AN ASSOCIATIVE MODEL OF WORD USE

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE
AND THE COMMITTEE ON GRADUATE STUDIES
OF BRAND-X UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Scott A. Waterman
February 2000
# Contents

Abstract 7

1 Introduction 9
   1.1 The Problem .................................................. 10
       1.1.1 Some examples ........................................... 11
       1.1.2 Word choice & context variation ...................... 12
   1.2 The goal ...................................................... 15
       1.2.1 A class-based model ................................. 16
   1.3 Empirical relations ......................................... 17
       1.3.1 Representing preferences ........................... 18
       1.3.2 Expressing systematic behavior .................. 19
       1.3.3 The categorical model .............................. 21
   1.4 Overcoming sparse data .................................... 22
       1.4.1 Systematic grammatical relations .................. 23
       1.4.2 Forming classes ....................................... 25
   1.5 Outline of the text ........................................ 26

2 Language structure 29
   2.1 Configuration ............................................... 30
       2.1.1 The range of effects .............................. 30
       2.1.2 Characterizing errors ........................... 32
       2.1.3 Degree of constraint ........................... 34
       2.1.4 Syntactic constraints are related to word meaning ... 35
   2.2 Structural origins ......................................... 36
       2.2.1 Counterexample: An unrestricted language .......... 37
       2.2.2 Case mechanisms .................................. 39
       2.2.3 The complexity of syntax ........................ 42
       2.2.4 Sentential interference .......................... 48

3 The Associative Model 51
   3.1 The form of the associative model ..................... 52
       3.1.1 Association ......................................... 52
       3.1.2 The Basic Model .................................. 56
3.1.3 The Class Model ....................................... 60
3.1.4 Class membership and variation .................... 63
3.1.5 Implications ......................................... 66
3.2 Simple Acquisition .................................... 69
3.2.1 Maximum likelihood estimates ..................... 69
3.2.2 Representational duals: c(W) and w(C) .......... 70
3.2.3 n-gram models ................................... 72
3.3 Category Discovery .................................... 73
3.3.1 Comparing Distributions ............................ 73
3.3.2 Identifying class membership from usage data .... 76

4 Smoothing Difficult Data .................................. 79
  4.1 Sparse data ............................................ 80
  4.1.1 Comparing distributions ......................... 84
  4.2 Estimating missing values ............................. 86
  4.2.1 Smoothing ......................................... 86
  4.2.2 Non-linguistic smoothing estimates ............. 89
  4.2.3 Linguistically informed methods ................ 92
  4.2.4 Classification methods ........................... 93
  4.3 Similarity and Classification ....................... 96
  4.3.1 Behavioral assumptions ......................... 97
  4.3.2 Structural Similarity ............................ 98
  4.4 Organizing the context space ....................... 101
  4.4.1 Grammatical analogy ........................... 101
  4.4.2 Structural context classes ....................... 102
  4.4.3 Bootstrapping Distributional Classes ........... 104

5 Quantifying similarity .................................... 107
  5.1 Sequence measures .................................. 108
  5.1.1 Set measures .................................... 109
  5.1.2 Aligned pairwise comparisons ................... 112
  5.1.3 Edit Distance .................................... 115
  5.1.4 Linguistic plausibility of comparisons .......... 120
  5.2 Richer measures ..................................... 120
  5.2.1 Locality in the metrics .......................... 121
  5.2.2 String alignments ................................ 126
  5.3 String measure experiments ......................... 127
  5.3.1 English sequence data ........................... 129
  5.3.2 Generated sequence data ......................... 130
  5.3.3 Distance measure results ......................... 131
6 Classification and Generalization 143
   6.1 Background in Clustering Methods .......................... 144
       6.1.1 Parametric ........................................ 145
       6.1.2 fixed .............................................. 145
       6.1.3 Hierarchical .......................................... 145
   6.2 Context Clustering ........................................... 145
   6.3 Classification stability and convergence ..................... 146
   6.4 Characterizing multiple contexts ............................ 146
       6.4.1 Simply conditioned models ......................... 147
       6.4.2 Lattice conditioned models ....................... 150
   6.5 Modelling the edit operations ................................ 150

7 Evaluating various models 151
   7.1 Predictive accuracy: perplexity ............................ 151
   7.2 Comparison to Part-of-Speech tagging ...................... 151
   7.3 Word sense identification .................................. 151
Abstract

I describe an associative model of word usage that categorizes the various paradigmatic syntactic contexts in which words appear, and categorizes the words according to the similarity of their syntactic usage, without resort to annotated or parsed corpus resources.

The model is derived by inductively classifying the co-occurrences of words and contexts in corpus, where contexts are compared according to syntactic configuration. Various measures of structural similarity are introduced to identify contexts with similar syntactic composition. This introduction of a measure over the space of contexts allows us to reduce the sparse nature of lexical co-occurrence data, and to extend the domain of the model to instances not observed during training.

Syntactic behavior of words can be defined in terms of their occurrence within the space of contexts. Distributional similarity allows us to use observations of word/context co-occurrence to compare the syntactic behavior of words throughout the language. By allowing comparisons between similar-but-not-equal contexts, the structural similarity measure increases the breadth of events over which syntactic behavior between words can be compared. Previous studies of distributional similarity are limited by the inability to compare non-equal contexts.

The associative model defined here represents word use in terms of a set of substitution classes, each identified with distinct types of syntactic behavior. These classes are defined in terms of the syntactic contexts constraining that behavior. Each word is assigned graded membership, according to the likelihood of its appearance in the context classes associated with that lexical class. These memberships are derived by categorizing the distributions of contexts associated with each word using an inductive clustering scheme.

The model is motivated by a desire to preserve the structural syntactic information which constrains word use, and especially to present it in a useful, applicable, and interpretable form. The categories composing the model provide two natural interpretations on the text: 1) a view of the context classes as families of syntactic constraints analogous to syntactic construction classes or part of speech, where each class has a particular distribution in terms of its realization as words of the language; and 2) a view of the distribution of any given word across these categories, as a separation of the usage of that word into usage classes, or modes of use, which may be indicative of semantic properties.
Chapter 1

Introduction

This work demonstrates a methodology for the statistical language modelling of lexical behavior, the *associative model*. The method acquires a lexical model directly from corpus, capturing the possible interactions between words and their syntactic contexts. The model consists of a set of lexical categories, and the expectations and constraints governing their appearances in various classes of syntactic contexts. The purpose of the work is to show that such a linguistically useful model can be constructed from unlabelled corpus with only minor assumptions about the nature of lexical interaction.

The *associative model* is an abstract model of words and constructions, built via corpus analysis without requiring annotated or labelled input for training. Words are described in terms of the range and distribution of syntactic categories in which they are observed to occur. The model is *associative* in that it describes the use of a word or lexical category through its typical associations with context sequences in the corpus. The breadth of syntactic environments in which a word is used forms an effective description of the underlying linguistic properties which govern the word’s behavior. This descriptive power allows one to compare and examine the use of a word or category relative to others, and to understand the scope of possible combinations which form grammatical constructions.
The categories of the model are formed by grouping together elements that share intrinsic and behavioral properties. Initially, categories of lexical environments, or contexts, are formed by grouping together sequences from the corpus that have common configurations of words. These context categories capture the families of paradigmatic syntactic contexts in which words may appear. Lexical categories are formed by grouping together words with similar distributions of syntactic behavior, as observed relative to these context categories.

Importantly, the model is organized in such a way that it is interpretable and open to inspection. The categories provide a manageable abstraction over the abundant raw corpus data, and describe the interrealions and dependencies between types of words and syntactic patterns. In this, it shares the analytic goals of more traditional lexical representations and grammatical descriptions. However, like other empirical models, it is statistically derived from the co-occurrences observed in corpus, as opposed to being the result of the investigators’ deep knowledge of linguistic interactions, or their innately driven ability to discern certain linguistic facts.

This categorial representation satisfies the two essential goals of a descriptive model: representational accuracy — the model should correctly match the data on which it is based, along with any new data from the same source; and analytic interpretability — the model should provide an abstraction over the data which is useful for understanding the underlying phenomena.

1.1 The Problem

The problem addressed here is that of how to describe the behavior of a word, relative to the usage of other words, and relative to the various possible syntactic environments in which it may occur. The kinds of behavioral properties we seek to investigate are, for instance, the license of words for use in a particular environment; the distinction between allowed and prohibited words (or word classes)
in a syntactic environment; the variation in licensing of words across various differing syntactic environments; the preference an environment might show for one word (or class of words) over others; or the specific differences between the licensed behavior of one word and that of other words.

The premise, then, is that all words carry selectional restrictions – that is, they will appear only in preferred and restricted syntactic environments. Words are bound by representational constraints of the language such that some combinations of word and context are not allowed. As the allowed combinations can be observed in language use, we can empirically identify the extent of the lexical and syntactic behavior by observing the corpus behavior.

### 1.1.1 Some examples

To describe the problem more fully, we first need to examine the behavior of words and constructions that we are concerned with. Take, for example, a very simple construction such as (1.1). A number of linguistic factor combine here to to constrain the role filled here by cat — only a very small set of fillers is available for this role in the construction. Although one has no difficulty thinking of scores, if not hundreds, of allowable substitutions, this is just a tiny fraction of the words of English. The ease of imagining substitutions is more a reflection of the enormous space of words to choose from, and of the natural human ability to disregard the the ‘obviously’ bad choices, than it is a statement about the clear limits imposed by the surrounding construction.

The orange **cat** sat on the mat.  \( (1.1) \)

The orange ____ sat on the mat.  \( (1.2) \)

It should be clear that most words can not be used meaningfully in this context. Most of the words in that last paragraph, for instance, are not allowable or
sensible fillers for the role (cf. ex. 1.3):

* The orange of sat on the mat. \hspace{1cm} (1.3)
* The orange the sat on the mat.
* The orange linguistic sat on the mat.
* The orange constrained sat on the mat.
? The orange hundreds sat on the mat.
? The orange reflection sat on the mat.
* The orange ability sat on the mat.

Rather than taking this a silly question with an obvious answer — since much of linguistics is concerned with far subtler questions of word choice — here these interactions are viewed as a fundamental property of word behavior that requires at least adequate description, if not explanation.

1.1.2 Word choice & context variation

As a speaker of English, it is easy enough to examine each of the possible constructions in (1.3) to decide which are, and which are not, valid English constructions. Given just the context frame (1.2), it is also fairly simple to identify good fillers for the gap. It has also been shown that the larger the context frame, the more limited the set of potential fillers, and also the easier it is for speakers to identify them (cf. [?miller+selfridge?]). In an information theoretic sense, the larger context frame carries more information about the utterance, leaving one with fewer choices for the gap.

There are certain types of change one could make in the context which would little change the set of reasonable and allowable choices in the central role. Context changes such as those in (1.4) and (1.5) have little impact on the choice of allowable words. Somehow, they carry the same determining information as (1.2).
1.1. *THE PROBLEM*

An orange ___ sat on that mat. \hfill (1.4)

The black ___ lay on the table. \hfill (1.5)

We can attempt to capture these similarities in the context by introducing categories of words which are allowable in each position of the context, such that substitutions would have little effect on the constraints imposed on the central gap. For instance, it doesn't matter much where the cat is, as long as it's a place, or what color it is, and so on. We could easily create a general description of each term position, reducing 'orange' to [color], for instance.

If one were to continue generalizing the positions, this categorical description would approach a set of categories very much like the classical parts of speech or typical syntactic categories. As we allow more flexibility in each of the positions, the possible choices for the others becomes less constrained in turn, and the description of each position becomes more generic and less informative. Such a progression of possible generalizations is shown in table 1.1, in which the last few steps are given in terms of traditional syntactic categories and constituent groupings. There is a clear division, however, between the stages in which we have identified each word with a generalized category, and the later stages in which some words have been 'chunked' together to form phrasal components. We will focus here only on the earlier, single-word levels of generalization and description.

**Structural constraints**

One says there are *constraints* imposed by the context (1.2) which limit the range of words which can act as grammatical fillers in the gap. One's beliefs about the nature of those constraints may depend heavily on the kind of grammatical theory one ascribes to, but it is clear that something fundamental and repeatable is occurring here.
Pairs of examples such as (1.4) and (1.5), along with our ability to generalize the interactions along the lines of Table 1.1, show that the structural interactions which limit word combinations, the grammatical constraints, do not operate at a level limited to individual words. There are classes, families of words, which can satisfy the limitations on any single given position. Irrespective of the specific meaning and reference of a word in a given context, there will be others that can be used just as acceptably in the same position. This substitutability or alternation also carries across contexts – for example, black and orange will be grammatically interchangable in most any other situation: an and the as well. These combinatorial patterns tell us that the properties that constrain the combinations are not idiosyncratically associated with individual words, but group themselves in patterns across word classes. This underlying assumption of class-based behavior is an fundamental feature of almost all syntactic theories.

There are clearly also gradations in the interactions between the behavioral classes of words, and likewise gradations in the constraints. There are many fewer substitutions we can make at the level of 1.1.b than at 1.1.c. And certainly, at the later, more abstract levels, we can insert almost any simple sentence. Understanding the distinctions between these levels of interaction will tell us a great deal about the nature of the underlying properties of the words from which they derive.

A good metaphor for this constraining behavior of the context is that of a jigsaw puzzle – the pieces surrounding a gap in the puzzle constrain the choice of...
1.2. *THE GOAL*

pieces that might fill that gap, and the larger and more complete the surround is, the more constraints imposed on the remaining pieces. Conversely, if the environment is more generic, many more pieces will adequately fill the gap. In some more complex puzzles, pieces may be partially interchangable, sharing the same outlines or color patterns. Language is similar, in that only certain words will 'fit' in a gap. The difficulty in language, however, is that there are few overtly apparent features constraining the words, analagous to the size, shape, or color of the puzzle-pieces, that would necessarily indicate the allowable fit or arrangement.

1.2 The goal

What is needed is a way to identify the limits on the combinations of words in the language, one that will show us the boundaries and extent of word behavior. If we understood the properties inherent in the words and their constraints on combinations, we would be able to work out the allowed combinations. This is the tack taken by traditional grammars.

Here, instead, we aim to make a detailed study of the allowed combinations, and work out from those the constraints on word combination. Once we obtain an empirical model that tells us which combinations of word and context do actually occur, then we can begin to speculate on the nature of the inherent properties of the words, and of their interactions, that would result in such constraints on their combinations.

For instance, consider the assumptions underlying the simple proposition that a ‘transitive verb’ is a ‘verb’ that takes a direct object. Both of the terms on the right hand side, ‘verb’ and ‘direct object’ are in need of definition. Typically, terms such as these rest on pre-theoretic intuitions of sentence structure, ‘verb’-hood, real-world semantics, etc., and are not rigorously defined. Even with the best of intentions, our deep dependence on linguistic modes of thought and expression make it difficult, if not impossible, to consider ‘verb’ apart from our intimate
knowledge of the term as associated with actions in the world.

In the approach taken here, and in other empirical modelling, we instead define classes and behavior solely in terms of their combinatorial distribution and co-occurrence behavior, irrespective of whatever other properties we 'know' the words to have. Models such as these make propositions more of the form "a word in class B37 is allowed in environments defined by the following word distribution...". Statements such as these are definitionally grounded only the observable data. Then, if one finds a correspondence between these models and more traditional syntactic theoretic descriptions, it will confirm and enlighten the pre-theoretic notions that derive those theories.

1.2.1 A class-based model

The methods demonstrated here construct a hierarchical system of word classes and of context classes that describe grammatical behavior. Context classes, which are generalizations of explicit context descriptions such as (1.2), are associated with a list of possible words as gap-fillers, each with a weight proportional to the strength of the association. Word classes, analogous to very fine-grained parts of speech or lexical categories, are associated, again through a set of weights, with the contexts in which they are allowed to appear. A single word or context might have membership in one or many of these classes, in order to describe its complete range of possible behavior. These sets of classes form an abstract categorical description of word behavior in the language. At an appropriate level of abstraction, such behavioral categories may also correspond to the underlying linguistic and semantic constraints which govern language use. It is to elucidate these underlying causes and interactions that we seek to build the model.
1.3 Empirical relations

These graded class membership values are acquired using observations over corpus data as their foundation. By using corpus observations to note which words actually are used in combination with a particular context, we can construct a model which describes the strengths of these associations, and the sets of words and contexts allowed or restricted given circumstances. Only after a descriptive model has been constructed, creating an abstract description of the behavior of the linguistic combinations, are we free to explore more fundamental models of the constraints which govern and limit those combinations.

Of course, a representation of word use that identifies which words are and are not compatible with the constraints of a context still leaves us no direct information of the nature of the underlying constraints. However, what we would have is a mechanism to compare those constraints implicitly, by using classes that comprise them as a description. Because the set of word classes and context classes can be much more compact than the original word use data, the nature of the constraints and interactions will become far more apparent. Still, it is abundantly clear, from the facts that users of the language agree 1 on what is and isn’t allowable in gaps such as (1.2), and that the allowable set doesn’t change greatly when we perturb the context slightly, as in (1.4) or (1.5), that there is a systematicity of the relationship between the words of the gap and the context frame.

From an empirical standpoint, the obvious approach to identifying this relationship is to study a very large corpus, and to simply list these co-occurrence sets. We simply note which pairs of context and word appear in actual usage, and keep a list, along with frequency information, so that we can judge relative preferences, and distinguish occasional or irregular usage from the common combinations. This kind of straightforward approach would seem to satisfy the descriptive modelling requirements.

---

1 For the most part. ??
This sort of empirical bookkeeping exercise will also require that we develop a model of distributional comparisons in order to make judgements about preference and similarity. One needs to be able to compare the occurrence distributions of two words, for instance, in order to determine whether and to what degree one may be substituted for other. Of course, with enough data, one may be able to determine this substitutability directly from the counts themselves.

### 1.3.1 Representing preferences

If we could indeed gather enough co-occurrence information to use large and linguistically relevant contexts in a statistical corpus study, the first thing we might like to capture is the sets of corresponding partners for each word or context in actual usage. For instance, we can find all the occurrences of the contexts (1.2) and (1.5) in a corpus, and identify the sets of words that appear with them, in order to evaluate whether they do in fact project similar constraints on word use. Similarly, we can find all the contexts in which a single word, such as *cat* appears, to evaluate the features of the syntactic environment compatible with that word.

These occurrences give a representation of the operational preference sets of each word or context. Let us use the function notation \( c_i(w_j) \) to indicate the preference a context \( c_i \) shows in usage for a single word, \( w_j \). This should be a function which is larger for those words for which there is an observed preference — such as the frequency of the context/word co-occurrence in corpus, or some other estimate of the conditional co-occurrence probability.

\[
\begin{align*}
    w_j(c_i) & : \ c \in C \mapsto [0, 1] \quad \text{a word } w_j \text{'s preference for a context } c_i \\
    c_i(w_j) & : \ w \in W \mapsto [0, 1] \quad \text{a context } c_i \text{'s preference for a given word } w_j
\end{align*}
\]

Similarly for a word, the function \( w_j(c_i) \) will represent the preference \( w_j \) might have for some context frame \( c_i \). The value of \( w_j(c_i) \) also is larger for contexts which appear with the word more frequently in the corpus. We can normalize these functions to the range \([0,1]\), where a value of zero indicates no co-occurrence is
1.3. EMPIRICAL RELATIONS

possible, and values towards one indicate a strong preference for this combination of word and context.

These functions, if known for all words and contexts, would provide a very powerful descriptive model of combinatorial behavior. The reason for separating the word’s preference from that of the context is simply that the relationship is not necessarily symmetric. In the extreme, a word \( w_j \), for instance, might appear exclusively in only one context, \( c_i \), while many other words also are used in \( c_j \). Thus, \( w_i \)'s preference is limited exclusively to the one context, while \( c_j \) may appear preferentially with some other word or set of words.

1.3.2 Expressing systematic behavior

While it is interesting to know the preference an individual word shows for a single context, it may not help directly in the understanding of the structure of the language as a whole. What is also needed is a way to view the behavioral patterns of words or contexts across the breadth of their use, in order to identify the ways in which they are used similarly and the ways in which they show conflicting or complimentary patterns of use, and to identify the linguistic properties of those patterns.

For instance, we would like to use this information about word/context co-occurrence in order to see that words such as dog and cat are used in similar contexts, such as (1.2), or “...pet the ____.” Also, we want to know that these words do not share the same sort of behavior with words such as price, color, or began. This kind of analytic knowledge, while 'obvious' to speakers of the language, allows a deeper understanding of language behavior, beyond the direct descriptive model.

In order to do this, the preferences of a given word across all contexts can be assembled, giving a characterization (eq. 1.8) of the selectional restrictions projected by that word in terms the contexts it is allowed and not allowed to appear

25 February 2000
in. Or, we can collect the preference information for a given context, relative to all words (eq. 1.9), to show the constraints imposed by that context in terms of the words allowed to operate within it.

\[
w_j(C) = \{w_j(c_1), \ldots, w_j(c_C)\} \quad \text{word preference for all contexts} \quad (1.8)
\]

\[
c_i(W) = \{c_i(w_1, \ldots, c_i(w_V)\} \quad \text{context preference for all words} \quad (1.9)
\]

**Comparisons and similarity**

These functions represent the behavior of a word or a context across the entire language, and will support comparison of the behavior of words or contexts. If we look at the the entire range of contexts associated with a single word, this context vector gives us a kind of behavioral profile for that word. By comparing the total allowed behavior of this word against that of others, we see where the selectional constraints of various words overlap or differ. Various measures of *distributional similarity*, based on statistics or information theory, are potentially available to judge the similarity of behavior and to make these comparisons.

These similarities can help to identify the fundamental types of constraint behavior that may be operating between the words and contexts. For a given context, the preferences in use across all words gives a characterization of the contraints imposed by that context. Shared preferences in behavior will show up as overlapping entries in the respective vector of word preferences. Explicitly, we can only observe the allowed words and their preference for appearing with this context, but implicit behind those are the constraints embodied in the speaker’s knowledge of the context and words that limit their combinations. Having this representation of the global behavior allows us to compare the systematic behavior of the words and contexts.
1.3.3 The categorical model

We can further condense these similarities into a compact form by introducing classes of words and contexts. Besides agreeing with our earlier intuitions about allowed substitutions, a class-model provides a more compact, abstract representation of the lexical interactions. This abstract form simplifies analysis and understanding of the model. It also simplifies mechanisms for inference and prediction using the model. For instance, if we have placed a word in a certain lexical class, then we should be able to predict the use of that word in environments associated with the class, even if we have never observed the word in those environments.

The explicit model described by eqs. 1.8 and 1.9 has some severe limitations. Primarily, direct acquisition of a model of this form is extremely limited. As has already been stated, and will be examined in detail in ch. 4, the sparse nature of the linguistic examples available from corpus data guarantees that we will have observed values for only a very few of the possible combinations of word and context.

The class model also provides a means for reducing the complexity of the explicit model. Not only do the classes provide a means of analysis over the model, but the class representation reduces the model’s size, since now only the associations between classes need to be represented. This makes it simpler to construct, represent, and make use of, the associations between words and contexts.

Form of the model

The form of the categorical model is analogous to that of the explicit word/category model. Again we need two sets of functions: one to represent the association that lexical categories, $\ell \in \mathcal{L}$, have for the context categories, $\gamma \in \mathcal{\Gamma}$, and the other to represent the association of contexts with lexical categories.

Of course, we also need to represent the membership of each word or possible
context in these categories. Eqs. (1.10 and 1.11)

\[ \ell_i(w_j) \] membership of \( w_j \) in class \( \ell_i \in L \) \hspace{1cm} (1.10)

\[ \gamma_i(c_j) \] membership of \( c_j \) in class \( \gamma_i \in \Gamma \) \hspace{1cm} (1.11)

Eqs. (1.12 and 1.13) show the relation between the association functions of the class model and the explicit model.

\begin{align*}
\text{lexical association:} & \quad \ell_j(\gamma_i) \leftarrow w_j(c_i) \hspace{1cm} (1.12) \\
\text{context association:} & \quad \gamma_i(\ell_j) \leftarrow c_i(w_j) \hspace{1cm} (1.13)
\end{align*}

Each class association function gives the conditional expectation of the appearance of a member of a class in the counterpart set. For example, \( \gamma_i(\ell_j) \) is the probability that a member of the lexical class \( \ell_j \) will be used, given an appearance of a context in the class \( \gamma_i \). We will define these explicitly in terms of the word and context probabilities in the later sections on category acquisition.

1.4 Overcoming sparse data

The disappointing fact is that this counting approach will not work well at all, due to the sparse nature of word co-occurrences in even the largest of corpora. Precise configurations of words recur so infrequently in real text that the statistics are mostly meaningless.

However, even with these problems, there have been studies inducing word-classes using context co-occurrences. The two most popular tactics for increasing the occurrence rates of these word/context events both involve limiting the
1.4. OVERCOMING SPARSE DATA

size of the context, since a smaller context will have a greater chance of occurring. One branch of study typical of \( n \)-gram models uses extremely small contexts — typically 1 or 2 neighboring words ([Church and Hanks, 1990, ?dagan?, ?etc.?]). Although there are still difficulties with observing enough word pairs or triples to build complete models there are also many clever techniques to overcome them. Experiments show that even this small a context enables one to find interesting correlations and build practical models ([Brown et al., 1992, ?mcmahon?, ?etc?] ). However, the relation between one word and its neighbor is a very limited interaction, from a linguistic point of view. There are clearly many factors involved in word usage that are not captured by these models.

A deeper approach involves selecting co-occurrence sets based on grammatical dependencies — parent/child relationships in a parse tree, for instance ([?magerman?, ?history-based-parsing?, ?michael-collins?, ?subcat-studies?]), or Subject-Verb relationships ([?greffenstette:book?, ?pereira+lee?, ?lee?, ?rooth:workshop?]). This also limits the number of variables in the observed events, and resulting in more robust statistics. This also focusses the observation on interactions that have a recognized linguistic significance, and the resulting models are easier to integrate with linguistic theory than the 'who's-my-neighbor' interactions of the \( n \)-gram models. The drawback, though, is that this requires a linguistic parse analysis of the data before counting co-occurrences. In this study, we are trying to identify the lexical and structural types inherent in the corpus without biasing it to a pre-existing analysis or even a particular linguistic theory.

1.4.1 Systematic grammatical relations

The primary contribution made here towards this problem of sparse co-occurrence data is an approach to collapsing groups of contexts that are not
strictly identical. Because a long context frame such as (1.15) does not recur frequently, studies have typically been forced to limit co-occurrence studies to much shorter context frames, such as (1.16) or (1.17).

\[
\begin{align*}
\text{The orange} & \quad \text{sat on the mat.} \\
\text{orange} & \quad (1.15) \\
\text{the orange} & \quad (1.17)
\end{align*}
\]

If, instead we could find a collection of longer contexts that are similar enough to be treated equivalently, at least as far as their grammatical projections are concerned, we would find many more such frames within a given corpus. For instance, if we could somehow decide that the contexts (1.18–1.20) project similar if not identical grammatical constraints on the gap position, much like (1.4) and (1.5), then we could combine them and treat them as a single unit, with a much higher recurrence rate than any of them individually.

\[
\begin{align*}
\text{downloaded classified} & \quad \text{into his unsecured home computer} \\
\text{copied classified} & \quad \text{onto his unsecured home computer} \\
\text{copied restricted} & \quad \text{onto her home computer} \\
\text{copied classified} & \quad (1.18) \\
\text{copied restricted} & \quad (1.19) \\
\text{copied classified} & \quad (1.20)
\end{align*}
\]

We have already discussed one mechanism for representing such similarity in grammatical inference – the use of class abstractions. If one had a reasonable idea that, say, \textit{classified} and \textit{restricted} were of the same grammatical type, then it becomes easier to predict that the contexts (1.18–1.20) will have similar projections on the gap position. Similarly for pairs such as \textit{into/onto}, \textit{his/her}, or \textit{copied/downloaded}. 

25 February 2000
1.4.2 Forming classes

By using word classes as equivalence classes, we can treat groups of non-identical contexts as if they were identical. This reduces the number of unique contexts in the corpus, and raises occurrence rates, thereby reducing the problems of sparse co-occurrence data. However, at the start of the study, we have no word classes to use. Fortunately, we can look at other properties of the contexts in order to judge similarity.

Two contexts that are not precisely alike may still have much in common, however. What we are really measuring when we compare contexts such as (1.18) and (1.19) is the structural similarity between the two. If we consider 'downloaded' the equivalent of 'copied' here, then we can claim the contexts are also equivalent, because the rest of the structure is the same. But, we could just as easily count this as a kind of matching error between the two, and count our expectation of similarity between the contexts as the degree to which the structure of the two can be matched, and discount for mismatches such as this.

Later, we explore a number of possible methods for comparing contexts in this way, using their intrinsic properties, rather than their distributional properties, to form classes of similar contexts. These context classes formed using structural similarity are the bootstrap mechanism through which we can fit an initial form of the associative class model.

Without collapsing these sets of contexts, the co-occurrence rates for word-context pairs is so low that there is simply not enough evidence to make a comparison of the distributional properties between two words, for any context more than about 3 words in length. With the construction of these initial context categories, we boost co-occurrence rates to the point where we can begin to compare the usage of words across the contexts, and judge similarities between word uses. In turn, this discovered word similarity can be used to better inform the construction of the context classes, and so on.
It is by iterating between these two methods of classification that the associative model of word and context use is constructed.

1.5 Outline of the text

Chapter 2 contains a discussion of the assumptions underpinning of the associative model. The principle discussion centers about the expectation that words will not behave arbitrarily, but will tend to have forms of behavior shared among groups or classes of words. The argument is based on the complexity of processing the relational structure of language, and the efficiencies gained through the use of a regular syntactic structure. This discussion sets out a small number of basic syntactic structuring mechanisms that we might expect to find in language. The remainder of the chapter explores how the study of the purely formal (syntactic) structure of the language can lead to a better understanding of the semantic nature of the language system.

Chapter ?? outlines the properties of the associative model, and formalizes the interactions between words and contexts. It shows how the co-occurrence data can be represented in both a word-centric fashion, as association weights against the contexts, or as a context model, using association weights against the words. The model is then extended to include category representations for items grouped into behavioral classes. Distributional similarity metrics that can be used to compare the behavioral properties of words or contexts are discussed, along with ways they can be used to identify the underlying classes of behavior. The use of these distributional methods is compared to other studies.

Chapter 4 contains a discussion of the sparse nature of linguistic co-occurrence data, and how this adversely affects the comparison of distributions outlined in the previous chapter. Various methods of interpolation and smoothing used in the literature are examined in relation to the very large space of contexts. The notion of structural similarity smoothing is introduced as a way to organize this sparsely
1.5. **OUTLINE OF THE TEXT**

populated space of contexts, and bootstrap the formation of context and word classes.

Chapter 5 discusses some appropriate structural similarity measures which can be used as the basis forming context classes using structural similarity. These measures include edit distance, the localized edit distance, and various set comparison measures. Differences in the measures which affect the nature of the resulting model are examined. Some experiments are presented which quantify the distinctions in the measures, and expose some of the criteria necessary for choosing an appropriate measure for our task.

A randomization experiment is also presented, which gives a baseline for these similarity measures. By comparing the structural similarity measures on real language data against the same measures evaluated on a simulated data set with identical first-order distributional properties, we show the extent to which the non-random linguistic structure of the data can be identified and extracted by the similarity measures.

Chapter 6 presents a number of techniques for deriving classes based on the similarity measures. The properties of the classification techniques are explained in terms of their interactions with the similarity measures and model formation. The bulk of the chapter discusses the building of a full model from a large corpus of newswire text.

this is followed by an evaluation of the constructed model. Intrinsic properties, such as model entropy, as well as comparisons to other computational models are used to show its effectiveness.
Chapter 2

Language structure

In order to design an appropriate model of word use, we first need to understand what kind of patterns and structure we should expect in their real use in language. This chapter outlines a very general theory of the configurational properties of language, as imposed by the efficiency of its use as a device for representation and communication. The immediate result, which should not be surprising, is that a small number of configurational constraints types – grammatical relations corresponding to functional combinations – greatly improve the communicative efficiency of sequences of words, in terms of the number of words required and the processing power necessary to decode them. This result implies that we have every reason to expect that a natural language, optimized in part for communicative efficiency, should exhibit exactly the kind of regular structure that a study of corelated co-occurrences is able to find. The effectiveness of the associative model and similar methods in the literature is purely contingent on the presence of this regular syntactic structure.

The deeper result is that regularities in semantics should coincide with structural, syntactic regularities. Thus, any procedure which sets out to find certain structural, distributional regularities in the language will also find parallel semantic properties. This implies that a properly structured acquisition of distributional
properties can not only be used to form a syntactic model of a language, but can also
be used to bootstrap a study of lexical semantics, and can form a basis for
semantic inference procedures.

2.1 Configuration

The acquisition of the associative model requires that we discover some non-
obvious properties of words from their appearance in corpus. Specifically, the
model will capture patterns of context, along with patterns or word use within
those contexts, as abstract categorical descriptions. We can later use this de-
scriptive model to accurately predict previously unseen word usage and to infer
similarity of meaning across sets of words.

In order to model these patterns, we first need to understand what kinds of
patterns we can expect to occur. This will give us an outline of the properties of
word use we need the model to represent, and so give rise to the form of the model,
as well as hint at methods for acquiring the parameters from actual corpus usage
data.

2.1.1 The range of effects

It should be clear that there are real constraints on the combinations of words
in language. One can not simply toss a string of words together and expect to
get a meaningful, or well-formed, expression. For example, (2.1) simply cannot
be considered an English construction, even if all the symbols are individually
acceptable.

\[
eat \text{ direct Reluctantly have (fish ? mechanism the strengthen:} \tag{2.1}
\]

Most would say that (2.1) is unacceptable because it does not meet the obvious
formal\(^1\) surface properties that are required of meaningful sentences — it does

\(^1\)"Formal" here meaning 'to do with form or shape,' not necessarily implying any degree of preci-

not match the syntax of English — not because of any deeper semantic considerations of failure in meaning. This sequence violates the normal arrangement of words, where 'normal' is somehow judged with regard to properties unrelated to specific meaning. If a word refers to something that is or could be participating in a relationship, we call it a NOUN. We know that, typically, in English, nouns appear before, but sometimes also after, a VERB, which identifies the relationship, which is possibly one of action, between those nouns or to the world at large. QUANTIFIERS appear before nouns, telling us to how many and to which items they might refer. These rules of placement and combination, which operate without regard to the deeper meaning of the words, are what is usually referred to as syntax or grammar.

One needn't look closely into the meaning of words in (2.1) to see that the verbs don't have nouns as subjects, the determiner is preceded by a noun and followed by a verb, and a host of other obvious form violations have been committed. It fails, by this argument, because there are rules of English grammar that allow certain combinations of word types and disallow others.

Some purely formal theories, such as some versions of transformational grammar, posit a process in which the structure of a sentence is created, and later specific words are introduced into positions by virtue of their category membership (cf. [Chomsky, 1957, ?chomsky:lsit?]). 'Category' in this sense is construed rather broadly, corresponding to traditional, high-level grammatical labels such as NOUN, VERB or ADJ. For example, if the first stages generated a structure such as (2.2), it might be filled in with lexical items as in 2.2).

$$\text{NP} \quad \text{ADV} \quad \text{V} \quad \text{PREP} \quad \text{ADJ} \quad \text{N}$$

$$\text{I readily believe in fairy tales.}$$

This works well enough, up to a point. There is nothing in this theory to restrict us from filling in lexical items which properly belong to the categories, but which do not combine in meaningful ways. Thus, we can fill in the same grammatically
acceptable structure with combinations of words which make less and less sense, as shown in (2.3)-(2.3). The logical result of this sort of replacement are Chomskian “green ideas”-type examples, which are, in that view, grammatical “though nonsensical.” [?chomsky:ss?]

\[
\begin{array}{cccccc}
\text{NP} & \text{Adv} & \text{V} & \text{Prep} & \text{Adj} & \text{N} \\
\text{The desk} & \text{yesterday} & \text{flew} & \text{through} & \text{rainy} & \text{weather.} \\
\text{Sand} & \text{metallurgically} & \text{accepts} & \text{between} & \text{pointy} & \text{water.} \\
\text{Cheese} & \text{grainily} & \text{waives} & \text{of} & \text{sapphire} & \text{maxima.}
\end{array}
\]

(2.3)

### 2.1.2 Characterizing errors

The kinds of constraints evident in these examples are certainly different in magnitude, but the question should also be considered whether they are also different in nature. It is true that we must examine the structural replacement examples much more closely than the ‘word salad’ of (2.1) to realize that something is wrong with what we are reading, that the words on the page don’t come together to make sense. How can we characterize the way these examples fail to meet our expectations about what a well-constructed, intelligible sentence or utterance in a language should be?

We can decipher the later examples to some degree, only because of the relative familiarity of the words used and the roles they are used in. Fortunately, most words have a relatively fixed meaning, and appear most often in only one of these high-level grammatical role categories, making the listener’s job of identifying the intended role of each word easier. When the rules of placement and combination are broken, however, the usual roles of the words come into conflict with the roles they must have in order to function meaningfully in combination.

The words as used in (2.1) do not have nearly the flexibility in roles that they need in order to make the sequence an interpretable English sentence.
2.1. CONFIGURATION

Sources of information

The information deciding the role each word plays in a construction comes from at least two sources. One is lexical knowledge – knowledge of the word itself; what roles it usually takes in certain circumstances, what other terms it usually interacts with, and other specific knowledge of the one particular word. The other source is syntactic knowledge – our general knowledge of how words combine, applied to the environment in which the word is used. Information about the possible roles and meanings a word might assume is given to us by the arrangement of its neighbors. General knowledge of word arrangements limits serious problems, such as those of (2.1). More detailed information is also available — because a word must interact with its neighbors, our knowledge of how the neighbors normally operate imposes certain choices on how the word in focus can behave.

To illustrate the differences of these knowledge sources, we can construct an example with a novel word, with which we have no pre-associated lexical constraints. We don’t know what this word might mean, but imagine it is used as in:

You should add some gardle to the soup. \[ (2.4) \]

If one assumes that this sentence is interpretable, then we now know many things about the new word, all projected onto it by the remaining words of the construction. “add some X to the soup” tells us generally that gardle is a thing, a noun, since it is something you add (the lack of a right-hand nominal dependent rules out the possibility that it is a modifier of any sort; as in “some gardle turnips”). More specifically, it refers to a kind of thing that is composite and divisible, since it is “some gardle” rather than ‘a’ or ‘the’. Also, it refers to something that goes into soup, and so is probably edible (that Scandinavian parsley people have been talking about, perhaps). The formal properties of syntactic combination have taken effect here, projecting an interpretation on the word, which seems acceptable because we have no prior knowledge about the word.

25 February 2000
CHAPTER 2. LANGUAGE STRUCTURE

What would happen, though, if we were to take a familiar word and place it in the same construction? Let me use the word *strengthen*, which was so out of place in (2.1). We now have

You should add some strengthen to the soup. \( (2.5) \)

which seems totally incongruous – at least as much, if not more than it did in its appearance in (2.1). Our familiar interpretation of *strengthen* as a verb indicating an action clashes with every constraint on meaning that the remainder of the construction projects. No previous knowledge of this word prepares us to make such a leap in altering its familiar interpretation. The general syntax forces a nominal interpretation; the use of ‘some’ gives a composite reading; the verb, ‘add’, tells us the referent is manipulable; and the fact that it is added ‘to the soup’ gives us edibility, all of which conflict directly with its familiar interpretation.

2.1.3 Degree of constraint

This example (2.5) sets up a complex matrix of projected constraints, all of which, from the nominal interpretation to the expectation of edibility, are broken by our existing knowledge of the word *strengthen*. All of the expectations that we have associated with the word, and all of the knowledge of its use in normal constructions, clashes with the relations forced upon it by the remainder of the construction.

The earlier *gardle* example (2.4), in contrast, is acceptable simply because the novel word does not carry any constraints along with it. It is merely acting as a carrier for the matrix of constraints projected upon it. We could use this word in place of practically *any* word, in any context, with identical results. While it may not have a predetermined meaning, we are more or less happy to allow its interpretation according to the constraints imposed by context. Further, those constraints tell us, more or less, what the word must mean.
2.1. **CONFIGURATION**

Example (2.1) violates the primary syntactic constraints of the language in that even the most general level of configuration of the terms does not allow the application of the normal constraints between them. Each word is used in a context so foreign to its normal modes of use that we can not even identify any familiar pattern.

The less incongruous examples (2.3)–(2.3) satisfy the basic agreements of context, but associations implied by the configuration fail to hold at a finer level of detail. In example (2.4), the implications of the configuration are quite acceptable, even though the word *gardle* contributes no information to the meaning. The final example, (2.5), contains a very different sort of error, one which the configuration imposes a coherent set of constraints, but that set clashes in every way with the familiar use of the offending word.

### 2.1.4 Syntactic constraints are related to word meaning

These examples demonstrate clearly that words place constraints on the words and constructions appearing jointly with them. Any configuration of words creates such constraints, whether they can be combined into a sensible, unified whole or not. If one is to call the construction part of the language, however, the words associated under that configuration must not carry conflicting information. I take this as a basic observation about the structure of language.

Some of that information is traditionally called ‘syntactic’, such as the category membership (N. V. etc.), while some is more toward the ‘semantic’ end of the spectrum — for instance, we know that cheese doesn’t do *anything* through any kind of maxima. However, we find that these aren’t truly separable properties of lexical items, but differ only in the granularity of the properties they describe. One does not find, in any language, high-level syntactic categories that are assigned to lexical items irrespective of their meaning. Verbs (words assigned to the VERB category) for example, denote actions and relations. They never denote
fixed objects one might find on one's desk. The information one has about a word's meaning, whether on the gross level (e.g. as is a member of VERB) or at finer grain (e.g. the non-volitional nature of cheese), determines the types of structures it can sensibly appear with.²

In this informational constraint view of language, effects that often are identified separately as syntactic or semantic are inextricably tied together — the configurations in which a word is used are dictated, in large part, by what that word means. The following sections demonstrate that the coincidence of these separate effects results in a far greater efficiency of representation than would be possible if configuration were independent of meaning.

There is nothing, I believe, in the functional concept of language as a communicative and representational device that limits the ordering and structure of combinations. The conclusion supported here, however, is that it is the practical, operational concerns of instantiating and using a language that lead to configurational constraints on word use.

### 2.2 Structural origins

We have seen that there are definite limits on the manner in which words may combine in language. Is there anything in the concept of language itself that inherently limits the ways in which symbols might be put together to create complex relationships? In that these limitations are easily violated — it is simple to form a group of words, tossed together like those of (2.1), for which there is no available interpretation — it is easy enough to see that such limits exist. What is the structure of well-formed strings such that they form a useful part of the language?

²Some would say that the knowledge that *garble*, for instance, is a noun, is purely syntactic knowledge, and not related in any sense to what the word means. I would argue, though, that even knowledge of just the part of speech, as here, vastly limits the range of possible meanings, and as such contributes very real information to the meaning of the word. Since its interpretation is restricted to some part of the semantic space, it would seem that this is truly knowledge, however imprecise, of the *meaning* of the word.
2.2. **STRUCTURAL ORIGINS**

One problem quickly encountered pursuing these questions is that we all have strong preconceptions of what ‘language’ is. For the present discussion, I take a rather loose view that language is primarily a representational system, using linear sequences of symbols to refer to mental constructs, which can exist in familiar predicate-argument relationships. Prototypical representations of these complex, multi-relational mental constructions can be found in modern cognitive theories (cf. [Sowa, 1984]), linguistics (cf. [montague?]), and formal logic theory (cf. [frege?, russel?]), stretching all the way back to the predicative mentalist logics of ancient philosophy (cf. [aristotle:?]).

Language, as used here, is a system for relating linear strings of symbols to these highly structured, non-linear mental constructions. Whether the symbols refer to some set of basic primitives or could refer recursively to other constructs and relationships is not in itself material.

While a language system with unconstrained configuration of words is conceptually feasible, it is not without implications. Some simple structural limitations greatly increase the functionality and economy of the system. Some of these economies, such as limited memory capacity and sensory bandwidth, are inherently constrained by our neurological construction. We can begin by taking these as a practical starting point.

### 2.2.1 Counterexample: An unrestricted language

Let us start with a simple example, representing a simple symmetric two argument relation as a string of words in a language. Let \( Z \) be the relation, while \( X \) and \( Y \) are the arguments\(^3\). In predicate logic we might write the relation we want to express linguistically as \( Z(X,Y) \). \( Z \) is a predicate which takes two arguments, \( \lambda x y. Z(x,y) \). Further, \( Z \) is symmetric in its interpretation, so that \( Z(X,Y) = Z(Y,X) \), and either will do. Let us introduce three words corresponding in meaning to

---

\(^3\)I will use capital roman letters, \( X, Y \), etc. for conceptual entities, and small roman letters, \( a, b, c \), for words
these three entities:

\[ [a] \to X \]
\[ [b] \to Y \]
\[ [c] \to Z \]  \hspace{1cm} (2.6)

We want to be able to convey the relationship \( Z(X,Y) \) using these symbols \( a, b, c \). To begin to see where some structural constraints might arise, let us start by looking at the extreme alternative: a language with no syntactic properties at all. Imagine there exists a language \( \mathcal{L}_0 \) in which symbol order and configuration play no part at all in determining the functionality of the language, where I will define \textit{functionality} as being the representational and communicative aspects of language, and which we can loosely refer to as the 'meaning' of statements.

Because we have no constraints on configuration, if \( a, b, \) and \( c \) are symbols in \( \mathcal{L}_0 \) we can state by assumption that the following sentences will have precisely identical interpretation, namely \( Z(X,Y) \):

\[ [a\ b\ c] \]
\[ [a\ c\ b] \Rightarrow Z(X,Y) \]
\[ [b\ a\ c] \]  \hspace{1cm} (2.7)

To ground this, say, for instance, that \( X \) and \( Y \) are two objects in a 'blocks world,' two blocks, say, (but more generally any objects, physical or not – e.g. ideas, a chair, the observable property of blueness, etc.) and \( Z \) is some symmetric relation that can exist between two things, such as being near. We could then paraphrase any of the expressions in (2.7) as "the things referred to by \( a \) and \( b \) (i.e. \( X \) and \( Y \)) are near each other." The non-configurational assumption allows us to re-order any string and claim that it must retain its interpretation. In this language, the composition of the meaning of the symbols, \( a, b, c \), into the meaning of the relationship \( Z(X,Y) \) is accomplished solely through the semantic features of the symbols. Ordering plays no part.

One of the more difficult problems of using a linear language is the representation of ordered, complex, nonlinear relationships. As far as symmetric relations such
as 'near' are concerned, this language $L_0$ has adequate expression. But, if the relationship is not symmetric, it fails to represent the desired meaning. Say that $c$ is a symbol for 'on top of' instead of 'near'. To be able to represent the situation adequately, we must be able to identify which of $a$ or $b$ is on top. The immediate result is that this is not possible in $L_0$ with the three symbols $a$, $b$, and $c$. Since the expressions (2.7) are, by assumption, equivalent, it is not possible to have two separate representations for 'a on top of b' and 'b on top of a'.

One could of course resolve this difficulty by imposing an ordering constraint on the symbols, using, for instance, the first object as the one on top, and the other underneath. In declaring that in $L_0$, interpretation is not dependent on configuration, we eliminate this possible solution.

### 2.2.2 Case mechanisms

What we are missing so far in interpreting this language is a mechanism for identifying the case of the terms – generally, their correspondence to the roles of the relation (cf. [?filemore:case?]). For instance, imagine the meaning of $c$ to be a logical predicate something like $\lambda z_1 z_2 Z(z_1, z_2)$. If $Z$ is non-symmetric, we need a mechanism for separating and identifying the two argument variables, $z_1, z_2$. In the case scenario, we make some fixed assignment between the two variables and a finite set of possible cases. The possible argument fillers, $a$ and $b$ in this example, are then also required to carry case, such that a listener may assign them properly to the roles of the predicate. The filler/role relationship between $c$ and $Z$, being the representation of the relation itself, is not typically identified as a 'case' per se, but is differentiated as the verb or predicate. Since we are already positing that the listener knows the referents for each of these symbols, the problem of which symbol represents the predicate is avoidable, if we ignore for the moment the possibility of noun/verb polysemy.

There are a number of mechanisms by which we can label or assign the case
of the argument terms in the language, so that they compose properly into the
semantic relationship we desire to communicate. However, almost all of them will
involve using configuration as a mechanism for making this assignment. Here we
give an overview of a few mechanisms commonly employed in natural languages.

Morphological case Morphological case marking, the mechanism which the
general case theory is unfortunately named for, involves a systematic alteration
of the base lexical item which identifies the case in which it is to be used. This
change can take the form of affixation, infixation, vowel shifting, or other struc-
tural morphological change. In this example, we might have two affixes, one of
which *₁ marks its item as being the first argument, z₁, the other, *₂, which marks
the argument belonging to the second position, z₂. We would then write any of the
forms (2.8) to express the proper relationship.

\[
\begin{align*}
  a_1 & b_2 c & \text{morphological case for argument assignment} \\
  a_1 & c & b_2 \\
  b_2 & c & a_1
\end{align*}
\] (2.8)

Functional markers Similarly, case can be identified externally to the lexical
item through the use of prepositions (e.g. English) or post-positional markers (e.g.
Japanese). In this strategy, case is identified by associating an external marker
with the base item, where the sole purpose of the marker is to identify the case of
the fillers so that they may be attached to the proper functional role in the predi-
cation. This is quite similar to an affix mechanism, except that the marker is not
considered an integral part of the lexical item, but nevertheless appears adjacent
to it as a marker (although other components, e.g. modifiers or quantifiers, are
allowed to interpose).
If we use a prepositional scheme, we might have a marker item $m_1$ which identifies the following item as the first argument $z_1$, and $m_2$ which identifies the second as $z_2$. We can then write any of the forms (2.9) and ensure the proper interpretation.

\[ m_1 \ a \ m_2 \ b \ c \quad \text{prepositional marking for argument assignment} \]
\[ m_1 \ a \ c \ m_2 \ b \]
\[ m_2 \ b \ c \ m_1 \ a \]  

2.2. **STRUCTURAL ORIGINS**

**Configuration as a case identifier** Case can also be identified purely positionally, through the order and configuration of the set of words that represent the predication. In the simplest situation, it might be that the first argument filler to appear is assigned the first role, the second to the second, and so on for as many arguments as there are. Since the word denoting the predicate must also appear in the sequence, though, it could be used effectively as a marker to separate the arguments into two groups, those before and those after it. This is typical of so-called SVO languages, such as English, in which one argument of a two-argument predicate appears in the leading position, and the other appears afterward. Using this system, by writing (2.10), we can get the proper interpretation without using any additional marker words.

\[ a \ c \ b \quad \text{strict ordering for argument assignment} \]

**Configuration and case**

It must be observed, however, that none of these case-marking mechanisms work without considering word order. It's obvious that configuration is essential for the configurational case identification, but it is also required of the other mechanisms described above. This is easiest to see for the prepositional functional markers. In

25 February 2000
order to know which object the marker \( m1 \) is associated with, the two would have to appear immediately following one another, or with some restricted set of words interposing. If we have for instance the sequence \( a \ b \ c \ m1 \ m2 \) it is unclear which of \( a \) or \( b \) the \( m \)'s correspond to. If the language uses functional case markers such as prepositions, then configurational constraints must be present.

The same is true for morphological case marking, although it operates at a more subtle level. If we consider a case-bearing suffix as a linguistic element that, of necessity, must appear immediately following the modified morpheme, then we are relying on this configuration as the indication that the two are linked. It is impossible, for instance, to have the case-bearing suffix appear after a different word than the one which is assigned the case. Although there are factors, such as shifted syllabic stress, which can make this less simple than the prepositional case, but it has been argued that there is no principled distinction between them ([Langacker, 1987. Taylor, 1995]). That it is a regularized, structural shift cannot be denied, and certainly one that relies on the configuration of the elements involved.

### 2.2.3 The complexity of syntax

We can compare the complexity of the language under various argument mapping or case marking schemes in terms of the number of symbols needed for a given level of expression, and evaluate these representations in terms of their economy of storage. Say that we wish to be able to represent a set \( N \) of objects and a set \( R \) of binary relations, using the terms necessary to represent the objects and predicate elements, and only the minimum necessary to ensure the interpretation of the argument mapping. By comparing the number of symbols a language user needs to

In the case of the completely non-configurational language above, \( \mathcal{L}_0 \), we need
only

\[ |L_0| = |N| + |R| \]  

(2.11)

total symbols. The language user in this case needs to store (remember) only one term for each object and each relation that could be mentioned. This base estimate is really only a reference point, since the best one can do with this language is talk about unordered, symmetric relations.

This works only for the symmetric relations; solutions for the asymmetric case must still be examined. Before going on the to configuration dependent case marking schemes outlined above, let us explore some other options.

**Extending the vocabulary**

One mechanism for representing the asymmetric predicates without requiring the use of configurational mechanisms is to extend the vocabulary of object terms such that they also represent the possible predicates. In the asymmetric relation 'on top of', we could, perhaps, posit another word, $a_c$, which means "a when $a$ is on top of something." We now need a second word for every possible object that could be on top of another. And, if we want to generalize this solution to all other asymmetric binary relations, we need to have an extra form of each object for each relation. I should note that this extension requires the new terms to be structurally unrelated to the existing ones, since we’ve already shown that a structural shift to create a new term, such as affixation, is really a configurational mechanism in disguise.

Using the alternate-object approach requires a set of new object terms $N^+$ whose size is the size of $N \times R$, which gives us $|N^+| = |N||R|$. A side effect of encoding the relation within the altered form of the object is that the explicit relation term becomes redundant in the expression. We have effectively Skolemized the binary relation, encapsulating the relation and one term into a new relational term which only requires one argument. If $a_c$ means ‘$a$ when $a$ is in relation $c$ with
Then the expression $a \epsilon X$ is equivalent to $a \epsilon c X$, and so we can eliminate $|R|$ redundant terms from the language. The functional denotations of these symbols can be written as the following lambda expressions:

$$
[a] \equiv a \\
[c] \equiv \lambda x \lambda y. (Z(x,y) \lor Z(y,x)) \\
[a, c] \equiv \lambda x. Z(a, x)
$$

With the additional terms required for the Skolemized objects, and the original set of unmarked object terms, the size of this new non-configurational form of the language $L_1$ is much larger than the original asymmetric case.

$$
|L_1| = |R||N| + |N|
$$

If we then attempt to extend beyond binary relations, to include more than two arguments, we must expand our vocabulary still further. For each argument we wish to add to the relations, we need another $|N||R|$ terms to express it with unique symbols. For $k$-place relations, then, we need a larger number of symbols:

$$
|L_k| = k|N||R|
$$

In order to express non-symmetric relations, the language user here needs to store a vastly greater number of terms for each object, on the order of the square of the base estimate.

**Generalizing marked objects**

Still considering only binary relations for the moment, we can generalize this alternate-object approach to a more economic scheme, that requires only twice the original number of terms. If we assign an arbitrary conceptual ordering of the roles for all the binary relations, then we need only have one special form for each object for each role, but not for each relation. The object given in special form can now be taken as the first of the ordered roles in the relation, while the normal form object fills the remaining role. This model of role assignment requires
2.2. **STRUCTURAL ORIGINS**

added processing on the part of the interpreter of this language, since each object term must now be assigned to the correct relational role, whereas the symmetric relations needed no assignment, and the Skolemized language encapsulated the assignment so that no further processing was needed. This scheme can be expanded to \( k \)-place relations by having \( k \) families of special forms corresponding to the \( k \) possible relational positions. So with \( k \) terms for each object, we have the size of this language \( \mathcal{L}_{\text{obj}} \) as given in eq. (2.15). This is a linear function of our base estimate for the asymmetric case, and perhaps not too great a tax on the mental capacity of the user. So by substituting a small amount of processing, we can reduce the size complexity of the language by a factor of \( O(|N|) \).

\[
|\mathcal{L}_{\text{obj}}| = k|N| + |R| \tag{2.15}
\]

We can see how this works through an example. Eq. 2.16 gives the interpretation of a group of object-denoting symbols, where the meaning of each is particular to the argument position it will play when composed with a predicate term. The example sentence (2.17) shows how these combine, irrespective of their relative positions or configuration, into the predication.

\[
[a_1] \rightarrow X \text{ as argument 1} \\
[b_2] \rightarrow Y \text{ as argument 2} \\
[c_3] \rightarrow Z \text{ as argument 3} 
\]

\[
[b_2 \quad a_1 \quad f \quad c_3] \rightarrow F(X,Y,Z) \tag{2.17}
\]

**Comparing case-marking systems**

Let us compare these systems to the configuration-based case marking systems outlined above, which are modelled after the processes appearing in actual languages. One factor we did not discuss is the question of how many case markers or case forms are really needed. One solution, which is very expensive in

25 February 2000
terms of the number of forms needed, is parallel to the mechanism used in $L_1$. For each term, we would generate a special case marker (either of the functional/prepositional or the morphological shift sort) which would mark each term specifically for each possible relation it could participate in. If we consider the marker needed to identify the object as a separate term, then this language, $L_{m1}$, needs as many terms for each object as there are predicates, resulting in a size of

$$|L_{m1}| = |N||R| + |N|$$  \hfill (2.18)

A much more economical scheme could be derived parallel to $L_{obj}$, in which only as many special forms are needed as there are separate argument positions in the predicate. Under this scheme, we would use one prepositional marker or one type of morphological shift for each differentiated argument position, regardless of the predicate used. Since we do not have to generate a term for each predicate, we only need as many extra items as possible arguments, resulting in the smallest language size yet.

$$|L_{case}| = |N| + |R| + k$$  \hfill (2.19)

This case marked language, $L_{case}$, is very similar to $L_{obj}$, in which each term was multiplied into $k$ forms for the argument positions. However, here in $L_{case}$, we do not need to generate $k|N|$ new terms, but only need to introduce a total of $k$ new terms to indicate argument positions. By simply juxtaposing the markers with the object terms, we can identify them as assigned to the proper place in the interpreted relation. This of course requires the the interpretive process is able to compose these adjacent pairs into the proper type of intermediate structure, which is a more complex type of processing than was required by the $L_{obj}$ or especially the $L_0$ language.

$$[a \ \ m_1] \rightarrow X \text{ as argument 1}$$
2.2. STRUCTURAL ORIGINS

\[ [b \ m_2] \rightarrow Y \text{ as argument 2} \]
\[ [c \ m_3] \rightarrow Z \text{ as argument 3} \]
\[ [f \ b \ m_2 \ a \ m_1 \ c \ m_3] \rightarrow F(X, Y, Z) \] (2.20)

Complete configuration

In the end, if we rely completely on configuration, we needn’t add any new terms at all. By making the interpretation of each predicate dependent on the order of its argument terms, we can use the same object terms, with no modification or marking, regardless of which argument position they are intended for. In the simplest mapping, we could have the first object to appear be used as the first argument, the second object as the second argument, etc. This language, \( \mathcal{L}_{con} \), has the same vocabulary size as the original asymmetric language \( \mathcal{L}_0 \).

\[ |\mathcal{L}_{con}| = |N| + |R| \] (2.22)

Interpreting examples of \( \mathcal{L}_{con} \) requires only trivial processing, assigning the objects to argument slots in the order they appear.

\[ [f \ b \ c \ a] \rightarrow F(Y, Z, X) \] (2.23)

Comparing the sizes in succession (eq. 2.24) is quite dramatic, showing that an enormous degree of vocabulary storage efficiency is gained through using configurational mechanisms.

\[ |\mathcal{L}_1| = |R||N| + |N| \]
\[ |\mathcal{L}_{mk}| = k|N||R| + |N| \]
\[ |\mathcal{L}_k| = k|N||R| \]
\[ |\mathcal{L}_{obj}| = k|N| + |R| \]
\[ |\mathcal{L}_{case}| = |N| + |R| + k \]
\[ |L_{conf} | = |N| + |R| \]  

(2.24)

### 2.2.4 Sentential interference

Focussing for a moment on the non-configurational solutions, it is clear that there are deeper problems than relative argument assignment. Even if we assume the efficient case marking morphology, many problems remain in the asyntactic language \( \mathcal{L} \). A particularly difficult question among them is the problem of delimiting relations. So far, we have dealt only with cases having one predicate along with the objects participating in it. What happens when a speaker wants to mention more than one relationship? Some mechanism is needed so that the various relationships are kept separate in interpretation, and that the components and specifiers of one relationship do not get mixed up in those of the other. This division, although somewhat ill-defined, is normally attributed to the clause or sentence level of syntactic analysis.

The most obvious way to define a sentence in \( \mathcal{L}_{case} \) would be to have objects participate in the closest relation for which the role is not filled. While this mechanism does not enforce any strict arrangement on the objects, it is still a configurational constraint, in that it will not allow arbitrary ordering of the terms involved. If each object term is associated with the relation closest to it in the sequence of symbols, then, if two relations are intermixed heavily enough, the object will participate in the other relation, and not in the one for which it was intended.

Our language will not be able to support multiple relations under non-configuration using either the multiplied or the case-marked object terms (\( \mathcal{L}_{obj} \) or \( \mathcal{L}_{case} \)). Since neither of these forms carry information about which relation they are to take part in, the possibility of confusion always exists when more than one relation is used. Because the Skolemized versions of the object terms carry the identity of their relation with them, multiple relations using this mechanism will
not become confused, at least up to a point. When reference needs to be made to two relations of the same type, however, it will no longer be able to differentiate between the objects of those two. And still, this solution carries the enormous size burden of requiring a separate form of each object term for each relation that is expressed within the language (not simply within a single statement, as with the case-marked version).
Chapter 3

The Associative Model

The goal of this work is to model the interactions between words and constructions. We are not investigating larger grammatical concepts, such as sentence structure, clausal boundaries, verb valency phenomena, etc. What we are interested in are questions about the relationship between the syntactic environment and the words used to comprise it:

• for a particular construction, which are the words that can be used within it?

• Are there preferences toward a particular word, or class of words?

• is there a class of words which can be freely substituted in a particular context?

• is there a class of contexts which all allow the same set of words?

• How much variation is allowed in the construction while maintaining the same word choices?

• is there some other class of constructions that allows the same set of words?

• is a word or construction naturally a member of only one class, or is the behavior better explained by membership in many classes?
The nature of lexical and syntactic interactions discussed in the previous chapter provides the fundamental axioms for the construction of a model of word use that directly incorporates these constraints on interaction.

This chapter outlines a rationale and a methodology for constructing such a catalog of constructions and word types.

### 3.1 The form of the associative model

The model developed in this chapter is one which captures the *interactions between words* in the language. In particular, the focus is on the interplay between a word and the constructions it appears in. Using those associations between words and constructions, one can

- find classes of words with similar usage, and their associated syntactic behavior
- identify the separate modes of use a single word might exhibit, which can aid in sense identification and disambiguation
- make judgements about the grammaticality of using a word in a particular construction,
- or predict word occurrence in a given context.

#### 3.1.1 Association

**Association between word and context**

What we are interested in knowing about the behavior of these words and contexts are their *associative properties*. Each word, due to its semantic properties and syntactic constraints, will be able to participate in only a limited set of contexts. Without specifying the nature of these constraints governing the interaction, we can form a model of the associations in the occurrence of words and contexts...
3.1. **THE FORM OF THE ASSOCIATIVE MODEL**

Together. This model will allow us to predict the set of contexts that license the use of the word, and conversely, the set of words which are licensed by a given context. It is these observable associations that we seek to discover.

In order to address these questions we first need to define what it is we mean by the terms *word* and *context*. Since we are using written text for our corpus, our definitions will be relative to that representation; however, the methods and insights should be equally applicable to other linguistic expressions, such as phonetic transcriptions or voice recordings.

**Definitions**

The written corpus consists of a sequence of words, delimited by whitespace and punctuation. These linguistic symbols have an implicit linear ordering, which enables us to label them with the natural numbers $1, 2, 3, \ldots$ (eq.3.1).

$$
corpus \equiv [t_1, t_2, \ldots, t_{i-2}, t_{i-1}, t_i, t_{i+1}, \ldots, t_N] \quad (3.1)
$$

Any individual word or symbol used in this sequence will be connected to the use of the other symbols, through the web of grammatical, semantic, and other linguistic mechanisms operating between them. Thus, in order to represent the entire dependency between a word, $w$, and its use in the sequence, we need to take into account the entire remainder of the sequence as the linguistic construction that affects its use. If the word $w$ appears as corpus term $t_i$, then the construction which determines its use is:

$$
c \equiv [t_1, t_2, \ldots, t_{i-2}, t_{i-1}, w, t_{i+1}, \ldots, t_N] \quad (3.2)
$$

However, we argued in the previous chapter that the kinds of interactions which most affect the choice of word at $t_i$ are necessarily those used in proximity to it. It would seem sensible to investigate only that local region of the context which will affect our study. Other statistical studies of word associations also use the local region, but typically use only a *very small region* of the local context to
condition their model of word use. Markov modeling techniques are based on very short \(n\)-grams, and typically use only one or two (!) words to the left to model the influencing factors ([?general-n-gram-techniques?]!). Other work has been done using variable length left contexts, but does not involve any generalization of those long contexts.([?pereira-suffix-trees?]!). Some work is limited to only noun-verb relationships within single sentences ([?greffenstette?, Hindle and Rooth, 1991, Brown et al., 1992]), but does not extend to larger context frames.

Harris, as one of the founders of this sort of co-occurrence study ([?harris-structural-linguistics?]), extends the notion of effective context to the utterance, “any stretch of talk, by one person, before and after which there is silence.” This definition will be very difficult to generalize to text, however.

In general, it would seem reasonable to limit the study of the total context to only that portion in some reasonable proximity to the index word. This local context can be represented by the \(n\) words preceding and the \(m\) words following the index word \(w = t_i\). This \(n,m\)-context is written:

\[
c = [t_{i-n} \ldots t_{i-1}, \quad \cdots \quad t_{i+1}, \quad t_{i+m}]
\] (3.3)

For simplicity, we can re-index the context to make it independent of its original position in the corpus, and more practical for comparison to others:

\[
c = [t_{-n}, \ldots, t_{-2}, t_{-1}, \quad (t_0 \equiv w), \quad t_1, \quad t_2, \ldots, \quad t_m]
\] (3.4)

The aim here is to use \(n, m\) large enough to encompass interesting lexical interactions. But, it is impractical to extend it such that every interaction is captured. Given that the average length of a sentence in the corpus used is \(\approx 20\) words, and consists of two clauses, four or five words to each side will capture almost all of the clause, and three of four will capture most of it.
3.1. THE FORM OF THE ASSOCIATIVE MODEL

Association weights

The model we seek will represent the associations between the use of a words and the contexts they appear in, in order to show which combinations are acceptable, typical, unacceptable and so on. One way to represent these associations between words and contexts is to use a set of weights to describe the affinity between them. Let us say that for each pair of word and context, \((w, c)\), we assign a real number which will be their strength of association, much as in word-word association studies (cf. [Hindle and Rooth, 1991], for example) or word-context \(n\)-gram probabilistic models (e.g. [Church and Hanks, 1990, ? others?]). Greater weights will indicate a high correlation, and smaller weights show that they rarely co-occur. For instance, a context representing a fixed idiom that allows no grammatical alternation will have a high association with the word completing the saying. Combinations of word and context that are ungrammatical, or nearly uninterpretable, will have very low, possibly zero, weights.

This relationship between the word and context is not symmetric, though, and we will need to separate the association into two functions. One, \(w(c)\), will measure the affinity a word has for a context, while the other, \(c(w)\), will indicate the preference a context has for a word.

\[
w_i(c), \quad \text{association of word } w_i \text{ with a context} \tag{3.5}
\]

\[
c_j(w) \quad \text{association of context } c_i \text{ with a word} \tag{3.6}
\]

We can see the discrepancy between the two through some simple examples. In (3.7), we see a word, the, used in a typical context. We would normally assign the weight \(c_1(w_{the})\) a fairly high value, since the combination is a common saying as well as being a grammatical construction. There are a few alternatives for the here (my, her, etc.), but the choices are not widely varied.

\[
c_1 = \text{don't rock the boat} \tag{3.7}
\]
\begin{equation}
\begin{align*}
c_2 &= \text{don’t rock the boat}
\end{align*}
\end{equation}

On the other hand, the word-based association weight \( w_{the}(c_1) \) should not be nearly as large as the context weight. Since there are an enormous number of alternative contexts for \( the \) to appear in, and there is no reason to prefer this context in specific, the assigned weight should not be extraordinarily high. We should like the values of \( w(c) \) to reflect the affinity that the word has for the given context. In the next example (3.8), we focus on the appearance of \( rock \) in the same phrase. Even though the co-occurrence counts for these two combinations (3.7) and (3.8) will always be precisely identical, we should expect that the weight \( w_{rock}(c_2) \) will be greater than \( w_{the}(c_1) \), because \( rock \) will have a lesser variety of alternate contexts it could appear in.

### 3.1.2 The Basic Model

These examples make it clear that we need to separate the two aspects of the association between word and context. We need to be able to view the use of a word and a construction independently as well as jointly. We have three properties to be concerned with: the joint use, or co-occurrence, of the word and context; the use of the word irrespective of the context or expression it is contained in; and the use of a construction, regardless of the word used to fill a particular role. Comparing these three different properties of usage will enable us to identify the set of contexts which license a word, separately from the set of words which are allowed in some given context.

In order to understand the word’s preference for appearing in some contexts, and for not appearing in others, we need a representation that reflects its use in a context relative to its global use in any context. The word’s preference for any context can only be expressed with respect to its use in other contexts.

For constructions, in order to understand any preference for a word in a particular context, we need to be able to contrast it with all the other possible words.
that might fill the given role in the context.

These two views of the relationship between word and context give us the basis for a model of word use. This *associative model* expresses the relations between words and constructions in usage, in such a way that the affinity between words and contexts, and the constraints that permit or deny their co-occurrence, become visible.

**The observable relationship**

The model we want to acquire should ideally give the *expected value* for the functions \( w(c) \) and \( c(w) \). This mythical expected beast is somewhat akin to the notion of *linguistic competence*—the ability of a speaker to identify and distinguish those word combinations allowed in the language. We are not interested, however, in any one individual’s competence, but in some conglomerate linguistic ability, averaged and spread out over a population of speakers.

The problem remains, though, that we cannot, for various reasons, either truly get inside people’s heads to inspect their ability, or place a great deal of trust in their own judgements of their ability. We do, however, have a great deal of information produced as a *side-effect* of that language ability. And we can place our faith in this measurable, observable reflection of the linguistic knowledge of the speakers. Corpus data gives us a resource for examining the *output* of that inherent linguistic ability. The record of actual, rather than hypothesised, linguistic behavior, gives us positive examples of the well-formed construction. Building an explicit model of these occurrences, and later abstracting that to a class-based generalization, will give us a realistic, analytic and predictive model of the interactions between words and constructions.

**Word association**

\[
 w_i(c), \quad \text{association of word } w_i \text{ with a context} \tag{3.9}
\]
$w(c)$ is the word $w$’s association with the context $c$. It represents the affinity of $w$ for $c$, with respect to all of the possible uses of $w$ across all contexts. This means that contexts which have a high association with, and appear often with the word $w$ will have relatively higher values of $w(c)$ than contexts in which $w$ appears infrequently. Contexts for which $w$ is grammatically prohibited from use should have a value of 0.

Going back to our last example (3.8), $w_{\text{rock}}(c_2)$ for $c = “ \text{don’t rock the boat}”$ would be smaller than the value for $c = “ \text{rock music}”$, because the latter is a more prototypical use of $\text{rock}$, and is likely to be used more often. But compared to $w_{\text{the}}(c_1)$ for $c_1 = “ \text{don’t rock the boat}”$, $w_{\text{rock}}(c_2)$ would be larger due its higher degree of prototypicality – the phrase is more specific to $\text{rock}$ than to $\text{the}$, which appears in a much larger variety of contexts.

We can use the word’s predictive or conditional expectation of the context as a good approximation for the desired behavior of $w(c)$. The expected probability $p(c = c_i | w = w_j)$ gives us the likelihood that, given we know $w_j$ was used, it appeared in context $c_i$. This expected probability\footnote{Using the probability of occurrence as a model of grammaticality does concede to some drawbacks, however. For instance, it may be more likely that someone will stutter out “the...the groceries” than use the phrase “the oomorphogenic imperative”, even though the latter is grammatical in a strict sense, while the former is not. In addition to the probability that a combination will be used, we perhaps need some mapping to some other indication of grammatical acceptability.} has the desired characteristics: a high value for prototypical contexts, low values for uncommon combinations, and zero values for combinations that will not appear in language due to grammatical constraints.

We define the word association (3.10) of a word $w_j$ for a context $c_i$ as the expected probability of the combination $(c_i, w_j)$ where the index word $t_0$ of $c_i$ is $w_j$, evaluated with respect to the space of all possible context sequences $C$.

$$w_j(c_i) \overset{\text{def}}{=} p(c = c_i | w = w_j) \quad \text{for some context-word pair } (c, w) \quad (3.10)$$

We can use the shorthand in (3.11) to express the total behavior of the word, in relation to the entire set of possible constructions. This representation of the word’s selectional restrictions expresses all of our knowledge of the use of this
word, in terms of its association to the constructions it is allowed (or not allowed) to participate in.

\[
w_j(C) \overset{\text{def}}{=} [w_j(c_1), w_j(c_2), \ldots, w_j(c_{C_{n,m}})]
\]  

(3.11)

**Context association**

\[
c_j(w) \quad \text{association of context } c_j \text{ with a word}
\]

(3.12)

c(w) gives the context c’s association with w, with respect to the appearance of any possible word within a fixed c. So c(w) will be large for “don’t rock the boat”, and smaller for other word choices, such as “don’t tip the boat”, or “don’t paint the boat”, because we have a greater expectation of rock in this construction. These other possibilities must still be represented as grammatically acceptable, although their lower expected occurrence should be reflected in a smaller value. Words restricted from this context should have a zero value; for example \(c(\text{restricted}) = 0\), (* “don’t restricted the boat”).

\(c(w)\) is very much like the predictor functions used in n-gram models. It gives the expected occurrence of \(w\) in the lexical environment specified by \(c\). As such, it represents the grammatical constraints imposed by the context, and any word used to fill the context must either play the role, or be coercible into playing the role, determined by those constraints.

We define \(c_i(w_j)\) (3.13) similarly to \(w(c)\), as the expected probability of occurrence of the word \(w_j\) as the index word \(t_0\) of \(c_i\), evaluated with respect to the space of all possible words:

\[
c_i(w_j) \overset{\text{def}}{=} p(w = w_j|c = c_i) \quad \text{given a context-word pair } (c, w)
\]

(3.13)

Again, we can use the vector shorthand (3.14) for the expected distribution over the possible appearance of all words in the entire vocabulary. This represents the
constraints projected by this construction onto the entire space of possible word choices.

\[ c_i(W) \overset{\text{def}}{=} [c_i(w_1), c_i(w_2), c_i(w_3), \ldots, c_i(w_V)] \]  (3.14)

### 3.1.3 The Class Model

These two association functions \( w(c) \) and \( c(w) \) (eq. 3.10 & 3.13) define the skeletal form of the associative model. The goal of acquisition will be to find parameters such that we can evaluate \( w(c) \) and \( c(w) \) for any pair of \( c \) and \( w \).

**Model size** Estimating these weights directly is an enormous task. One of the benefits of using a class abstraction is a simplification of the model itself. As we will discuss in the next chapter, learning the explicit values of \( w(c) \) and \( c(w) \) is difficult, if not impossible, due to the enormous amount of data needed. Reducing the size of the model aids both in its utility as a predictive tool and a descriptive abstraction.

The explicit model is comprised of a vast number of parameters. If we take \( V \) as the number of words, and \( C = V^{n+m} \) as the number of possible contexts of size \( n, m \), we need \( 2V^{n+m+1} \) explicit parameters for \( w(c) \) and \( c(w) \). With a reasonable vocabulary size (English is \( O(10^5) \)), and especially with the size of context we have been considering here (say, \( n, m \geq 3 \)) this is completely impractical (this estimate gives \( O(10^{35}) \) for a model of English!).

If we instead use a set of lexical and context classes, \( L \) and \( \Gamma \), the size of the model can be reduced tremendously. If we take \( l \) as the number of lexical classes, and \( k \) as the number of context classes, we need only \( 2lk \) parameters to express the association weights between the classes. Assuming the the number of classes is far less than the number of original elements, this results in a number of parameters for the class model far below that needed for the explicit model. Even if we only reduce the number of context classes so that \( k \approx O(V) \), we now only
require $O(V^2)$ parameters, as opposed to $O(V^7)$. This reduction aids tremendously in the storage and manipulation of the model, as well as reducing the amount of corpus data necessary for training.

We still require an additional $lV$ parameters for the lexical memberships, and potentially $kV^{n+m}$ for the context class memberships. However, I will show later (chs. 4 and 5) how a functional representation of context membership, using structural similarity, can be used rather than an explicit listing, reducing the size of the context class representation enormously.

**Lexical and syntactic classes**

It is by reducing the model to a class-based descriptive tool that we can begin to see clearly the relationships between words, and between words and their lexical environments.

Classes of words and contexts based on grammatical behavior will provide a good and useful descriptive generalization over the possible combinations. Rather than use the model over specific instances — for example to identify the words that can be used in the context (3.15) — we can approach the interaction in a more general way, and ask whether the word can appear in the subject position of a *report*-type verb, as in the generalized context shown in (3.16).

\[
\text{[the } \underline{\text{}} \text{ announced yesterday that]} \text{ (3.15)}
\]

\[
\text{[DET} \underline{\text{}} \text{ [V}_{\text{report}] [ADV}_{\text{time}] [CLAUSE\_MARKER]} \text{ (3.16)}
\]

The licensed set of words will be about the same for the two, but the latter type of context class description has broader applicability and scope. Consider the problem of acquisition — if we are trying to find instances of the class of words defined by the explicitly defined context (3.15), then we can only count words appearing in that context exactly. If we want to find words of the syntactic class defined by (3.16), then we potentially have a much richer set of examples in the
same corpus. And the words found in the context \((3.16)\) will also be licensed in \((3.15)\), even though we haven’t observed the co-occurrence in the corpus.

Using a behaviorally defined class of contexts in this way encompasses many similar questions about usage and co-occurrence, and provides a level of reasoning about grammatical behavior not available with discrete instances.

Exactly this same idea underlies most grammatical theories. Instead of explicitly writing out each and every combination of words that is allowable, they take advantage of the families of properties that are held in common among groups of words. Using lexical features or class membership to express constraints and allowed constructions is far more compact and interpretable. Sensibly, one states rules such as \(N' \to \text{Adj} \ N\), and not \textcolor{red}{\text{red boat}} \to \textcolor{red}{\text{red boat}}. The latter, however, is essentially what is done in much statistical modelling.

**Classes**

Let us define a class for present purposes as follows: a class is composed of a set of elements \(x_i\) and a membership function \(k() : x_i \mapsto [0, 1]\), each element having a membership value \(k(x_i)\) in the class in the range \([0, 1]\). Elements with a membership value different from 0 are said to be members of the class.

\[
\text{a class } k \text{ is comprised of} \quad X = x_1, x_2, \ldots
\]
\[
\text{a set of elements} \quad k(x_i) \mapsto [0, 1]
\]

We need to define two sets of classes: \(L\), the lexical classes, and \(\Gamma\), the context classes, ranging over all possible words and all possible contexts, respectively:

\[
L = \ell_1, \ell_2, \ldots, \ell_x
\]
\[
\Gamma = \gamma_1, \gamma_2, \ldots, \gamma_y
\]
3.1. THE FORM OF THE ASSOCIATIVE MODEL

We also need membership functions to identify which words and contexts are members of which classes. We will use these memberships to express the probability that an occurrence of the class is the item in question.

\[ \ell_i(w_j) \text{ membership of } w_j \text{ in class } \ell_i \] (3.20)

\[ \gamma_i(c_j) \text{ membership of } c_j \text{ in class } \gamma_i \] (3.21)

**Class association**

We need also define the association functions between lexical and contextual classes. We can do this by direct analogy with the associations defined between words and contexts. The following list gives the class-based functions, and the non-class counterparts from which they derive.

**lexical association:** \[ \ell_j(\gamma_i) \leftarrow w_j(c_i) \] (3.22)

**context distribution for a lexical class:** \[ \ell_j(\Gamma) \leftarrow w_j(C) \] (3.23)

**context association:** \[ \gamma_i(\ell_j) \leftarrow c_i(w_j) \] (3.24)

**lexical distribution for a context class:** \[ \gamma_i(L) \leftarrow c_i(W) \] (3.25)

Each class association function gives the conditional expectation of the appearance of a member of a class in the counterpart set. For example, \( \gamma_i(\ell_j) \) is the probability that a member of the lexical class \( \ell_j \) will be used, given an appearance of a context in the class \( \gamma_i \). We will define these explicitly in terms of the word and context probabilities in the later sections on category acquisition.

3.1.4 **Class membership and variation**

The assignment of membership values to the words and contexts that comprise each class is obviously fundamental to the model. In assigning words and contexts
to the various classes, what we seek to do is form a valid generalization over the behavior of the individuals – the associations for the classes should correspond to the associations of the members. If, for instance, if \( w_j(c_i) \) is high (i.e. \( w_j \) selects for the context \( c_i \)) then the class membership should be such that \( \ell_j(\gamma_i) \) is also high if \( w_j \in \ell_j \) and \( c_i \in \gamma_i \). Finding the appropriate division of the space of words and contexts to the classes is the focus of much of the rest of this work.

Each class encompasses a set of behavioral properties of a group of words or contexts. Since we are concerned with the co-occurrence and associational behavior of the items, but not necessarily their identity in any other sense, it seems clear that we need to allow for a word or context to be a member of multiple classes. A given word might express many different grammatical behaviors, all of which are distinct, and these should not be collapsed into a composite description of the total behavior of the word.

This type of partial or fuzzy kind of class assignment is also typical with traditional grammatical classes. Consider a word such as ‘purchase’, which can take many grammatical roles. In one major role, it can act as a transitive verb, for which it can take a variety of different selectional frames:

\[
\begin{align*}
\text{[AGENT]} & \quad \text{purchase}_{\text{V}} \quad \text{[PATIENT]} & (3.26) \\
\text{[AGENT]} & \quad \text{purchase}_{\text{V}} \quad \text{[PATIENT]} \text{ for [AMOUNT]} & (3.27) \\
\text{[AGENT]} & \quad \text{purchase}_{\text{V}} \quad \text{[PATIENT]} \text{ from [POSSESSOR]} & (3.28)
\end{align*}
\]

Although these transitive uses all have somewhat different syntactic expectations, this range of behavior is shared as a whole with other words. Because they all exhibit this range of usage, we can identify a category of ‘transitive exchange verbs’: \{buy, sell, purchase, acquire, exchange, \ldots\}.

However, in describing this usage of ‘purchase’ as a verb, we do not want to limit ourselves from describing conflicting uses of the same word. Traditional analysis, using parts of speech or other categorial labels, allows separate occurrences of a given word to carry different grammatical labels.
the purchase\textsubscript{N} of \text{[PATIENT]} 
(3.29)
the purchase\textsubscript{MOD} price 
(3.30)

The word 'purchase' takes other very different grammatical roles, which we will certainly want to put in separate lexical classes, each with its own set of possible contexts. 'Purchase' can be used as an event noun, as in example (3.29), where the 'of \text{PATIENT}' postmodifying phrase may or may not be present. 'Purchase' is also used as a prenominal modifier in constructions such as (3.30) – 'purchase price' or 'purchase warrant'.

Having a set of classes with explicit real-values membership functions gives us the ability to divide these behaviors into separate classes, in order to preserve the clarity of the class abstraction. Some of the other 'transitive exchange' terms, while clearly in the same verbal behavior class as \text{purchase}, do not share the other nominal or modifier behavior. Consider \text{sell v. sale}, \text{acquire v. acquisition}. Instead of conflating the description by using only one class, we will want to divide these behaviors into separate categories. One class will have as members those words which exhibit the verbal behavior, and another class will have those which act as nominals. The word \text{purchase} will simply be a member of both, since it exhibits both associations.

\textbf{Behavioral, not enumerative, types}

As opposed to many lexical categorization schemes, which seek to enumerate the lexicon into distinct and separate groups of words, the set of types $L = \ell_1, \ell_2, \ldots$ is really an attempt to create a set of \textit{lexical behavior types}. Rather than segmenting the vocabulary into groups, for which each word would be a member of only one group, we need to segment the \textit{behavior} of the words, such that each word will be a member of those classes with which it shares grammatical behaviors, as
defined by its associations with contextual classes. Each class then defines a type, or perhaps a group of types, of grammatical behavior that are associated with the member words. Each word can reasonably be expected to appear in the contexts which constitute that behavior.

The same need for multiple membership arises for contexts within context classes. A grammatical context, having a matrix of constraints resulting from the interactions between a number of words, may be far more restrictive in the variation of usage allowed than a single word is, but will still permit some variation. Consider, for instance, a context calling for some kind of prenominal modifier, such as

$$\text{with another } \_\_\_\text{ investment}$$

(3.31)

There are many terms that could be used in the key position. Some of these are used, in general, only as modifiers: 'industrial', 'great', or 'bad'. Some terms can also be used in other, more noun-like contexts: 'petroleum' or 'telecommunications'. Some terms may have a very broad spectrum of uses, for instance 'sound' or 'failing'.

There are a variety of different behaviors possible with this context sequence. It shares some selectional properties with contexts that specify certain classes of adjective, others that specify a type of business-oriented noun, and still others that are less restrictive. There will exist other contexts that select for these types of words independently of the other words, defining separate categories of context behavior. Our original context can be said to participate in a large range of these behaviors. To account for this, we need to allow contexts as members of multiple categories.

3.1.5 Implications

Classification of words and constructions into associational categories serves two important functions.
Abstract description

Firstly, it raises the level of description from the vast numbers of individual occurrences to a more abstract, more comprehensible level. At this level, linguistic behavior is understood not as an unwieldy collection of particular facts, but as the generalized set of relationships between types of words and the constructions they participate in. By grouping constructions and words into classes which have common functional behavior, we can abstract away from any idiosyncratic behavior of individual words, and begin to see these more general patterns of behavior. Having a lexicon or grammatical model which only models interactions with respect to individual words, and has no internal structure, does not gain us much understanding of the real properties of language.

We have already noted that words are not used completely independently, but occur in families of similar function. It is these family similarities that we capture in the class level of the associative model. The classes provide a level of description which encompasses the interactions between these families of functional types, and their expression in the syntax.

In the previous chapter I reasoned that, because of the large correspondence between actual combinations of words seen in use and the meaning relationships among them, the lexical and grammatical constraints acting between words could not be completely arbitrary. The properties that control the interaction will occur across many words, and the various combinations of words comprising grammatical constructions will also contain recurring sets of constraint properties. It is the side effects of these constraints, as expressed in the grammatically allowed (and prohibited) combinations of construction and word, that we will capture when forming these behavioral classes.

This level of description offered by classes, and the operations available only at class level, provides the connection to the theory underlying the model. Knowing \( c() \) and \( w() \) for any particular word or context is certainly interesting in and of
itself, but it will not provide much insight into the more general structure of language without being able to compare words or contexts against one another. Word classes (such as the familiar part-of-speech labels, or other grammatical labels, e.g. verb phrases, \( \bar{N} \)-constructions, appositives, etc.) provide the vocabulary for discussing language behavior for understanding the underlying properties of the language.

Utility

Secondly, and of immediate practical importance, the categories allow us to extend the predictions of the model to combinations of word and context which have not been observed while learning the model. This is a very real concern, due to the fact that we rarely have a complete set of observations, covering all the allowed combinations for any particular word or construction (cf. ch. 4). This means that we can rarely claim complete knowledge of a word or construction.

However, if we have enough usage evidence about a word to justify its membership in one or more lexical classes, we can extend our predictive inferences to include combinations of its containing class, \( \ell_i \), with constructions belonging to the projection of that class, \( \ell_i(\Gamma) \). It follows from the inclusion of the word into the class that the properties of the class can be used to describe the word.

In this way, we no longer need to have examples of all the possible allowed combinations of word and context – by generalizing into classes of constructions and words, we simply need enough examples to cover some representative set of interactions, and we can use the generalized class behavior to model that of individual words and contexts.

This kind of predictive inference through \textit{grammatical analogy} is the most powerful feature of a class-based grammatical model.
3.2 Simple Acquisition

The goal of the acquisition is to find parameters such that we can evaluate \( w(c) \) and \( c(w) \) for any pair of \( c \) and \( w \). Knowing the values of the associative functions \( w() \) and \( c() \) for any word or construction will obviously provide a great insight into the structure and use of the language, both directly, and through the powerful abstraction afforded by classes based on them. Since these functions model the use of words combined in constructions, we can attempt to acquire their values through the observations of real usage, thorough corpus data.

The remainder of the chapter discusses methods for acquiring values for these functions that follow reasonably from observed usage data. We will start with the most basic model, a maximum likelihood model built directly from corpus co-occurrence frequencies. This type of model has a number of well-known deficiencies, due to the simple fact that there is never enough corpus data to cover all the possibilities. Despite this, it can serve as the first step in building a model closer to the ideal, a starting point from which we can derive one that is better, more accurate, and closer to the linguistic reality.

3.2.1 Maximum likelihood estimates

Earlier, the functions \( w(c) \) and \( c(w) \) were defined in terms of probability of occurrence (eqs. 3.10 and 3.13). We can create a simple estimate of these probabilities by counting the frequency of co-occurrence over a large sample of text.

The maximum likelihood estimate (MLE) of the context association function is simple that estimate which maximizes the chance that the sample data is derived from the estimate. It turns out that this is simply the frequency of the target word with respect to the context.

This frequency-based estimate of the context association function \( c_j(w_t) \), given in (3.32), is the count of the word \( w_j \) within the context \( c_t \), normalized with respect
to the total number of occurrences of the context.

\[
\text{MLE } p(w_j | c_i) \overset{\text{def}}{=} c_i(w_j) = \frac{\|\{c_i | t_0 = w_j\}\|}{\|c_i\|} \tag{3.32}
\]

We can create an estimate for the word association functions in the same way, normalizing the co-occurrence counts by the total occurrences of the word.

\[
\text{MLE } p(c_i | w_j) \overset{\text{def}}{=} w_i(c_i) = \frac{\|\{c_i | t_0 = w_j\}\|}{\|w_j\|} \tag{3.33}
\]

**Probabilistic interpretation**

It is easy to see this is a probability density – since we have normalized by the total number occurrences of the context, this frequency distribution \(c_i(W)\) sums to 1, and we can treat it as a probability density. These simple frequency counts give us the maximum likelihood estimate of the co-occurrence probabilities w.r.t. the corpus.

\[
\sum c_i(W) = \sum_j c_i(w_j) = \sum_j \frac{\|\{c_i | t_0 = w_j\}\|}{\|c_i\|} = \frac{1}{\|c_i\|} \sum_j \|\{c_i | t_0 = w_j\}\| = 1 \tag{3.34}
\]

**3.2.2 Representational duals: \(c(W)\) and \(w(C)\)**

These two representations, taken together — the context frequency of words, and the word frequency of contexts — give us an interlinked dual model of the associative properties of the language, from which we can begin to uncover the classes of common occurrence properties, and to decipher the constraints governing these associations.

By reframing the language usage data, we can see that the two sets of vectors, \(c(W)\) and \(w(C)\) are dual representations, representing identical information.
in different ways. If we write the co-occurrence data as a very large matrix, with contexts indexed as the rows, and individual words identifying the columns, we obtain the count matrix \( \mathbf{A} \) shown in eqn. 3.35. Each element \( a_{i,j} \) in the matrix is the number of times the particular context \( c_i \) and word \( w_j \) have been used together.

\[
\mathbf{A} = \begin{pmatrix}
a_{1,1} & a_{1,2} & \cdots & a_{1,W} \\
\vdots & & & \vdots \\
a_{C,1} & \cdots & a_{C,W}
\end{pmatrix}
\]

\[ a_{i,j} = \| \{ c_i | t_0 = w_j \} \| \] (3.35)

The word association vector, \( w(C) \), is equivalent to a normalized column of the matrix \( \mathbf{A} \). We defined \( w(C) \) (eqn. 3.11) as the collection of elements \( w_j(c_i) \) (eqn. 3.33), where each \( w_j(c_i) \) is the absolute count of \( w_j \) occurring within the context \( c_i \), normalized by the total occurrence of \( w_j \). The \((i,j)\)-th element of the count matrix, \( a_{i,j} \), gives the same count of occurrences of \( c_i \) with \( w_j \). The total number of occurrences of \( w_j \) is simply the sum of all elements in the \( j \)-th column, \( a_{.,j} \), so we have \( \| w_j \| = \sum_i a_{i,j} \). Knowing this, the relationship between the \( j \)-th column of \( \mathbf{A} \) and the context distribution \( w_j(C) \) is simply:

\[
w_j(C) = a_{.,j} / \sum_i a_{i,j} = [a_{1,j}, a_{2,j}, \ldots, a_{C,j}] / \| w_j \| \] (3.36)

Similarly, the word frequency distributions, \( c_i(W) \), can be seen to be the rows of \( \mathbf{A} \), \( a_{i,.} \), normalized by the number of occurrences of each context, \( \| c_i \| = \sum_j a_{i,j} \).

\[
c_i(W) = a_{i,.} / \sum_j a_{i,j} \] (3.37)

Either of the frequency distributions \( c(W) \) or \( w(C) \) can be used to represent the entire set of occurrence data. They represent different points of view on that data, since they are normalized independently, with respect to the counts of the context or the word.

25 February 2000
3.2.3 \textit{n-gram models}

The functions $q_1(W)$ constructed so far clearly form a kind of \textit{n-gram model} ([Church and Hanks, 1990]). An \textit{n-gram model} is a probability-based predictor model, in which the identity of some sequence of $n - 1$ words is used to predict an $n_{th}$ word. In this case, specifically, the predictor words are the $n + m$ words, $t_{-n} - t_m$ of the $n, m$-context, and the predicted word is the index word $w = t_0$. The estimates we have derived directly from the co-occurrence counts of context and word form what is known as the \textit{maximum likelihood estimate} (MLE) of the co-occurrence probabilities. This is the estimate that maximize the the probability of the source data under the model. However, this training data doesn't necessarily contain the wide variety of words usage and constructions that are representative of the language as a whole. The goal of most work in statistical language modelling, and the goal of the remainder of this chapter, is to transform this observed estimate into a generalized, abstracted model that more accurately reflects the overall behavior of the language.

\textbf{Other predictors}

The context distribution of a word, $w_i(C)$, is also a predictor function – the context is now being predicted by the word. This lexically centered view is not often used as a statistical language model itself, although comparisons using the context distribution for small sets of words is used in the distributional similarity methods described below (cf. §??).

Lexically-centered grammatical descriptions, such as LTAGs ([\texttt{ltags}]), supertags ([\texttt{srini-supertags}]) or link grammar ([\texttt{link-grammar}]) are based this notion of lexically encoded restrictions on the syntactic environment, but these representations typically do not use corpus acquisition or statistical modelling in their formation. $w(C)$ encodes the local selectional restrictions of the word, and so work on describing or acquiring selectional frames (such as [\texttt{grishman+sterling}];

72
brent:subcat?]) is related. I am not aware of any previously published work describing the acquisition of this type of selectional restriction as a general model of the language, however.

3.3 Category Discovery

The preceding sections have described how evidence of the associational behavior of words and contexts can be acquired from corpus data. This section, is an overview of how these behavioral distributions can be used to identify the families of lexical and context behavior, and to construct the class representations $L$ and $\Gamma$ that we’ve discussed as the heart of the associative model.

3.3.1 Comparing Distributions

The classes we wish to construct are those that represent the common associative behavior of their members. For the lexical classes, this means that the members of a class $\ell_i$ should all be grammatically licensed in the same set of constructions. In terms of our distributional representation, this translates into each word $w_k \in \ell_i$ having a context frequency distribution $w_k(C)$ similar to, or even identical with, the distributions of the other members of the class. Similarly, to find the context classes $\gamma \in \Gamma$, we want to compare the word frequency distributions $c(W)$ in order to identify the families of similar context behavior.

Comparison measures

In order to identify the families of similar behavior in the co-occurrence data, and build classes to represent them, we need to be able to search among the words or contexts to find groups of items which have associative distributions with similarities and common traits. To compare distributions in this way, we need to have some measure of ‘similarity’ between associative distributions, and
to formally define our notion of class in terms of some function or optimization over the similarity values. But what kind of comparison might be appropriate?

What we need is a measure that can compare two (or possibly more) distributions, reward associations held in common, and perhaps deduct for behavior present in one but not in others. Consider the situation shown schematically in figure 3.1, where the context frequency distributions of two words are given for a small set of contexts. Given that the two words have such overlap in their sets of associated contexts, we would like to say that these two words are quite similar in their associative behavior.

Various measures of distributional similarity are available to us to quantify this similarity.

**Predictive similarity.** Work by Lee, Dagan, Pereira and others ([Lee:thesis?, Pereira-tishby-lee?]) ([Dagan et al., 1993]) have been the principal studies into the use of information theoretic measures such as the Küllback-Liebler distance.
(also known as the relative entropy) or the Shannon-Jensen distance (total divergence) to study distributional similarity. This type of measure is a function of predictiveness between two distributions, essentially addressing the question of how much of the one observed distribution was produced assuming an underlying process characterized by the other distribution.

This might be a very useful type of measure for our application, because the question we are considering can be addressed from this same perspective. One way of characterizing our notion of a class is that a class member's behavior is determined by the functional features of the class, plus some specific idiosyncrasies. Then, to identify the class of origin for a word with some observed behavior, we want to know how reasonable it is to assume that a word's context distribution (or a context's word distribution) could be produced by the distribution we have found for the class as a whole.

Model error. One could also define the identification of classes in terms of representational error. Using this criterion, the best class representation would be framed in terms of the error between the distributions calculated from the class model and the observed sample co-occurrence distributions. Deciding how that error is to be measured is another set of decisions. The information loss approach described above is essentially a specialization of this least error approach, where 'error' is defined as 'information loss'. Alternatively, we could try to minimize the absolute error in the number of occurrences, or the square of this, for a least-squares error approach. Or perhaps we could normalize the error on a per-word or per-context basis.

In [Schütze, 1993a, ?schutze:others?, ?finch:thesis??], a least-squares approach is implicitly used in a singular-value-decomposition based data reduction. This gives some promising results, but is based on a heavily massaged data representation that greatly increases the density of the co-occurrence data, at the
cost of greatly limiting its resolution of individual word-context pairings, eliminating the ability to separate the uses of a word across multiple differing contexts. (These concerns are discussed more at length in §??..??).

**Information loss.** Another way of looking at this same problem is one of *information loss*. The question of the suitability of the class as a description for the individual word or context can be answered by examining how much information is lost between the average, generalized description assigned to the class, and the specific, higher information contained in the observed distribution. This is similar to the method used in [Brown *et al.*, 1992], in which classes were built incrementally by combining the most similar existing classes to give the lowest information loss at each step. This is of course different from finding those classes with the least global information loss, but serves as a practical method of finding a good set of classes.

### 3.3.2 Identifying class membership from usage data

Having just outlined a few different measures that have been used in the literature for comparing associative distributions, it remains to be seen how any of them might be applied to the formation of the associative class model we have discussed. One important difference between the works mentioned above and our goals here is the size of the context frame involved. We wish to include a relatively large context, in order to capture the inter-lexical dependencies of usage. Where most of these techniques rely only on one, sometimes two words of context as a predictor, we would like to use *three or four* to each side. The co-occurrence techniques will be impractical for this task due to the sparse nature of word distributions – there is not enough data available to find co-occurrence distributions robust enough that they can be compared effectively.
3.3. CATEGORY DISCOVERY

The next chapter (ch. 4) discusses in great detail the difficulties that the extremely sparse nature of the co-occurrence data poses in the comparison of distributions, and how this interacts with the class formation problem. It also introduces the technique of structural similarity smoothing used to overcome some of these difficulties. In the chapters following that, the specifics of the techniques and implementation are detailed.
Chapter 4

Smoothing Difficult Data

The most relevant problem in acquiring a distributional co-occurrence model of word usage like the associative model described here is that of sparse data. Even with astronomically large corpus resources, building a complete model of $w(C)$ and $c(W)$ using the simple acquisition methods outlined above is impractical at best; at its worst, utterly impossible. The data to cover all the possible, or even interesting cases, will simply not appear in even the largest manageable corpus.

This chapter explores the nature of these difficulties with the data, and the linguistic phenomena that lead to them. I also cover some of the smoothing techniques that have been used to address the problem of handling previously unseen words and constructions. Those familiar with the problems may wish to skip ahead.

The later section of the chapter develops the particular form of solution to these problems adopted in the associative model: a class-based model in which our knowledge about contexts and words can be extended by analogy with others that have similar behavior, in which sparse data is condensed using structural similarity metrics. This kind of class-based description allows a kind of grammatical inference available only in models which assume an underlying systematicity
to the language. Many other statistical language models capture only the independent behavior of contexts or words, and so can offer no insight into the extended structure of the language system.

4.1 Sparse data

The acquisition of the MLE parameter model of word and context associations given above is fine in the ideal, but the world of real data makes it quite impractical to pursue. The difficulty here is that the corpus of training data used to form the original ML estimates, is not, and practically can not be, sufficiently representative of the language in general.

With a realistically large vocabulary of, say, at least \( V=10^5 \), the number of possible distinct word/context combinations for even a short window such as a trigram model (or \((2,0)\)-context in our representation) swells to \( V^3 = 10^{15} \) combinations! Given that current very large text data resources count their words only in the 100's of millions, or occasionally in the billions\(^1\), the size of the data relative to the number of possibilities guarantees that most of these \( 10^{15} \) distinct triples will not appear even once. If we had billion-word corpus (an American billion: \( 10^9 \) words), we would see at most \( 10^6 \) unique trigrams, but more likely far fewer, due to repetition an natural patterns of the language. So we are left with fewer than one in a million of the possible trigrams accounted for. And, for those contexts that do appear, the conditional count \( \| \{ c_i | t_0 = w_j \} \| \) will be zero for almost all words \( w_j \).

Gale and Church [Gale and Church, 1990] have observed that even as the test resources grow, the observed vocabulary size \( V \) grows at a rate which restricts even the possibility of forming a complete bigram model.\(^2\)

\(^1\)As a size comparison, consider that the Internet Archive (http://archive.org) maintains an archive of all accessible web content, which is currently \( \approx 10^{12} \) words, ignoring the fact that much of the archive is image, program code, and other non-textual information.

\(^2\)Many of the terms observed in this extended vocabulary were surely rare words words such as proper names, of which almost as many new examples can be found or constructed as one might like. There is of course a limit to the number of English words under a certain length, so there is in some sense a limit to this vocabulary growth. However, for corpora of reasonable and manageable
4.1. SPARSE DATA

This affects the possible analysis tremendously. One conclusion we can draw, since we already assume that grammatical constraints will rule out the use of a certain number of constructions, is that an unobserved context, or set of contexts, will never be seen. Perhaps these patterns are excluded from the language by grammatical constraints. This is the naively optimistic conclusion. Because we have so little data relative to the possible number of word patterns, it is much more reasonable to suspect that if we had more data, at least some of these 'excluded' constructions would indeed appear. This is simply to say that novel constructions and combinations of words are encountered all the time, regardless of how much data we have previously seen.

**Word frequencies**

The problem is compounded further by the fact that the frequency distribution of single words generally follows a form known as Zipf's Law [Zipf, 1972, pp. 25 ff.], which, in the simple form, estimates word frequency as the inverse of the frequency rank by the formula \( f(w) = K/r(w) \), where \( f(w) \) is the number of occurrences of the word \( w \), and \( r(w) \) is the rank of that word in terms of frequency, with the most frequent word having rank 1, and the least frequent word rank \( V \).

This relation means that common words are far more ubiquitous than uncommon words. Figure 4.1 and table 4.1 demonstrate this for the news corpus used in the experiments of chapters 5 and 6. Although it is somewhat difficult to see in the log-log plot, low frequency words make up a large proportion of the total data. Singleton words (frequency = 1) comprise 23030 of the 70530 distinct tokens in this sample, accounting for 33% of the total vocabulary.

**Context frequencies** This type of word distribution in the language conspires against us when we try to acquire estimates for context frequencies. A context containing a low-frequency word can have a frequency no greater than that word.

---

25 February 2000
Figure 4.1: log frequency v. log frequency rank for tokens in WSJ corpus demonstrating that \( \text{freq} \times \text{rank} \approx \text{const} \)

More likely, the expected occurrence of the combination of words that make up the context is orders of magnitude less — if we assume words are independent, it is 
\[
p(c) = \prod p(w_i),
\]
for \( w_i \) a word in the context \( c \). If more than one word in the context has a low expected frequency, the expectation of their occurrence together will be astronomically low.

Some quick calculations to build an “average-case” model of context occurrence will help to estimate how low that can really be. Consider for a moment a uniformly independently distributed example (which is in contradiction to Zipf’s distribution, and in contradiction to known facts about interword dependence). With a vocabulary size \( V \) and a corpus size \( N \), each word then has, by definition, frequency \( N/V \). The probability that any particular token in the corpus is a given word is \( 1/V \). The chance then that a given context of length \( k \) occurs as a particular subsequence of length \( k \) is \( V^{-k} \). Since there are \( \approx N \) such subsequences in
the corpus, the probability that a given context appears somewhere in the entire corpus is $p(c_i) = 1 - (1 - V^{-k})^N$ (this is the complement of the chance that it does not appear). The second term, $(1 - V^{-k})^N$, is exactly $\sum_{j=0}^{N} -1 \binom{N}{j} V^{-jk}$. For large $V^k \gg N$, we can estimate this term as $(1 - V^{-k})^N \approx 1 - NV^{-k}$. So for a context $c_i$ of length $k$, $p(c_i) \approx NV^{-k}$.

If we want to verify, directly from our observations, that a given $c_i$ is in the language, or to conclude that it is excluded from the language due to structural constraints, we need a corpus large enough to make the probability of the observation reasonable. The current estimate implies that for a $1/2$ chance of seeing this particular context we need a sample length at least $N = V^k/2$ (this estimate is actually quite low, because we’ve exceeded the assumption that $V^k \gg N$). This corpus size is completely unattainable even for the case of English trigrams, where $V^k > 10^{15}$. The situation becomes exponentially worse as we choose to look at longer contexts.

The real situation is even worse. Since the language really is Zipf-distributed, most symbols have very low frequencies, so that any chance of finding a context composed of a few low frequency symbols requires even larger amounts of data.
the interesting symbols have even as much as 1/10 the average frequency (already a high estimate) then we need a sample length at least \( N = 10^k V^k / 2 \), which is \( 10^k \) times larger than needed previously. This is prohibitive, to say the least. Since the Zipf distribution implies that most symbols have even lower frequency, we would need an even larger corpus, and it becomes clear that ever acquiring enough data for a reasonably complete estimate is simply impossible.

## 4.1.1 Comparing distributions

One of the goals of our language model was to be able to compare the use of words, in order to identify words that obey similar constraints. Earlier, I suggested using the context frequency distribution as a mechanism for comparing the grammatical properties of words. The idea is that by considering the contexts in which two words appeared, one could gain an understanding of the associative constraints common to both by observing the commonalities within the contexts — the similarities of their projected constraints would be apparent in their common context sets.

The situation described could be schematically conceived as shown in figure 4.2, where the common contexts (e.g. \( c_1 \) and \( c_3 \)) indicate a shared grammatical constraint, whereas the appearance of certain contexts in the usage of only one word or the other helps determine which syntactic features are not shared.

However, the incredibly low probability of contexts has a serious consequence on this proposal. Because the chances that any particular context will appear are so small, the chances that any two words will appear in an identical context are incredibly low, even if the words project very similar constraints! In this case, there will be very few pairs of words that can be compared through their context frequencies, because very few words will appear in identical contexts. The diagram in reality looks much more like figure 4.3 than it does the original figure 4.2. The immense diversity of possible contexts also implies that even for a single word,
4.1. SPARSE DATA

![Graph comparing association of two words across limited high-probability contexts](image)

Figure 4.2: Comparing two words across a limited number of high-probability contexts.

few, if any, contexts will ever repeat exactly, so that we will also have no evidence of the word’s preference for one over another.

Nothing to compare

This is the most serious problem with such sparse data. There will be so few occurrences of the same context appearing more than once that we really have no basis for comparison at all. As was made clear above, limited data and sparse distributions guarantee that the count \(|\{c_i | t_0 = w_j\}|\) will be zero for most combinations of \(c_i\) and \(w_j\).

For an example of these extremely sparse conditions, Brown et al. [Brown et al., 1992] report that for a particular 3-gram model, using a very large corpus of \(3.7 \times 10^8\) words, only \(7.5 \times 10^7\) of \(1.8 \times 10^{16}\) possible 3-grams appeared at all, for a data rate of \(4.3 \times 10^{-9}\). Of those that did appear, 71% appeared only once. If one attempted to use this model to analyze new data from the same source, almost 3 of every 4 tri-grams seen in the new data would not have been represented in the model.
Figure 4.3: Comparing two words across an enormous number of low-probability contexts. Even with high word frequencies, correspondence between the distributions will be rare.

All of these approaches are further complicated by the fact that the chance that a given context will be observed in corpus diminishes exponentially with the length of the context, meaning that if we were to represent each possible context explicitly, most counts will simply be zero. This implies that any system, human or machine, attempting to acquire a notion of syntactic constraints based on this kind of distributional matching will have a very difficult time.

4.2 Estimating missing values

4.2.1 Smoothing

One approach to estimating values for these context distributions is to smooth them. Since the data has been gathered through a discrete process, with a context appearing an integer number of times (sometimes not at all), the collected statistics suffer from quantization effects. If a low probability event appears even once, its frequency will be estimated many times too high; if it doesn’t appear in
the corpus, it will be grossly underestimated. Likewise, if a moderate frequency event appears even one too few times within the particular sample of text, its frequency will also be grossly underestimated. The typical Zipf distribution tells us that "moderate" frequencies will be somewhere in the 2–10 range, so a sample frequency that differs from the 'true' frequency by even one event is a large fraction of evidence.

Because of the magnitude difference between the enormous space of possible events and the small number of available sample contexts, most events have a very low expected frequency events and will be seriously misrepresented within the sample. An event that does not appear in the corpus sample could easily have a non-zero chance of occurring. Smoothing operations re-estimate the sampled frequencies in order to reduce the effects of this quantization.

**Intrinsic smoothness lacking**

Trying to accurately smooth linguistic occurrence data can be quite difficult. Smoothing data from other types of systems, such as a measurement on some physical system, is made easier by the assumptions that can be made about the underlying relations in the data. Assumptions about the metric space in which the events occur, continuity, limited derivatives resulting from the physical limits of the device being measured, and models of the physical attributes and behavior of the device greatly simplify the tasks of estimating missing values, identifying spurious measurements, and drawing relations between the observations.

In these cases, values at unmeasured points can be estimated because there is a smooth model of how values will deviate away from known or measured points. The situation is often as in figure 4.4 in which missing data can be filled by interpolation, and erroneous data can be seen as deviating from the model and can be normalized using techniques such as local averaging. The end result is that the data can be fitted to the form of a smooth curve, because there is an
explicit smooth model of the underlying phenomenon.

![Graph showing smoothed, interpolated estimator]

Figure 4.4: Interpolating missing values smooths the curve, if one has a good theoretical model to fit the curve to.

**Linguistic data is discrete**

Unfortunately, with linguistic data, the typically more accurate picture is as shown in figure 4.5, in which there is no underlying model to structure the domain, and where most values are missing. What makes this most difficult is the discrete nature of the data — words and context sequences, which are either exactly equal, or not equal at all. Most statistical language models do not include any common-sense notions of word similarity, e.g. ‘the’ and ‘a’ are much more alike – closer – than say, ‘the’ and ‘republican’. Because the observed events are of this kind of point-like, nominal observations, with no underlying continuous model, the non-observed events cannot be estimated by averaging between the known points — there is no well-defined ‘between’ that can be used to calculate the average.

Typically, each possible context (or n-gram) is also treated as a completely distinct entity. Because there is no metric for defining how one word relates to any other, contexts that contain any differences in their constituent words must also be treated as completely different. Because there is no mechanism to determine
which other contexts would be useful as references, each point must be treated independently, which eliminates local averaging and interpolation as applicable methods.

### 4.2.2 Non-linguistic smoothing estimates

Without having a continuous model with exploitable local properties, as one has in a physical system, one must use other assumptions about the theoretical shape of the phenomena in order to smooth and interpolate the data. There are some smoothing techniques which rely on very broad assumptions about the statistical structure of the data in order to re-estimate observed frequencies. Methods such as the Good-Turing estimate [Good, 1953, Church and Gale, 1991] and the *deleted* estimate [?jelinek+mercer85?] assign estimated frequencies to each unobserved event based on the relation between the number of possible events and the number which were observed.

The deleted estimate relies only on the assumption that the same stationary process underlies two separate language samples. By observing the number of events in the second sample that were not seen in the first, one can construct an estimate of expected frequency for classes of these unobserved events. The Good Turing estimate, based on species population studies, uses this along with the
assumption that the counts of symbols are binomially distributed [Good, 1953]. Both of these methods treat all events observed exactly \( k \) times (including \( k=0 \)) as a class, and assigns an estimated frequency uniformly to the members of the class. Unfortunately, this means that the frequency predicted for all unobserved events is the same. The predicted frequency of constructions which are not part of the language will be the same as that predicted for the smaller but still considerable number of allowable events whose frequency is low enough that they do not appear in the sample.

Church and Gale [Church and Gale, 1991] introduce the important class of enhanced Good Turing and deleted estimates, incorporating the known additional information about the distribution of single words (or unigrams). This information is used to categorize the possible events according to the expected frequency, calculated as if the constituent words were independent, in order to improve the resolution of the frequency assignment. This means that an unobserved combination of independently high-frequency words will be classified differently than an unobserved combination of intrinsically low-frequency words, and the estimated frequencies will be different. Although they report a considerable improvement in the predicted frequencies under this method, the technique still uses no linguistic information about the configuration of words, and cannot distinguish between meaningful and non-meaningful arrangements of symbols. A non-meaningful sequence composed of high-frequency words that was not observed (for good reason!) will still be assigned an estimated frequency higher than an unobserved, but allowable event that was not observed due only to low frequency.

There are many other approaches to improving the resolution of frequency assignment. Jelinek and Mercer [Jelinek+Mercer80?] propose an interpolated estimation technique for \( n \)-gram frequency estimates (eqn. 4.1). Each estimated value \( f() \) has associated with it a weight, \( \lambda() \), which adjusts the contribution this value represents, based on frequency properties of the current predictor context.
\[ f(w_0|w_{-n}-w_{-1}) = \lambda_n(w_{-n}-w_{-1})f_n(w_{-n}-w_{-1}) + \lambda_{n-1}(w_{-n+1}-w_{-1})f_{n-1}(w_{-n+1}-w_{-1}) + \ldots + \lambda_1(w_{-1})f_1(w_{-1}) + \lambda_0f_0(w_0) \] (4.1)

If the higher-order predictors \((f_n())\), which use the largest context available, are deemed unreliable or uninformed due to a lack of sufficient data, they are given a low weight \((\lambda_n())\), and the brunt of the prediction is shifted to the lower order estimates, which are based on shorter contexts. This decision is referred to as 'backing off' to a lower order model. One could also view this as a 'fleshing out', where the base model is a low order estimate, refined by the inclusion of higher-order statistics when there is sufficient evidence to judge them reliable. From a statistical point of view, they are using the marginal values from a high order contingency table to estimate values for sampling zeroes and low frequency events within the table.

The major drawback to these estimates, however, is that they almost completely ignore linguistic abstractions of the data.\(^3\) They are based on general truths about population frequencies, and use only the most limited amount of contextual information as an indicator of word use. They treat the data as a stochastic stream of tokens, for which the redundancies and repetitive structures can be modeled in terms of the predictive capacities of a short string of symbols (one or two is typical). This approach ignores the view of language as a *meaningful* stream of symbols, where the interrelations between symbols are governed by some underlying systematic constraint structure.

\(^3\)This is not entirely true — the back-off models reduce the context so that it is always adjacent to the predicted symbol. This at least incorporates the idea that inter-word dependencies are local in nature.

25 February 2000
CHAPTER 4. SMOOTHING DIFFICULT DATA

Treating language as a stochastic stream is not at all a step backwards, though. It certainly provides a broader view of language data than we can garner from introspecting on sets of examples, or investigating correlations between small sets of occurrences as is often done in linguistics literature. But, because we know so much more about language, these limited stochastic properties can serve only as a groundwork for more complete and accurate models.

4.2.3 Linguistically informed methods

Adding linguistic information to a language model can improve its accuracy. In particular, a linguistically motivated account of word behavior can be used to better estimate the frequencies of unobserved events. A variety of linguistic generalizations can be used to improve these interpolation methods, and model unseen events by analogy with the observed behavior. Dagan and others 4 [Periera et al., 1993, Dagan et al., 1994b, dagan:similarity-disambiguation?] have presented methods which take into account similarities among word behavior in order to predict word use. In order to better model a word \( w_i \)'s expected behavior, they average the context distributions of 'similar' words to create a more robust, composite estimate of the distribution for \( w_i \). Similarity in this case was arrived at by comparing the context distributions \( w(C) \) using an information-theoretic measure, the Kullback-Liebler distance. The KL distance measures the degree to which one distribution predicts another, so that words whose distribution of behavior, defined in terms of observed contexts, closely matches the target word's observed distribution are used to round-out the target's expected distribution of behavior.

The assumption behind this is of course that words whose behavior closely approximates one another's are filling the same functional role in the language. Thus, an appearance by one in a situation implies that the other is also licensed there.

\(^4\)cite Lillian Lee (Tishby, Lee, etc.) here as well!
4.2. ESTIMATING MISSING VALUES

These similarity judgements can be computed only where there is some reasonable overlap of behavior. If the data is too sparsely distributed, or the observed frequency of the interesting words is low, then the chances that two words will have any behavior in common is also low. The KL distance is computed over a discrete distribution of the individual conditional contexts. If two words share little or no behavior exactly, then their similarity will be very low.

Because Dagan et al. use bigram data, their similarity estimates are robust enough to be practical. One word to the left gives all of the information — this is an incredibly short predictor. A larger context, or longer \( n \)-gram, would reduce the data density so much, though, that I do not believe these methods would be directly extensible past the bigram. Because the occurrence data is so truly sparse, a larger context would give practically no context associations in common between two words, and thus no means to compare the words. This situation was pictured schematically in figure ?? above.

4.2.4 Classification methods

The most obvious abstract linguistic extensions to the stochastic models can be drawn from the existence of constrained, regular syntactic structure. By noting that individual words do not behave completely independently and idiosyncratically, but instead follow patterns of behavior similar to others, we immediately arrive at the idea of categorizing words into lexical classes which can capture the general patterns of behavior.

If we observe the interactions among this reduced set of classes, rather than among the numerous individual words, the available data becomes much more complete. Estimation of the occurrences and interactions among the classes becomes more tractable. The condensed model is easier to represent, reason about, and compute with. From a model of word class interaction, the behavior of individual words can be re-estimated according to their membership in the particular
classes.

Many approaches of this type have been developed. One needs of course a measure for defining the similarity between words. Once a measure for comparing behavior has been selected, there are many different clustering algorithms available, all of which attempt to form groups whose members are 'more similar' to each other than they are to members of other groups [for an overview see Anderberg, 1973] or [Everitt, 1980]). Agglomerative clustering is a common technique, in which at each stage the items which are judged most similar are joined to create a new cluster. After re-evaluating the distances to the new cluster, clustering continues by finding the next pair of items that are most similar, and so on. This creates a binary tree of items joined into clusters.

Inversely, one can form a tree of clusters by optimally dividing the groups of items into smaller and smaller clusters. Divisive clustering techniques start with the entire collection of items as a parent cluster, and seek to divide it into groups that are most dis-similar. This also creates a hierarchical tree of clusters and items.

Another class of techniques, non-hierarchical optimization clustering, seeks to form the set of clusters that best optimizes some global similarity function. Rather than comparing the distance between each pair of existing items (as in the agglomerative techniques) or proposed split (divisive methods), a global function of the total fitness is optimized by re-assigning items to clusters. The $k$-means algorithm does this by shifting cluster centroids until an optimal geometric fit is achieved.

**Classification for word classes**

A number of studies have been done which use agglomerative clustering techniques to derive classes of words with similar behavior. In all of [Schütze, 1993a], [Finch and Chater, 1992], [Hughes, 1994] and [Futrelle and Gauch, 1993], clusters
are formed by agglomerating words based on distributional measures of similarity. Each of these studies uses some description of the distributional properties of a word, in which all the occurrences of each word are represented in the distribution.

**Limitations of fixed word classes**

Many of these class-based methods suffer a serious drawback, however, in that they make the classification too rigidly. If a word is assigned to only a single class, and then the class behavior is used as the representation for the word's behavior, they imply that each member word of the class is completely substitutable for any other. The class representation may indeed be an improvement over individual word statistics, but this overgeneralization is perhaps too severe. In truth, each word may have a variety of aspects of behavior, any one of which it shares with a number of other words. This is the same as saying that it may assume a number of different functional roles within the language, each operating under its own set of constraints. But to say that all the roles and constraints associated with a word are to be found shared by all other members of the class, and to the same degree, is an assumption that we should avoid making.

This overzealous grouping can be avoided by adopting a class representation in which each word can have membership in more than one class. Both [rooth:clusters?] and [lee:thesis?] demonstrate class formations schemes in which membership in classes is graded. Rooth presents a class-based model of noun-verb selectional types derived using the iterative EM algorithm. Each selection type in this model carries both a probability distribution of verbs and one of nouns, indicating the words most likely to appear in interactions of this type. Each type carries distinct distributions. Furthermore, each type is assigned an occurrence probability (eqn. 4.3).
verb probability \( p_v^\tau, \quad \sum_v p_v^\tau = 1 \) \hspace{1cm} (4.2)

noun probability \( p_n^\tau, \quad \sum_n p_n^\tau = 1 \)

type probability \( p_\tau, \quad \sum_\tau p_\tau = 1 \)

These distinct distributions allow us a variety of ways to view the data. The \( p_v^\tau \) distribution gives us the chances of all verbs with respect to a given type of N-V interaction. One can also view this as the chances of a verb given a certain expectation of the noun. The type \( \tau \) defines a class of N-V interaction, and in the case of \( p_v^\tau \) effectively references a fuzzy class of nouns involved in the interaction. This distribution \( p_v^\tau \) gives us the expected relationship of verbs to this class. The distribution \( p_n^\tau \) is the dual of this, the expectation from the point of view of the noun.

Of course, Rooth’s model deals only with the N-V relationship, and relies on having input pre-labelled with that relationship. Modelling this kind of limited interdependence also eliminates the problems posed by the syntactic diversity and ambiguity in language. In general, though, English words are remarkably flexible in their usage, and a single term might appear in verbal, nominal and modifier roles, with perhaps multiple sense interpretations in each. Dissecting and categorizing this complex web of unrestricted usage is accordingly more difficult.

4.3 Similarity and Classification

The approaches described so far all rely on distributional similarity in order to make comparisons and to form classes. This similarity is judged over the total range of behavior of an item, taking into account its co-occurrence with a variety of predictor contexts. However, as was pointed out in the first sections of this
4.3. SIMILARITY AND CLASSIFICATION

chapter, for the large contexts that we would like to use, there not nearly enough
data to be able to make such comparisons effectively. This is not because there
are too many words, but rather because there are too many combinations used
in the language.

If we could organize the immense number of combinations into groups that
each had some consistent and predictable properties, then the occurrence
frequency of each group might be high enough that to give some reasonable occur-
rence density of word usage across the groups. Knowing the associative behavior
of words in each group, in turn, enables us to compare word usage across the
language and to create a more complete model of word use.

This section outlines a method for doing exactly that. Using the assumptions
of language structure outlined in chapter 2, I outline organizing principles which
can be used to find context sequences which can be expected to have similar be-
havior. I then show how this similarity can be used to condense the cooccurrence
statistics to a point where real comparisons of behavior become practical.

4.3.1 Behavioral assumptions

In constructing a model, we initially know nothing about the relationship between
the linguistic behavior of any given words — in fact, this is exactly what we are
trying to learn. Neither do we know anything about the relationship between the
behavior of various possible context sequences that may occur. What we do as-
sume, however, is that words in the language are not behaving arbitrarily. Words
carry selectional restrictions that constrain their usage in combination with other
words. Further, we can expect that the number of structural mechanisms and
configurational patterns will be small, compared to the number of possible con-
figurations.

These are the foundational premises which forms the basis for this study of co-
occurrence and language modelling. We expect to see categorical patterns in the
behavior of words. These expectations arise both from the theoretical concerns outlined in chapter 2, and from the success of other statistical co-occurrence studies and grammatical representations alike. I’ve used these assumptions thus far to justify the associative functions, which describe the behavior of words and constructions. They can also be used to carry us through the problem of sparse occurrence data.

4.3.2 Structural Similarity

The contexts are not discrete objects — they are composed of sequences of words, words which each project configurational constraints, or selectional restrictions, onto the remainder of the context. While we have been treating individual words as discrete objects, which we know nothing about before we begin our modelling, we can treat the contexts as composites, built from a structured assembly of separate words. And the same constraints which operate between the words and the context will also interoperate between the words used within the context. The matrix of constraints set up by the ordered sequence of words in the context operates to limit possible variations in the context words, as much as it does in the gap filler, \( t_0 \).

Up until now, we have only been concerned with the interactions between \( w_0 \) and the remainder of the sequence, viewed as a monolithic whole. If we consider the other positions, we can just as well reason that their variation is also highly constrained by the remainder of the sequence. If we look at any other position independently, they must also be limited by the matrix of constraints imposed by the rest of the construction.

If we consider a specific word-context pair, such as

\[
\begin{array}{ccccccc}
  t_{-3} & t_{-2} & t_{-1} & t_0 & t_1 & t_2 & t_3 \\
  c = & \text{with the red book lying on the}
\end{array}
\]  

we know that the indexed position \( t_0 \) is limited to some set of common nouns
which generally reference physical objects — ‘envelope’, ‘dog’, or ‘cup’ would fit equally as well as ‘book’ does (they would all have relatively high $c(w)$ values). Most words, though, are restricted from this context; they are ungrammatical in this use. For instance, ‘without’, ‘achieve’, and ‘having’ are all common words which cannot be used here. The context's projection onto the lexicon, $c(W)$, selects for a relatively small set of words.

**Allowing substitutions**

We can use the same reasoning to conclude that the other positions must also be constrained in their usage. All other things being equal, we can replace, for instance, $t_{-2} = \text{the}$, only with words belonging to the limited class of determiners. If this position were occupied instead by a different determiner, such as ‘some’ or ‘a’, the total matrix of constraints operating between the context and the index word at $t_0$ would remain much the same, in terms of the word choices allowed. The lexical projection of this new context, $c'(W)$, would be very similar to the projection $c(W)$ of the original context. Other words in the context $c$ could be replaced with words having similar behavior, and these changes would also result minor impact on the associative behavior of the context.

So we see that the one particular context specified in (4.3) gives rise to a whole family of grammatically related contexts, all of which project similar lexical constraints, and specify almost the same set of index words. In traditional terms, we might generalize this family of contexts using lexical class labels along with a partial tree structure, or by labelling the sequence with part-of-speech or other grammatical categories, as in (4.4).

\[
\begin{array}{cccccc}
 t_{-3} & t_{-2} & t_{-1} & t_0 & t_1 & t_2 & t_3 \\
 \text{with the red book lying on the} \\
 \text{Prep det Adj} & - & V_{\text{ing}} & \text{Prep det} \\
\end{array}
\]

This new sequence of part-of-speech labels (4.4) obviously specifies a more
general set of possible index words. And while this set might have many possible members, the size of the set is microscopically small compared to the total range of possible context sequences, because these contexts are limited structurally to be like the original. And still, the new sequence projects very similar constraints on the key position, \( t_0 \), which we can expect also to have a very similar distribution of allowable words as did (4.3).

**Similarity**

There are a number of features to consider when we try to compare two context sequences for this kind of similarity. The number of words which are identical between the two sequences is of critical importance. Certainly, if two sequences contain the same words in the same positions, we would define them as identical. If only one word is different, we might still believe that they project very similar constraints on \( t_0 \) (in the absence of other, more linguistic, knowledge). As more and more words fail to match between the two sequences, our prediction that they have similar grammatical properties, and therefore similar projections, decreases accordingly.

The type of mismatch is also very important. If, for instance, we were to compare the sequences in (4.5), we see that many tokens fail to match because we have inserted one word, *large*, displacing the others. Certainly, the insertion of a modifier such as this does not dramatically change the projections of the context.

\[
\begin{array}{ccccccc}
  t_{-3} & t_{-2} & t_{-1} & t_0 & t_1 & t_2 & t_3 \\
\text{with the red} & \_ & \_ & \_ & \_ & \_ & \_ \\
\text{with the large red} & \_ & \_ & \_ & \_ & \_ & \_ \\
\end{array}
\]  \tag{4.5}

Location of the discrepancies is also an important feature. If the tokens at \( t_1 \) do not match, there will be a greater change in the constraints on the gap than if they match up to \( t_{100} \). The more local the mismatch is to the central position, the greater effect it can have on the constraints projected onto the gap. This agrees
well with our reasoning from ch. 2 – that configurational constraints are limited, in general, to the local region of a word.

The type of disagreement between the sequences is also quite important. As in the example above (4.4), substituting a term with another from the same category (i.e. one that we know projects similar syntactic constraints) will have much less of an impact than using an unrelated term. Replacing *red* with *blue*, for instance, has almost no effect on the range of possible fillers for $t_0$.

### 4.4 Organizing the context space

Having a measure of context similarity allows us to treat the discrete space of context sequences quite differently than before. If we imagine that we can construct a measure of context similarity that has these desirable properties, we can see it would have great advantages in dealing with the sparse nature of co-occurrence data. Because we now have a real sense of what a 'close' context is, it gives us the ability to smooth the space and average across context occurrences in some linguistically motivated fashion.

What this creates is a structured space of contexts, over which we can use the kind of exploitable local properties we need to smooth the co-occurrence data and infer values at unseen points. If, as we just argued, we can make a principled judgement of how much the constraints on the central $t_0$ term will vary depending on the degree of context similarity, we then have a mechanism via which we can combine discrete information about separate contexts.

#### 4.4.1 Grammatical analogy

There are two important ways we can use this new linguistically smooth structure of the context space. The first is through inference. If we know the range of associated use for a context, say $c_a$, and we know that this context is close to another, $c_b$, we should then be able to infer that the associations for context
$c_b$ should be close to those of $c_a$. More formally, let us use the function $s(c_a, c_b)$ for the structural similarity between the two context sequences. And, let $d(X, Y)$, where $X$ and $Y$ are distributions, be a distributional similarity measure like those discussed earlier in §3.3. Then our inference can be stated that if the two contexts are similar enough, $s(c_a, c_b) < \epsilon$, then the allowed distributions of words, $c_a(W)$ and $c_b(W)$, should also be close under the distributional similarity measure: $d(c_a(W), c_b(W)) < \delta$, where $\delta$ is some function of $\epsilon, c_a$, and $c_b$. This is a general result about the expected variation of the associated word distribution. To state it another way, as long as the two contexts are similar enough, then the associated word distributions can only vary within some known bounds.

This kind of inference will allow us to use a process of grammatical analogy to extend the data about co-occurrences and constraints that we collect from corpus. The constraints and associations of contexts that we have not observed can now be derived by analogy with similar contexts for which we have collected empirical corpus evidence.

### 4.4.2 Structural context classes

The second important way in which we can use this structured context space is in the construction of context classes. The space of possible contexts is enormous, exponentially larger than the number of words. It would be an intractable task to have to derive and store the separate associations of each possible context. By dividing the space into a smaller number of context classes, we can reduce the burden of modelling while also increasing our ability to acquire the model empirically.

Since our similarity measure can now tell us which contexts are alike, we can group the set of contexts, $C = \{\ldots, c_i, \ldots\}$, into a much smaller set of context classes, $\Gamma = \{\ldots, \gamma_i, \ldots\}$, where each $\gamma_i$ is a composite description of a number of $c_i$. We can accomplish this perhaps by simply dividing the space uniformly into
a number of regions. Unfortunately, though, there are large areas of the context sequence space that simply do not contain grammatical sequences, and we should not waste our efforts trying to model behavior in those regions.

More reasonably, we can use the sorts of adaptive clustering techniques discussed earlier with distributional studies to find just those groups of contexts which are pertinent to linguistic behavior. Using, say, an agglomerative clustering scheme over corpus data would create context classes that cover only those regions of the context space that are actually used. Further, such a scheme is adaptive in the number and regions of the groupings created, such that more classes would be created in those regions that were most heavily populated, allowing finer distinctions to be made in those regions.

Once we have created classes with this kind of clustering process, they would be useful in a number of ways. In terms of data reduction, each class could be treated as a single entity for measuring co-occurrences. Any word that appeared with a context which is a member in the class would count toward the associative distribution of that context class. The word distribution for the class would be the sum of the distributions for the member contexts, resulting in much more robust statistics for the class description (4.6).

\[
\gamma_j(W) = \sum_{c_i \in \gamma_j} c_i(W)
\] (4.6)

These composite class descriptions would be robust enough that one could begin to make distributional similarity comparisons over the associated word distributions. This would extend the model to include functional similarity judged by comparing the associated sets. Even though the basis of this initial class formation is solely the structural composition of the contexts, a pair of contexts may project very similar constraints and yet have little in common structurally. The more robust aggregate class descriptions allow one to use the same distributional techniques as for the bi-gram word similarity techniques, except that the elements being compared are now the context sequences.
These clustered context classes would also provide a kind of dictionary of behavioral types for the associated words. We have already discussed the difficulties of dealing with words which display multiple kinds of grammatical behavior. By mapping the words' usage over the range of context classes, we will be able to divide the appearances of each word into distinct modes of use, which likely correspond with sense distinctions [Waterman, 1993], yarowsky:disambiguation?].

4.4.3 Bootstrapping Distributional Classes

One of the most important uses of the structural context classes would be as an aid to the formation of word classes. For words, we have no detailed information available with which to compare of their component, structural parts as we have for the context sequences. We could, as was mentioned before, use a morphological model to judge similarity among words, but this would be at a level far coarser than we have been imagining, and would have no implications for inter-word syntactic dependencies. It is better to treat the words as completely independent entities, and compare them solely on the basis of their distributional properties against the contexts.

With the co-occurrence data being as sparse as it is, we have no hope of using the raw corpus co-occurrences to build such a model. However, by grouping the original contexts into structurally similar classes, we can aggregate the co-occurrence data to a point where comparison among the words becomes possible. The only path to learning about inter-word dependencies lies through these associative characteristics of the words. By condensing the space of co-occurrence data, we can begin to see usage of different words in similar, but no longer necessarily identical, contexts. This similarity of behavior will allow us to derive the classification of words into classes of usage, and allow us to identify those groups of words which follow similar patterns of constraints and selectional restrictions. It will also allow us to build a more accurate model of word use, and to be better
able to describe and predict the combinations and usage of words.

What’s next?

The following chapter examines a number of structural similarity metrics that can be used to organize the enormous space of contexts. By comparing the measures on corpus data to the same measures on simulated data, we can observe the degree to which these similarities are present in the real language data, and the degree to which the measures are able to extract and identify them.

The chapter following used these measures to build a set of context classes. The resulting classes are evaluated in terms of their predictive capacity and in terms of their linguistic relevance.