB variant learning combined NE and syntactic structure and evaluating on NE structure alone. Error analysis on these models finds the LOW model indeed minimises the total number of errors and nodes affected by errors, indicating that it would be better for certain applications requiring high precision.

We have experimented with two different NER systems, and in Chapter 7, present the first results semi-structured named entity recognition, by projecting the full structure of nested named entities into a flat label. We compare the results of a state of the art NER system to the performance of both a highly scalable NER system, and the parsing models we developed in Chapter 6 on these semi-structural projections.

The LIBSCHWA NER model performs strongly on the fine-grained NE corpus, but data sparsity substantially affects training time. LIBSCHWA NER models learn variants which capture partial structure with high accuracy, but

take several weeks, and are unable to learn reliable representations of the full structure in a reasonable time frame. The Parsing models perform strongly, trailing the LIBSCHWA NER performance by 2 to 5% F_1-score in the least sparse variants, and achieving an F_1-score of around 77.0 and 79.8 for the STACK and COMPRESS respectively.

8.3 Summary

The primary contribution of this thesis is the *Nested Named Entity* corpus – the principled annotation of fine-grained, nested named entities over the WSJ portion of the Penn Treebank, its merged version with the PTB syntactic analyses, and detailed annotation guidelines that documents the annotation scheme.

With this NNE corpus, we have eliminated the spurious ambiguity of linear NER and found a way to represent metonymous mentions consistently. Using this corpus, we have answered the question of how well existing parsers and NER systems perform with more complex structural named entities, and established that the more complex task is feasible to learn. We have examined performance on the straight tasks of NER and parsing, and the combination of the two, but the use and applicability of this NNE corpus is far more extensive, and we look forward to its utilisation in many future NLP tasks.

Bibliography

Beatrice Alex, Barry Haddow, and Claire Grover. 2007. Recognising nested named entities in biomedical text. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, pages 65–72.

Olga Babko-Malaya, Ann Bies, Ann Taylor, Szuting Yi, Martha Palmer, Mitch Marcus, Seth Kulick, and Libin Shen. 2004. Issues in synchronizing the English treebank and propbank. In *HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation*, pages 70–77.

Christian Bering, Witold Drożdżyński, Gregor Erbach, Clara Guasch, Petr Homola, Sabine Lehmann, Hong Li, Hans-Ulrich Krieger, Jakub Piskorski, Ulrich Schäfer, Atsuko Shimada, Melanie Siegel, Feiyu Xu, and Dorothee Ziegler-Eisele. 2003. Corpora and evaluation tools for multilingual named entity grammar development. *Proceedings of Multilingual Corpora Workshop at Corpus Linguistics 2003*, pages 42–52.

Ann Bies, Mark Ferguson, Karen Katz, and Robert MacIntyre. 1995. Bracketing Guidelines for Treebank II Style Penn Treebank Project. Technical report.

Daniel M. Bikel. 2004. *On the Parameter Space of Generative Lexicalized Statistical Parsing Models*. Ph.D. thesis, University of Pennsylvania, Philadelphia, PA, USA.

Daniel M Bikel, Richard L Schwartz, and Ralph M Weischedel. 1999. An Algorithm that Learns What's in a Name. *Machine Learning*, 34(1-3):211–231.

Ezra Black, Steven Abney, Dan Flickinger, Claudia Gdaniec, Ralph Grishman, Philip Harrison, Donald Hindle, Robert Ingria, Frederick Jelinek, Judith Klavans, Mark Liberman, Mitch Marcus, Salim Roukos, Beatrice Santorini, and Tomek Strzalkowski. 1991. A Procedure for quantitatively comparing the syntactic coverage of English grammars. In *Proceedings of the February 1991 DARPA Speech and Natural Language Workshop*, pages 306–311.

Ezra Black, Fred Jelinek, John Lafferty, David M Magerman, Robert Mercer, and Salim Roukos. 1992. Towards history-based grammars: Using richer models for probabilistic parsing. In *Proceedings of the workshop on Speech and Natural Language*, pages 134–139.

Oriol Borrega, Mariona Taulé, and M. Antònia Martı. 2007. What Do We Mean When We Speak About Named Entities. In *Proceedings of the 4th Corpus Linguistics Conference*.

Ada Brunstein. 2002. Annotation Guidelines for Answer Types. LDC2005T33, Linguistic Data Consortium, Philadelphia.

Kate Byrne. 2007. Nested Named Entity Recognition in Historical Archive Text. In *Proceedings of the First IEEE International Conference on Semantic Computing (ICSC 2007)*, pages 589–596.

Aoife Cahill, Mairéad McCarthy, Josef van Genabith, and Andy Way. 2002. Automatic Annotation of the Penn-Treebank with LFG F-Structure Information. In *Proceedings of the Third International Conference on Language Resources and Evaluation Workshop on Linguistic Knowledge Acquisition and Representation - Bootstrapping Annotated Language Data*, pages 8–15.

Xavier Carreras, Lluís Márquez, and Lluís Padró. 2003. Learning a perceptron-based named entity chunker via online recognition feedback. In *Proceedings*

of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pages 156–159.

Eugene Charniak. 1997. Statistical parsing with a context-free grammar and word statistics. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence (AAAI-97)*, pages 598–603.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-00)*, pages 132–139.

Hai Leong Chieu and Hwee Tou Ng. 2003. Named entity recognition with a maximum entropy approach. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 160–163.

Nancy Chinchor. 1998. Overview of MUC-7. In *Proceedings of the 7th Message Understanding Conference*.

Laura Chiticariu, Yunyao Li, and Frederick R. Reiss. 2013. Rule-based information extraction is dead! long live rule-based information extraction systems! In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 827–832.

Michael Collins. 1997. Three Generative, Lexicalised Models for Statistical Parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 16–23.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania, Philadelphia, PA, USA.

Michael Collins. 2000. Discriminative Reranking for Natural Language Parsing. In *Proceedings of the 17th International Conference on Machine Learning (ICML-00)*, pages 175–182.

Michael John Collins. 1996. A New Statistical Parser Based on Bigram Lexical Dependencies. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 184–191.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

James R Curran and Stephen Clark. 2003a. Investigating GIS and smoothing for maximum entropy taggers. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 91–98.

James R. Curran and Stephen Clark. 2003b. Language independent NER using a maximum entropy tagger. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 164–167.

James R. Curran and Stephen Clark. 2003c. Language independent NER using a maximum entropy tagger. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 164–167.

Robert Dale and Paweł Mazur. 2007. Handling conjunctions in named entities. In *Computational linguistics and intelligent text processing*, pages 131–142. Springer.

Tim Dawborn. 2015. *DOCREP: Document Representation for Natural Language Processing*. Ph.D. thesis, University of Sydney, Sydney, Australia.

Tim Dawborn and James R. Curran. 2014. docrep: A lightweight and efficient document representation framework. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 762–771.

George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie Strassel, and Ralph M Weischedel. 2004. The Automatic Content Extraction (ACE) Program-Tasks, Data, and Evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation*, pages 837–840.

Joe Ellis and Jeremy Getman. 2015. TAC KBP 2014 - Entity Discovery and Linking Query Development Guidelines. Linguistic Data Consortium. Version 1.5, 2015.8.1, http://www.nist.gov/tac/2014/KBP/ColdStart/guidelines/TAC_KBP_2014_EDL_Query_Development_Guidelines_V1.5.pdf.

Joe Ellis, Jeremy Getman, and Stephanie Strassel. 2014. Overview of Linguistic Resource for the TAC KBP 2014 Evaluations: Planning, Execution, and Results. In *Proceedings of the Text Analysis Conference (TAC2014)*.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. 2005. Unsupervised named-entity extraction from the Web: An experimental study. *Artificial Intelligence*, 165(1):91–134.

Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 363–370.

Jenny Rose Finkel, Alex Kleeman, and Christopher D. Manning. 2008. Efficient, Feature-based, Conditional Random Field Parsing. In *Proceedings of ACL-08: HLT*, pages 959–967.

Jenny Rose Finkel and Christopher D. Manning. 2009a. Joint Parsing and Named Entity Recognition. In *Proceedings of Human Language Technologies:*

The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 326–334.

Jenny Rose Finkel and Christopher D. Manning. 2009b. Joint parsing and named entity recognition. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 326–334.

Jenny Rose Finkel and Christopher D. Manning. 2009c. Nested named entity recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 141–150.

Jenny Rose Finkel and Christopher D. Manning. 2010. Hierarchical Joint Learning: Improving Joint Parsing and Named Entity Recognition with Non-Jointly Labeled Data. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 720–728.

Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. 2003. Named Entity Recognition through Classifier Combination. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 168–171.

Yoav Freund and Robert E Schapire. 1999. Large margin classification using the perceptron algorithm. *Machine learning*, 37(3):277–296.

Chunyuan Fu and Guohong Fu. 2012. A dual-layer CRFs based method for chinese nested named entity recognition. In *Proceedings of the 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 2546–2550.

Guohong Fu and Kang-Kwong Luke. 2005. Chinese named entity recognition using lexicalized HMMs. *ACM SIGKDD Explorations Newsletter*, 7(1):19–25.

William A. Gale, Kenneth W. Church, and David Yarowsky. 1992. One sense per discourse. In *Proceedings of the Workshop on Speech and Natural Language*, pages 233–237.

Olivier Galibert, Sophie Rosset, Cyril Grouin, Pierre Zweigenbaum, and Ludovic Quintard. 2011. Structured and Extended Named Entity Evaluation in Automatic Speech Transcriptions. In *IJCNLP*, pages 518–526.

Sylvain Galliano, Edouard Geoffrois, Guillaume Gravier, Jean-François Bonastre, Djamel Mostefa, and Khalid Choukri. 2006. Corpus description of the ester evaluation campaign for the rich transcription of french broadcast news. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, pages 315–320.

Ralph Grishman, Catherine Macleod, and John Sterling. 1992. Evaluating Parsing Strategies Using Standardized Parse Files. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 156–161.

Cyril Grouin, Sophie Rosset, Pierre Zweigenbaum, Karën Fort, Olivier Galibert, and Ludovic Quintard. 2011. Proposal for an extension of traditional named entities: From guidelines to evaluation, an overview. In *Proceedings of the 5th Linguistic Annotation Workshop*, pages 92–100.

Baohua Gu. 2006. Recognizing nested named entities in genia corpus. In *Proceedings of the HLT-NAACL BioNLP Workshop on Linking Natural Language and Biology*, pages 112–113.

Taiichi Hashimoto, Takashi Inui, and Koji Murakami. 2008. Constructing Extended Named Entity Annotated Corpora. In *IPSJ SIG Notes 2008 (In Japanese)*, pages 113–120.

Julia Hockenmaier. 2003. *Data and Models for Statistical Parsing with Combinatory Categorial Grammar*. Ph.D. thesis, University of Edinburgh, Edinburgh, UK.

Julia Hockenmaier and Mark Steedman. 2007. CCGbank: a corpus of CCG derivations and dependency structures extracted from the Penn Treebank. *Computational Linguistics*, 33(3):355–396.

Matthew Honnibal and James R. Curran. 2009. Fully Lexicalising CCGbank with Hat Categories. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1212–1221.

Matthew Honnibal, James R. Curran, and Johan Bos. 2010. Rebanking CCGbank for Improved NP Interpretation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 207–215.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60.

Frederick Jelinek, John Lafferty, David Magerman, Robert Mercer, Adwait Ratnaparkhi, and Salim Roukos. 1994. Decision tree parsing using a hidden derivation model. In *Proceedings of the workshop on Human Language Technology*, pages 272–277.

Heng Ji, HT Dang, J Nothman, and B Hachey. 2014. Overview of TAC KBP 2014 entity discovery and linking tasks. In *Proceedings of the Text Analysis Conference (TAC2014)*.

Aravind Joshi and Yves Schabes. 1992. Tree adjoining grammars and lexicalized grammars. In *Tree Automata and Languages*, pages 409–432.

Ronald Kaplan and Joan Bresnan. 1982. *Lexical-Functional Grammar: A formal system for grammatical representation*. MIT Press, Cambridge, MA, USA.

Tadao Kasami. 1965. An efficient recognition and syntax analysis algorithm for context-free languages. Technical Report AFCRL-65-758, Bedford, MA, USA.

J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. 2003. GENIA corpus – a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl 1):i180–i182.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 1989–1993.

Dan Klein and Christopher D. Manning. 2003. Accurate Unlexicalized Parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430.

Dan Klein, Joseph Smarr, Huy Nguyen, and Christopher D. Manning. 2003. Named entity recognition with character-level models. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 180–183.

George R Krupka and Kevin Hausman. 1998. IsoQuest, Inc.: Description of the NetOwl^{TM} Extractor System as Used for MUC-7.

Seth Kulick, Ann Bies, Mark Liberman, Mark Mandel, Ryan McDonald, Martha Palmer, Andrew Schein, Lyle Ungar, Scott Winters, and Pete White. 2004. Integrated Annotation for Biomedical Information Extraction. In *HLT-NAACL 2004 Workshop: BioLINK 2004, Linking Biological Literature, Ontologies and Databases*, pages 61–68.

Jonathan K. Kummerfeld, David Hall, James R. Curran, and Dan Klein. 2012. Parser showdown at the wall street corral: An empirical investigation of error types in parser output. In *Proceedings of EMNLP*, pages 1048–1059.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33:159–174.

Xin Li and Dan Roth. 2002. Learning Question Classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1*, COLING '02, pages 1–7.

Dekang Lin and Xiaoyun Wu. 2009. Phrase Clustering for Discriminative Learning. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1030–1038.

Xiao Ling and Daniel S. Weld. 2012. Fine-grained entity recognition. In *In Proc. of the 26th AAAI Conference on Artificial Intelligence*.

Lucian Vlad Lita, Abe Ittycheriah, Salim Roukos, and Nanda Kambhatla. 2003. tRuEcasIng. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 152–159.

Xiaohua Liu, Shaodian Zhang, Furu Wei, and Ming Zhou. 2011. Recognizing Named Entities in Tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 359–367.

David Magerman. 1994. *Natural Language Parsing as Statistical Pattern Recognition*. Ph.D. thesis, University of Pennsylvania, Philadelphia, PA, USA.

Christopher Manning. 2006. Doing named entity recognition? Don't optimize for F_1. In *NLPers Blog*, 25 August. http://nlpers.blogspot.com.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn

Treebank: annotating predicate argument structure. In *Proceedings of the Workshop on Human Language Technology (HLT-94)*, pages 114–119.

Mitchell Marcus and Beatrice Santorini. 1991. Building very large natural language corpora: the Penn Treebank. *Submitted manuscript*.

Mitchell Marcus, Beatrice Santorini, and Mary Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Lluís Màrquez, Lluís Padró, Mihai Surdeanu, and Luis Villarejo. 2007. Upc: Experiments with joint learning within semeval task 9. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 426–429.

Lluís Màrquez, Luis Villarejo, MA Martí, and Mariona Taulé. 2007. Semeval-2007 task 09: Multilevel semantic annotation of Catalan and Spanish. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 42–47.

Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2005. Probabilistic CFG with Latent Annotations. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 75–82.

Andrew McCallum and Wei Li. 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 at HLT-NAACL 2003*, pages 188–191.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Reranking and Self-Training for Parser Adaptation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 337–344.

David D McDonald. 1996. Internal and external evidence in the identification and semantic categorization of proper names. In *Proceedings of the Workshop on Acquisition of Lexical Knowledge from Text*, pages 32–43.

A. Meyers, R. Reeves, C. Macleod, R. Szekely, V. Zielinska, B. Young, and R. Grishman. 2004. The NomBank Project: An Interim Report. In *HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation*, pages 24–31.

Andrei Mikheev, Marc Moens, and Claire Grover. 1999. Named Entity recognition without gazetteers. In *Proceedings of the ninth conference of European chapter of the Association for Computational Linguistics*, pages 1–8.

Yusuke Miyao, Takashi Ninomiya, and Jun'ichi Tsujii. 2004. Corpus-oriented grammar development for acquiring a Head-driven Phrase Structure Grammar from the Penn Treebank. In *Proceedings of the First International Joint Conference on Natural Language Processing (IJCNLP-04)*, pages 684–693.

Andriy Mnih and Geoffrey E Hinton. 2009. A scalable hierarchical distributed language model. In *Advances in neural information processing systems*, pages 1081–1088.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30:3–26.

David Nadeau, Peter D. Turney, and Stan Matwin. 2006. Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. In *Proceedings of the 19th International Conference on Advances in Artificial Intelligence: Canadian Society for Computational Studies of Intelligence*, pages 266–277.

NIST-ACE. 2008. Automatic Content Extraction 2008 Evaluation Plan (ACE08). NIST.

Jorge Nocedal and Stephen J. Wright. 1999. *Numerical Optimization*. Springer.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R Curran. 2013. Learning multilingual named entity recognition from Wikipedia. *Artificial Intelligence*, 194:151–175.

Naoaki Okazaki. 2007. CRFsuite: a fast implementation of Conditional Random Fields (CRFs). http://www.chokkan.org/software/crfsuite/.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon infused phrase embeddings for named entity resolution. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 78–86.

Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi. 2004. WordNet::Similarity - Measuring the Relatedness of Concepts. In *HLT-NAACL 2004: Demonstration Papers*, pages 38–41.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning Accurate, Compact, and Interpretable Tree Annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 433–440.

Andrew Philpot, Eduard Hovy, and Patrick Pantel. 2005. The omega ontology. In *Proceedings, IJCNLP workshop on Ontologies and Lexical Resources (OntoLex-05)*.

Carl Pollard and Ivan Sag. 1994. *Head Driven Phrase Structure Grammar*. CSLI/Chicago University Press, Chicago, IL, USA.

Adam Przepiórkowski, Rafal L Górski, Marek Lazinski, and Piotr Pezik. 2010. Recent Developments in the National Corpus of Polish. In *Proceedings*

of the Seventh International Conference on Language Resources and Evaluation (LREC'10).

James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro, et al. 2003. The timebank corpus. *Corpus linguistics*, 40:647–656.

Lance A. Ramshaw and Mitchell Marcus. 1995. Text Chunking Using Transformation-Based Learning. In *Proceedings of the Third ACL Workshop on Very Large Corpora*, pages 82–94.

Lev Ratinov and Dan Roth. 2009. Design Challenges and Misconceptions in Named Entity Recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, pages 147–155.

Alexander E. Richman and Patrick Schone. 2008. Mining wiki resources for multilingual named entity recognition. In *Proceedings of ACL-08: HLT*, pages 1–9.

Brian Roark and Michiel Bacchiani. 2003. Supervised and unsupervised PCFG adaptation to novel domains. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 204–207.

Tony Rose, Mark Stevenson, and Miles Whitehead. 2002. The Reuters Corpus Volume 1-from Yesterday's News to Tomorrow's Language Resources. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 827–832.

M Rössler. 2004. Corpus-based learning of lexical resources for German named entity recognition. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation*, pages 1455–1458.

Erik F Sang and Jorn Veenstra. 1999. Representing text chunks. In *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*, pages 173–179.

Agata Savary and Jakub Piskorski. 2010. Lexicons and grammars for named entity annotation in the National corpus of Polish. *Intelligent Information Systems*, pages 141–154.

Agata Savary and Jakub Piskorski. 2011. Language resources for named entity annotation in the National Corpus of Polish. *Control and Cybernetics*, 40:361–391.

Satoshi Sekine. 2008. Extended Named Entity Ontology with Attribute Information. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 52–57.

Satoshi Sekine and Chikashi Nobata. 2004. Definition, Dictionaries and Tagger for Extended Named Entity Hierarchy. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation*.

Satoshi Sekine and Elisabete Ranchhod. 2009. *Named entities: recognition, classification and use*, volume 19. John Benjamins Publishing.

Satoshi Sekine, Kiyoshi Sudo, and Chikashi Nobata. 2002. Extended Named Entity Hierarchy. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 1818–1824.

Mark Steedman. 1996. *Surface Structure and Interpretation*. MIT Press, Cambridge, MA, USA.

Mark Steedman. 2000. *The Syntactic Process*. MIT Press, Cambridge, MA, USA.

Stephanie Strassel, Mark A Przybocki, Kay Peterson, Zhiyi Song, and Kazuaki Maeda. 2008. Linguistic resources and evaluation techniques for evaluation

of cross-document automatic content extraction. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*.

Beth M. Sundheim. 1995. Overview of results of the MUC-6 evaluation. In *Proceedings of the Sixth Message Understanding Conference*, pages 13–31.

Mariona Taulé, Antònia Martí, and Marta Recasens. 2008. AnCora: Multilevel Annotated Corpora for Catalan and Spanish. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*.

Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *Proceedings of the 6th Conference on Natural Language Learning*, pages 1–4.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Maksim Tkachenko and Andrey Simanovsky. 2012. Named entity recognition: Exploring features. In *Proceedings of KONVENS 2012*, pages 118–127.

Richard Tzong-Han Tsai, Shih-Hung Wu, Wen-Chi Chou, Yu-Chun Lin, Ding He, Jieh Hsiang, Ting-Yi Sung, and Wen-Lian Hsu. 2006. Various criteria in the evaluation of biomedical named entity recognition. *BMC Bioinformatics*, 7:96–100.

David Vadas. 2007. Noun Phrase Bracketing Guidelines: Version 0.9. Technical report.

David Vadas. 2010. *Statistical parsing of noun phrase structure*. Ph.D. thesis, University of Sydney, Sydney, Australia.

David Vadas and James Curran. 2007. Adding Noun Phrase Structure to the Penn Treebank. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 240–247.

Ralph Weischedel and Ada Brunstein. 2005a. BBN Pronoun Coreference and Entity Type Corpus. LDC2005T33, Linguistic Data Consortium, Philadelphia.

Ralph Weischedel and Ada Brunstein. 2005b. *BBN Pronoun Coreference and Entity Type Corpus*.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. 2013. OntoNotes Release 5.0. LDC catalog no.: LDC2013T19.

Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Nianwen Xue, Martha Palmer, Mitchell Marcus, Ann Taylor, et al. 2010. OntoNotes Release 4.0. *Linguistic Data Consortium, Philadelphia*.

Hiroyasu Yamada and Yuji Matsumoto. 2003. Statistical dependency analysis with Support Vector Machines. In *Proceedings of the 8th International Workshop on Parsing Technologies*, pages 195–206.

Jie Zhang, Dan Shen, Guodong Zhou, Jian Su, and Chew-Lim Tan. 2004. Enhancing HMM-based biomedical named entity recognition by studying special phenomena. *Journal of Biomedical Informatics*, 37(6):411–422.

Guodong Zhou, Jie Zhang, Jian Su, Dan Shen, and Chewlim Tan. 2004. Recognizing names in biomedical texts: a machine learning approach. *Bioinformatics*, 20(7):1178–1190.