# Everything You Always Wanted to Know About Probability Theory, But Were Afraid to Ask*

David M. Magerman
Stanford University
magerman@cs.stanford.edu

**Abstract**

With the renewed interest in probabilistic language modeling and analysis, there is a need for an introduction to probability theory that doesn't assume a strong mathematical background, but which gives a sufficiently detailed account of the subject to provide the reader a working knowledge of probabilities. This paper is an attempt to provide that introduction. More precisely, the goals of this paper are (1) to introduce to the reader a correct usage of the notation found in probability theory, (2) to describe the basic theorems and definitions of probability theory, (3) to superficially discuss the advanced topics that are relevant to CL, (4) and to provide a few examples of how all of the above items can be applied to CL. Along the way, I will also point out some of the ways in which current probabilistic CL applications violate or misuse probability theory.

## 1 Introduction

One of the most overwhelming aspects of studying computational linguistics (CL) is the cross-disciplinary nature of the field. In order to begin to understand the current theories in computational linguistics, you need not only have a working knowledge of computer science and linguistics, but also of psychology, philosophy, mathematics, and, as of late, probability theory. How can a student (or a senior researcher, for that matter) keep up with the latest ideas when there is so much overhead involved in learning the related disciplines?

The answer to this question revolves around the interpretation of the term "working knowledge." In order to understand the applications of psychology to language acquisition, for instance, you need not understand or know of the existence of personality theory, behavior theory, or abnormal psychology. And, once you overcome the terminology barrier, you can

*I would like to give credit where credit is due. The contents of this paper (except for the errors) are a direct result of the teaching and guidance of Prof. Max Mintz at the University of Pennsylvania. His patience and instruction has led me, along with many other students, to a deeper understanding of probability theory.

quickly acquire a sufficient familiarity with the elements of psychological research relevant to the study of language.

However, in the case of acquiring a "working knowledge" of probability theory, there is much more to be concerned with. The notation used in probability theory can be cumbersome and imposing, but it is at the same time necessary. And learning the notation is only the beginning. Probability theory is based on a rigorous set of theorems and principles. A "working knowledge" of probability theory requires at least a familiarity with, if not an understanding of, these underlying theorems and principles. But, there are far too many to learn and only a few which are relevant to computational linguistics.

With the renewed interest in probabilistic language modeling and analysis, there is a need for an introduction to probability theory that doesn't assume a strong mathematical background, but which gives a sufficiently detailed account of the subject to provide the reader a "working knowledge." This paper is an attempt to provide that introduction.

More precisely, the goals of this paper are

- to introduce to the reader a correct usage of the notation found in probability theory,
- to describe the basic theorems and definitions of probability theory,
- to superficially discuss the advanced topics that are relevant to CL,
- and to provide a few examples of how all of the above items can be applied to CL.

Along the way, I will also point out some of the ways in which current probabilistic CL applications violate or misuse probability theory.

These goals appear ambitious, but in fact they are not. I do not propose to teach probability theory. I only hope to provide a sufficient introduction so that computational linguists can understand the way probability theory is being applied to their field and so that it will not be so imposing a task to pick up a probability textbook and start reading.

If you have ever picked up a probability textbook, you will know that many of the introductory examples involve dice, coins, and playing cards. Since this paper is aimed at computational linguists, I will use language-oriented examples whenever possible.

# 2 Terminology

The first barrier encountered when reading a probability textbook is the jargon. Most textbooks assume fluency in mathematical terminology, and new concepts are generally defined by terms with which you may not be familiar. Here, I will define the concepts and terms which are necessary for discussing probabilities.

## 2.1 Set Operations and Notation

Set theory is one of the building blocks of probability theory, so some familiarity with the basic set operations and notation is necessary.

**Universe** $\Omega$ usually denotes the universe, the set of all elements of the domain.

**Empty Set** $\emptyset$ denotes a set with no elements.

**Union and Intersection** If $A$ and $B$ are sets, the union of $A$ and $B$, $A \cup B$, is the set containing all of the elements that are in $A$ or $B$. The intersection of $A$ and $B$, $A \cap B$, is the set containing only those elements that are in both $A$ and $B$.

Given a sequence of $n$ sets, $A_1, A_2, \ldots, A_n$,

$$\bigcup_{i=1}^{n} A_i = A_1 \bigcup A_2 \bigcup \cdots \bigcup A_{n-1} \bigcup A_n \tag{1}$$

$$\bigcap_{i=1}^{n} A_i = A_1 \bigcap A_2 \bigcap \cdots \bigcap A_{n-1} \bigcap A_n \tag{2}$$

**Element of** $a \in A$ means that $a$ is an element of $A$.

**Subset** If $A$ is a subset of $B$, $A \subset B$, then every element of $A$ is also an element of $B$.

**Complement** $A^c$ is the set of all elements in $\Omega$ that are not in $A$.

## 2.2 Trial

A *trial* is the experimental unit of probability theory. It represents one occurrence of the phenomenon you are modeling.

For example, in speech recognition, a trial is the utterance of a sequence of words. The goal of a probabilistic model of speech, then, is to use probabilistic information to determine the words associated with an utterance.

In probabilistic parsing, a trial is also a spoken or written sentence; however, it is a word string together with a syntactic structure, and the goal of the model is to determine that structure.

In some cases, models break these objects into smaller pieces and use the parts as trials. For instance, Church [5] uses a trigram[1] part-of-speech model to assign words to part-of-speech categories. In this case, a trial is a sequence of three words which are to be categorized.

## 2.3 Event

An *elementary event* is any value which the can result from a trial. *Events* are sets of elementary events. If, in the domain of probabilistic context-free parsing, the trials are sentences to be parsed, then the events in this domain are the sets of context-free rules associated with the parse trees of the trial sentences.

[1] An $n$-gram is a sequence of $n$ things. A unigram is an individual item, a bigram is pair of items, and so on.

## 2.4 Discrete vs. Continuous Sets

From set theory, a *discrete* set is a set which is either finite or countably infinite[2]. For the sake of brevity, a *continuous* set is one which is not discrete.[3]

## 2.5 Disjoint Sets and Partitions

*Disjoint* sets are sets which have no elements in common. So, sets $A$ and $B$ are disjoint if $A \cap B = \emptyset$.

A sequence of sets $A_1 \ldots A_n$ are considered *pairwise disjoint* if no element is in more than one set. Pairwise disjoint sets can be viewed as bins, where no object can be in two bins. Formally,

$$\forall_{A_i, A_j, (i \neq j)} A_i \cap A_j = \emptyset. \tag{3}$$

A sequence of sets $A_1 \ldots A_n$ form a *partition* of a set $A$ if $A_1 \ldots A_n$ are pairwise disjoint and $\cup_{i=1}^{n} A_i = A$. In other words, a partition of $A$ is formed by taking all of the elements of $A$ and putting them into bins.

# 3 Basic Elements

Any discussion of probability theory should begin with a probability space (p-space). It is disturbing, then, that few (if any) of the probabilistic CL applications define the probability space upon which their application is based.

Probability spaces are made up of three parts: an event space $\Omega$; a $\sigma$-field (sigma-field) $\mathcal{F}$; and a probability function $\mathcal{P}$.

## 3.1 The Event Space, $\Omega$

The event space, usually denoted by $\Omega$, is the set of all of the events which can happen in the domain.

One problem in computational linguistics which lends itself to a simple application of probability theory is the task of assigning unknown words to part-of-speech categories (where an unknown word is a word not in a lexicon of known words). The event space for this task is the set of open-class categories, i.e.

$$\{\text{noun, verb, adjective, adverb, } \ldots\}. \tag{4}$$

Event spaces are classified as either discrete or continuous. Both types of spaces occur in CL. The set of open-class categories is an example of a discrete space. In speech recognition,

[2] An infinite set is countably infinite if there is a one-to-one mapping from that set to the positive integers.

[3] This definition of continuous is not precisely correct. However, the precise definition requires more formal mathematics than I want to include. This definition should be sufficient for all of the sets which you encounter in computational linguistics.

the set of possible pitch signal levels is a continuous set, since analog pitch signal levels are real-valued. However, the set of analog pitch signal levels within a certain range rounded to a finite number of decimal places is a discrete set, since the set is restricted to a finite number of values.

Discrete sets are far more common in computational linguistics, since it is difficult to implement and manipulate a continuous model efficiently. Thus, the remainder of the paper will restrict itself to discussions of discrete-event spaces.

### 3.2 The $\sigma$-field, $\mathcal{F}$

A $\sigma$-field of a set $\Omega$ is a collection of sets,

$$\{S_1 \ldots S_k \ldots\}(\text{where } S_i \subset \Omega), \tag{5}$$

which obeys three rules:

- $\emptyset \in \mathcal{F}$
- $A \in \mathcal{F}$ implies $A^c \in \mathcal{F}$
- $A_i \in \mathcal{F}$ implies $\bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$

The domain of the probability function in a p-space must be a $\sigma$-field. Note that the $\sigma$-field in a p-space includes *sets* of elementary events. In particular, it includes the sets of elementary events which you might want to know probability of.

For instance, you might be interested in knowing the probability of a word being either a noun or a verb. So, the set {noun, verb} should be an element of the $\sigma$-field.[4]

### 3.3 The Probability Function, $\mathcal{P}$

The final part of a p-space is the probability function, $\mathcal{P}$. While $\mathcal{P}$ is generally described in probability-oriented CL papers, it is rarely completely defined.

A probability function, $\mathcal{P}$, is a function which maps the p-space's $\sigma$-field onto the set of real numbers from 0 to 1:

$$\mathcal{P} : \mathcal{F} \to [0, 1] \tag{6}$$

In most CL applications, the elements of $\mathcal{F}$ about which we are most concerned are those containing elementary events, i.e. {noun} or {verb}. But, when reading about probability functions, you should be aware that $\mathcal{P}$ can be applied to larger sets of these events as well.

The probability function, $\mathcal{P}$, is the most notationally abused and overloaded function in all of probability theory. In a typical paper, the function $\mathcal{P}$ will refer to any number of

[4]Note that the set {noun, verb} denotes the event of either a noun occurring or a verb occurring, *not* the event of a word being in the class of words which can be both a noun *and* a verb (i.e. fly, show, dance, etc.). If that event is of interest, a category for that elementary event (i.e. noun-verb) must be explicitly included in the event space.

different functions with different domains. Frequently, different probability functions will be referred to as $\mathcal{P}$ in the same equation.

For instance, in one of my earlier papers ([14]), I define a statistical function as

$$\frac{\mathcal{P}(x_1 \dots x_n)}{\mathcal{P}(x_1 \dots x_k)\mathcal{P}(x_{k+1} \dots x_n)}. \tag{7}$$

There are potentially 3 unique probability functions referred to here: a function mapping a sequence of $n$ events onto $[0,1]$, a function mapping a sequence of $k$ events onto $[0,1]$, and a function mapping a sequence of $n-k$ events onto $[0,1]$. To be more precise, I could have defined this function as

$$\frac{\mathcal{P}_{\Omega^n}(x_1 \dots x_n)}{\mathcal{P}_{\Omega^k}(x_1 \dots x_k)\mathcal{P}_{\Omega^{(n-k)}}(x_{k+1} \dots x_n)}. \tag{8}$$

In $\mathcal{P}_X$, $X$ is understood to be the event space which corresponds to the probability function.

The abbreviated definition (7) is considered acceptable, since the more precise version (8) is far less readable, and the domain of the function is discernible from the arguments to the function. However, when the arguments do not disambiguate the function, unique function descriptors should be used.[5]

The remainder of this paper will be a discussion of various types of probability functions.

# 4 Estimating $\mathcal{P}$

The probability of an event can be interpreted as the relative frequency with which that event occurs in a sequence of independent trials.[6]

A simple example of probability function is one for which every elementary event in $\Omega$ is equally likely. In this case, the probability of each elemntary event is $\frac{1}{n}$, where $n$ is the number of elements in $\Omega$, and the probability of an event (set of elementary events) $A$ is the ratio of the size of $A$ to the size of $\Omega$.

If we lived in a world in which each elementary event was equally likely, probability theory would be much simpler (but life would be far more unpredictable). Especially in computational linguistics, we rarely encounter such simple probability functions.

For instance, consider the probability function for the unknown word categorization problem. Let's call this function $\mathcal{P}_T(t)$, which represents the likelihood the category of an unknown word is in the set $t$, where $t \subset T$. If $\mathcal{P}_T(\{\text{noun}\}) = 0.5$, then half of the unknown words in a sufficiently large corpus of samples are expected to be nouns.

Only in the simplest cases, e.g. fair dice, random decks of cards, are the exact probability functions known from combinatorial arguments. In general, $\mathcal{P}$ must be estimated.

There are two common techniques for estimating a probability function. The first is to find a family of distributions whose behavior is similar to that of the probability function.

[5]In general, an abbreviation should only be used if you and your readers know what the abbreviation represents.

[6]An independent trial is one which does not depend on the outcomes of previous trials. Independence will be discussed in more detail later.

The second is to empirically estimate the values of the function at each point (or at many points).

The first technique is a potentially powerful but also potentially dangerous way of estimating a probability function. Each family of distributions has a parameterized formula for its probability functions. By specifying the parameters associated with your function, you completely define the probability function. If the domain of the probability function is a continuous set, this is usually the best technique to use, since you can only empirically estimate a function at a finite number of points. The advantage of this technique is that once you define a function as belonging to a certain family, you can apply all of the theorems associated with that family to the function. The disadvantage is that probability distributions encountered in the world rarely coincide with the known families of distributions. If you incorrectly assume the distribution is in a certain family, the resulting probability function will produce a poor approximation of the phenomena you are modeling. Further, the theorems associated with that family may not apply to your problem. When a family's theorems are inappropriately applied, wildly inaccurate conclusions about the domain can result.

The second technique is to empirically estimate the probability function. This technique is more accurate when applied to discrete distributions, although it can be used to interpolate continuous distribution functions with lower precision.

This technique requires a set of trials or examples of events from the event space. For estimating the probability function for the unknown word categorization problem discussed above, you might want to collect a large number of unknown words from a corpus of sentences, and manually assign those words to categories. The estimate for an unknown word being assigned to category $t$ from the set of categories $T$ might be defined to be

$$\frac{freq(t)}{\sum\limits_{t' \in T} freq(t')}.^{7} \tag{9}$$

The accuracy of an empirical estimate of a probability function is related to the size of the example set used. It is also important to use an set of examples which is representative of the domain which you are modeling. For instance, if you are modeling the constituent structure of modern English, using the works of Shakespeare as an example set will produce an inaccurate (though perhaps eloquent) model.

Even if given an appropriate set of examples, the empirically estimated function will undoubtedly be inaccurate to a certain degree. The goal of this estimation process is to specify a function which approximates the actual probability function *closely enough* to make the estimated probability function useful in modeling future events.

A number of techniques have been developed to refine the empirical estimation process by modeling the error associated with the probability estimates, and adjusting the estimates accordingly. Some of these techniques have been applied to CL ([13], [6], [10], [11]).

[7] $\sum_{\omega \in \Omega} f(\omega)$ is the sum of the values of $f$ evaluated for all of the elements in $\Omega$.

# 5 Properties of $\mathcal{P}$

The probability function isn't just any function that happens to have a $\sigma$-field as a domain. It also has to obey three rules:

- $\mathcal{P}(A) \geq 0$ for all elementary events $A \in \Omega$.
- A trial will always produce an event in the event space. That is, the probability that something in the event space will happen, namely $\mathcal{P}(\Omega)$, is 1.
- If $A_1, A_2, \ldots$ are pairwise disjoint sets in $\mathcal{F}$,

$$\mathcal{P}(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} \mathcal{P}(A_i). \tag{10}$$

As long as these two rules hold, $\mathcal{P}$ is a probability function.

Applying algebra and a little set theory, these two rules generate a number of important properties of probability functions. For instance:

- $\mathcal{P}(\emptyset) = 0$, where $\emptyset$ is the empty set.
- If $A \subset B$, then $\mathcal{P}(B) \geq \mathcal{P}(A)$.
- $0 \leq \mathcal{P} \leq 1$.

A number of other elementary properties also result from the definition of a probability function, but they are less relevant to our concerns.

## 5.1 $\sum_{\omega \in \Omega} \mathcal{P}(\omega) = 1$

It is obviously important that a probability estimator exhibit all of the elementary properties of a probability function. However, the first property is easy to verify (especially when the event space is finite), and it is frequently ignored in practice.

For instance, in an infinite domain, every event can have a zero probability. There is nothing wrong with this, since zero probability events *can* occur.

To illustrate this, pick a number at random. By "at random," I mean that each number has a probability of being picked equal to $\frac{1}{n}$, where $n$ is the size of the event space. Here, since $n$ is infinitely large, this probability is zero. Let's say you selected 1,425,678,901,647,123,669. That number had a zero probability of being picked, and yet it was. So, while it seems contradictory to the rules, it is possible to have a probability space with an infinite domain where every event has a zero probability.

Assigning a *non-zero* probability to every event in an infinite space also seems to violate the elementary principles. However, this also is not true. Consider a probability function which assigns the probability $\frac{1}{2^i}$ to the event $\omega_i$, for all $\omega_i \in \Omega$. So, the first event has probability $\frac{1}{2}$, the second $\frac{1}{4}$, the third $\frac{1}{8}$, and so on. If $\Omega$ is a countably infinite set, then the sum of the probabilities of each elementary event is 1.

In practice, you will be provided only a finite number of trials. As long the sum of the probabilities of the events which have occurred in the given trials is less than or equal to 1, the elementary principle is satisfied.

In a discrete domain, events rarely have zero probability. Consider the unknown word categorization problem. Assume that the corpus used to estimate the probability function contained 50 unknown words, including 25 nouns, 20 verbs, and 5 adjectives. Using a simple estimation technique, you might define your probability function as:

$$\mathcal{P}_T(t) : \begin{cases} 0.5 & \text{if } t = \{ \text{ noun } \} \\ 0.4 & \text{if } t = \{ \text{ verb } \} \\ 0.1 & \text{if } t = \{ \text{ adjective } \} \end{cases} \tag{11}$$

Now, assume you encounter an unknown adverb. According to the estimate,

$$\mathcal{P}_T(\{ \text{ noun, verb, adjective } \}) = 1. \tag{12}$$

In order to maintain the elementary principles, the probability that an unknown word is an adverb must be zero. If you estimate $\mathcal{P}_T(\{\text{adverb}\})$ to be anything but zero, you are in violation of the elementary principles, and the laws of probability theory no longer apply to the use of $\mathcal{P}_T$.

## 5.2 Disjoint Events and Adding Probabilities

Something which you frequently need to do is calculate the probability of a set of events from the probabilities of two parts of that set. (If you have never needed to do this, *you will tomorrow,* so pay attention.) If the two parts of that set are disjoint, then this task is simple. If $A$ and $B$ are disjoint sets, then $\mathcal{P}(A \cup B) = \mathcal{P}(A) + \mathcal{P}(B)$.

For instance, let's say you want to know the probability that an unknown word will be a verb, but you only have categories for active verbs and passive verbs. You have already estimated $\mathcal{P}(\{\text{active verb}\})$ and $\mathcal{P}(\{\text{passive verb}\})$. What you need to know is $\mathcal{P}(\{\text{active verb, passive verb}\})$. Since a verb cannot be both passive and active, the sets of active verbs and passive verbs are disjoint. Also, a verb must be either passive or active. So,

$$\mathcal{P}(\{\text{verb}\}) = \mathcal{P}(\{\text{active verb, passive verb}\}) = \mathcal{P}(\{\text{active verb}\}) + \mathcal{P}(\{\text{passive verb}\}). \tag{13}$$

Now, consider the problem of determining the probability that an unknown word will be a noun, given that you have categories for singular nouns and animate nouns. Since animate nouns can be singular, these sets are not disjoint. Furthermore, there are nouns which are neither singular nor animate, so the union of these two sets do not equal the set of nouns. Thus, to solve this problem, you can not use the sum of $\mathcal{P}(\{\text{singular noun}\})$ and $\mathcal{P}(\{\text{animate noun}\})$, nor can you use any function of those two probabilities without further information.

# 6 Joint Probabilities and Conditional Probabilities

Joint probability functions and conditional probability functions describe the anticipated behavior of two events in a given trial. Their definitions are intertwined, since either one can be defined in terms of the other. I will attempt to define them independently, and then present the formula which relates them.

## 6.1 Joint Probability Functions

Joint probability functions are probability functions which describe the likelihood of two events occurring. Given a p-space with domain $X$ and probability function $\mathcal{P}_X$, and another p-space with domain $T$ and probability function $\mathcal{P}_Y$, the joint probability function, denoted $\mathcal{P}_{X,Y}$, represents the probability of a set of elementary events from $X$ and a set of elementary events from $Y$ occurring. The probability functions for the individual events, $\mathcal{P}_X$ and $\mathcal{P}_Y$, are referred to as *marginal* probability functions.

Once again, consider the unknown word categorization problem. Let's call the domain of that problem $T$, namely the set of open-class categories. Consider also the problem of determining which context-free rule from a set of rules describes the constituent of which that word is a part. We'll call the domain of the second problem $G$, namely the set of context-free grammar rules.

The probability that an unknown word is a noun and is the head of the rule, "NP $\leftarrow$ determiner noun," is described by the joint probability function, $\mathcal{P}_{T,G}$. In particular, this value is denoted by:

$$\mathcal{P}_{T,G}(\{\text{noun}\}, \text{NP} \leftarrow \text{determiner noun}) \tag{14}$$

This joint probability might be useful, for example, in evaluating the likelihood of a particular parse of a sentence containing that word. Notice that the category we assign the unknown word determines which rules in $G$ can describe the word's constituent. At the same time, the rule which describes the word's constituent significantly narrows down the possible categories for that word. This leads us to the next topic, conditional probabilities.

## 6.2 Conditional Probabilities

Conditional probability functions are useful for using partial information about the outcome of a trial to predict information about the rest of it. In assigning categories to unknown words, you might decide that the categories of the words adjacent to that word would be helpful, or predictive. For instance, the probability that a word is an adjective if it falls between a determiner and a noun is very high, even though the probability that a word is an adjective in general is much lower. And, the probability that a word is a noun if it comes between a pronoun and a verb is very low, even though noun is the most likely classification of an unknown word, in general.

This problem invites a conditional probability function. Let's call the sequence of the categories of the three adjacent words $t_1t_2t_3$, with the unknown word at the center. Let the domain of $t_1$ and $t_3$ be the set of all terminal categories, $T^*$, and let the domain of $t_2$ be the set of all open class categories $T$. The conditional probability function of the category of the middle word given the categories of the adjacent words is denoted by

$$\mathcal{P}_{T_2|T_1^*,T_3^*}(t_2|t_1,t_3).^8 \tag{15}$$

Based on the example given above, we might say that

$$\mathcal{P}_{T_2|T_1^*,T_3^*}(\{\text{adjective}\}|\{\text{determiner}\},\{\text{noun}\}) > \mathcal{P}_T(\{\text{adjective}\}) \tag{16}$$

## 6.3 Independence and Dependence of Two Events

Knowledge of how one event impacts on the likelihood of another is very useful in determining joint probabilities and conditional probabilities. There are three classifications for the relatedness of two events: completely dependent, independent, and dependent (i.e. somewhere in-between).

If one event is completely dependent on another event, then in every trial that the second event occurs, the first occurs also. For instance, if event $x$ is completely dependent on event $y$, then if you know that $y$ has happened, you also know that $x$ has happened. Using conditional probabilities, $x$ completely depends on $y$ implies that

$$\mathcal{P}_{X|Y}(x|y) = 1 \tag{17}$$

and

$$\mathcal{P}_{X|Y}(x^c|y) = 0 \tag{18}$$

This is not a very practical result in computational linguistics, since one event rarely *completely* defines another in natural language. However, to illustrate dependence, I'll contrive an example.

Assume you are given a language in which there is only one syntactic structure for questions: wh-word verb determiner noun. Assume also that this language has a context-free grammar which contains, among others, the rules S $\leftarrow$ question, question $\leftarrow$ wh-word VP, VP $\leftarrow$ verb NP, and NP $\leftarrow$ determiner noun. In this grammar, all of these rules are completely dependent on a question event. Thus,

$$\mathcal{P}_{G|G}(\{\text{VP} \leftarrow \text{verb NP}\}|\{\text{S} \leftarrow \text{question}\}) = 1 \tag{19}$$

and

$$\mathcal{P}_{G|G}(\{\text{NP} \leftarrow \text{adjective noun}\}|\{\text{S} \leftarrow \text{question}\}) = 0. \tag{20}$$

On the other hand, two events are considered independent if the occurrence of one event has no impact on the likelihood of occurrence of the other event. In the case of independent events,

$$\mathcal{P}_{X|Y}(x|y) = \mathcal{P}_X(x). \tag{21}$$

[8] This can be read: "the probability of $t_2$ given $t_1$ and $t_3$."

Since there are fewer data points to estimate in a marginal probability function than in a conditional probability function, knowing that two events are independent is very useful information. *However,* while it is desirable to assume two events are independent, if that assumption is incorrect, significantly inaccurate probability estimates can result.

Consider the problem of determining the context-free rules which form the parse tree of a sentence. Since it is easier to calculate the probability of individual rules being in the parse tree for a sentence than it is to calculate the probability of the entire parse tree, it would be a great stroke of luck if the events of rules occurring together in a parse tree were independent. Sadly, in most cases, these events are not independent. Consider the case of the rules NP $\leftarrow$ pronoun, S $\leftarrow$ NP VP, and VP $\leftarrow$ V NP. It is an empirical observation that subject pronouns are far more common than object pronouns in English. This information can be represented as

$$\mathcal{P}_{G_r|G_p}(\text{NP} \leftarrow \text{pronoun} \mid \text{S} \leftarrow \text{NP VP}) > \mathcal{P}_{G_r|G_p}(\text{NP} \leftarrow \text{pronoun} \mid \text{VP} \leftarrow \text{V NP}), \qquad (22)$$

where $G_r$ is the event of a rule and $G_p$ is the event of the parent rule, i.e. the rule one step closer to the root of the tree.

But, if the rules in a parse tree are independent, then

$$\begin{aligned} \mathcal{P}_{G_r}(\text{NP} \leftarrow \text{pronoun}) &= \mathcal{P}_{G_r|G_p}(\text{NP} \leftarrow \text{pronoun} \mid \text{S} \leftarrow \text{NP VP}) & (23) \\ &= \mathcal{P}_{G_r|G_p}(\text{NP} \leftarrow \text{pronoun} \mid \text{VP} \leftarrow \text{V NP}) & (24) \\ & & (25) \end{aligned}$$

which contradicts 22. There are many such examples of rules in a parse tree which are dependent. For instance, the parse trees of two conjoined phrases are dependent, since more similar structures are more likely to be conjoined than very different structures. Finally, most (if not all) CFGs contain rules for handling movement and inversion in questions. The events of these movement and inversion rules are also dependent. In fact, they are likely to be nearly completely dependent, since inversion rules predict questions, and questions in turn predict movement rules.

I have not said much here about the estimation of joint probabilities and conditional probabilities when the marginal probabilities are neither completely dependent nor independent. I will reserve that discussion for a later section, in which I will speculate on possible methods for making such an estimation for problems in the CL domain using linguistic information.

## 6.4 The Multiplication Rule

The multiplication rule dictates the relationship between the joint and conditional probability functions. The multiplication rule states that

$$\mathcal{P}_{Y|X}(y|x) = \frac{\mathcal{P}_{X,Y}(x,y)}{\mathcal{P}_X(x)} \qquad (26)$$

or, equivalently,

$$\mathcal{P}_{X,Y}(x,y) = \mathcal{P}_X(x)\mathcal{P}_{Y|X}(y|x). \qquad (27)$$

By the multiplication rule, you can determine the conditional probability of a CF rule, $g$, describing an unknown word's constituent given that the unknown word's category is $t$ :

$$\mathcal{P}_{G|T}(g|t) = \frac{\mathcal{P}_{G,T}(g,t)}{\mathcal{P}_T(t)}. \tag{28}$$

The multiplication rule can be applied to the product of the joint probability of three events as well. Church provides a model by which the unknown word categorization problem can be solved using the conditional probability of the categories of the adjacent words. A model of probabilistic parsing, such as the context-sensitive probability model in $\mathcal{P}$earl ([15]), must solve the problem of determining the CF rule which describes that unknown word's constituent. If the parser's decision were conditional on the categories of the words adjacent to the unknown word, then the probability of the CF rule given the word's category and the categories of the words adjacent to it can be defined as

$$\mathcal{P}_{G,T_1^*,T_2,T_3^*}(g,t_1,t_2,t_3) = \mathcal{P}_{G|T_1^*,T_2,T_3^*}(g|t_1,t_2,t_3)\mathcal{P}_{T_2|T_1^*,T_3^*}(t_2|t_1,t_3)\mathcal{P}_{T_1^*,T_3^*}(t_1,t_3), \tag{29}$$

or, using abbreviated notation,

$$\mathcal{P}(g,t_1,t_2,t_3) = \mathcal{P}(g|t_1,t_2,t_3)\mathcal{P}(t_2|t_1,t_3)\mathcal{P}(t_1,t_3). \tag{30}$$

## 6.5 Joint Probability Estimation and Independence Issues

Joint probability functions are useful in computational linguistics because they allow the prediction of more than one phenomenon. It should be clear now that it is more effective to determine the category of a word and its constituent's rule at the same time than to decide the word's category without taking the grammar into account.

When there is enough data to estimate each point of a joint probability function, joint probabilities functions can be powerful and accurate modeling tools. But, frequently there is not enough data. For instance, the joint probability function in the context-sensitive probability model from the previous section needs to estimate joint probabilities for every possible rule/trigram combination which can occur in the language. Given a CF grammar with 100 rewrite rules and 50 part-of-speech categories, and there are 12.5 million possible rule/trigram combinations. Certainly many of the trigrams do not occur in the language (i.e. noun determiner verb), and many trigrams are ungrammatical when combined with certain rules (i.e NP $\leftarrow$ noun, pronoun verb preposition). However, there are still too many events for which to calculate probabilities, since each possible rule/trigram event must occur *many* times in a corpus in order to empirically estimate the joint probabilities accurately.

An alternative to empirical estimation is to find a known family of distributions which behaves similarly to your joint probability function. If the joint distribution belongs to a parameterized family of distributions, or if the marginal distributions belong to certain families, there are theorems for those families about how to estimate the joint probability from the marginals. However, in the example above, there is no evidence that any of the probability functions, joint or marginal, belong to known families.

As we discussed above, the reason we want to estimate a conditional or a joint probability is that these marginal distributions are *not* independent. And if we cannot assume independence, probability theory does not dictate any method for estimating joint probabilities. In their illuminating paper, "Poor Estimates of Context are Worse than None" [10], Church and Gale demonstrate how invalid assumptions in the probability estimation process can produce results that are worse than having a random model (i.e. guessing at the answer). Based on this result, if there is not enough data to estimate a joint probability function for two events which are not independent, the best alternative is to pick a different model.

# 7 Case Study: Joint Probability Estimates with Sparse Data

I have quite clearly stated that you should avoid at all costs trying to estimate joint probability functions using marginal probability functions of dependent events. Now, I will contradict myself and discuss how to estimate such a joint probability function.

In the last section, I stated that the best alternative in the face of dependent marginal distributions is to pick another model. However, what if there is no other model that doesn't suffer the same fate of dependence? What if you are trying to model two related phenomena, but sparse data prevents the joint probability estimation?

When probability theory fails, linguistic intuition must take over. Consider the problem of determining the joint probability of all of the rules $g_1 \ldots g_n$ in a parse tree of a sentence. Based on the principles of probability functions, there are two clearly defined cases where joint probabilities can be calculated from marginals: when the rules $g_1 \ldots g_n$ are independent, and when the rules $g_1 \ldots g_{k-1} g_{k+1} \ldots g_n$ are completely dependent on $g_k$.

If all of the rules are independent, then the probability of a parse tree is merely the product of the marginal probabilities of the rules in the parse tree:

$$\mathcal{P}_{G_1 \ldots G_n}(g_1 \ldots g_n) = \prod_{i=1}^{n} \mathcal{P}_{G_i}(g_i) \tag{31}$$

In the case of complete dependence, the joint probability of all of the rules would be $\mathcal{P}_{G_k}(g_k)$. Since the the rule $g_k$, upon which all others are dependent, can not be determined *a priori,* an overestimate of the joint probability of the rules is the maximum marginal probability:

$$\prod_{i=1}^{n} \mathcal{P}_{G_1 \ldots G_n}(g_1 \ldots g_n) = \max_{i=1}^{n} \mathcal{P}_{G_i}(g_i) \tag{32}$$

If, using linguistic knowledge, we hypothesize that rules in a parse tree tend to be nearly completely dependent, i.e. a rule in a parse tree is highly predictive of those around it, then we could estimate that the joint probability of these rules will be greater than the product of the marginal probabilities, but less than the maximum marginal probability:

$$\prod_{i=1}^{n} \mathcal{P}_{G_i}(g_i) \leq \mathcal{P}_{G_1 \ldots G_n}(g_1 \ldots g_n) \leq \max_{i=1}^{n} \mathcal{P}_{G_i}(g_i). \tag{33}$$

If we believe that the rules in a parse tree are generally more dependent than independent, we would guess that the joint probability is closer to the maximum marginal probability than to the product of marginal probabilities. One function which fits this description is the product of maximal marginal rule probability and the geometric mean[9] of the remaining rule probabilities:

$$\mathcal{P}_{G_1 \ldots G_n}(g_1 \ldots g_n) \sim \mathcal{P}_{G_m}(g_m) \left( \prod_{i=1, i \neq j}^{n} P_{G_i}(G_i) \right)^{\frac{1}{n-1}}, \tag{34}$$

where $\mathcal{P}_{G_m}(g_m)$ is the maximum marginal rule probability.

There is no empirical justification for this formula over any other estimation that falls between the two endpoints. If we could estimate the entire joint distribution, we could confirm or reject this estimate. But, the assumption of this discussion is that such an estimate is unfeasible for all of the points in the function because of insufficient data. There is no reason, however, why we can't estimate the points in the joint distribution corresponding to high frequency events. Given a few empirically calculated points in the joint distribution, we could then determine the accuracy of a candidate estimator function.

For the sake of an experiment, let us restrict our domain to subtrees of depth 3. Consider the $N$ highest frequency events in this domain, i.e. the $N$ trees of depth 3 which occur most frequently in a corpus of parse trees. Estimate the joint probability of the rules in these trees by the empirical analysis and compare that result to the function described in the previous paragraph evaluated on the same trees. Applying this type of experiment to a number of different joint estimator functions, we could at least determine the relative fitness of the estimator functions. Certainly, linguistically motivated estimates which are accurate on high frequency events are preferable to inaccurate estimates based on unmotivated assumptions.

# 8 Case Study: Hidden Markov Models

Another type of probabilistic model used in CL is the Hidden Markov Model (HMM). HMMs are attractive models for linguistic events because they presume locality in determining probability, and thus do not suffer from the sparse data problems of more context-sensitive models.

The problem which HMMs solve is that of estimating the joint probability of a sequence of $n$ events. The crucial assumption of HMMs is that the probability of the sequence of events is only dependent on a shorter subsequence of those events, i.e.[10]

$$\mathcal{P}(x_{n+1} | x_1 x_2 \ldots x_n) = \mathcal{P}(x_{n+1} | x_{n-k} x_{n-k+1} \ldots x_n) \tag{35}$$

where $k$ is a constant smaller than $n$.

[9] The geometric mean of $x_1 \ldots x_n$ is the $n$th root of their product:$(\prod_{i=1}^{n} x_i)^{\frac{1}{n}}$ .

[10] For this discussion, we will dispense with the subscripting of probability functions, unless the function's domain is ambiguous.

Part-of-speech assignment[11] is a task which can be accurately modeled using HMMs. This may seem surprising, since syntactic decisions can depend on structures which span many words in a sentence. As a result of long-distance dependencies, garden path sentences, like "The horse raced past the barn fell," are likely to be misinterpreted by a HMM part-of-speech tagger. However, it turns out that garden path sentences do not occur very frequently. And, further, there are relatively few cases where long-distance information is necessary to override local information to prevent a mistaken part-of-speech assignment. As a result, HMM part-of-speech tagging has received much recognition recently in the CL community.

The idea behind the HMM part-of-speech model is based on a locality assumption that the probability of a word being assigned to a part-of-speech category depends only on the word and the parts of speech of the two words preceding it. The model assumes that this part-of-speech assignment decision is independent of the remaining words and parts-of-speech in the sentence. Formally,

$$\mathcal{P}(t_k|w_1/t_1 w_2/t_2 \ldots w_k \ldots w_n/t_n) = \mathcal{P}(t_k|w_k t_{k-1} t_{k-2}), \tag{36}$$

where $t_i$ is the part-of-speech for the word $w_i$.

A further assumption is frequently made to avoid sparse data problems. This assumption is that the lexical probability of a word[12] is independent of the word's context. The validity of this assumption is less obvious; however, implementations of the model use this assumption with no significant loss of accuracy. This second assumption suggests that

$$\mathcal{P}(w_k|t_k t_{k-1} t_{k-2}) = \mathcal{P}(w_k|t_k). \tag{37}$$

The goal of this model is to find the part-of-speech sequence $t_1 \ldots t_n$ for a given word sequence $w_1 \ldots w_n$ which maximizes the joint probability function $\mathcal{P}(t_1 \ldots t_n, w_1 \ldots w_n)$. Assume that the word sequence has been padded with beginning-of-expression markers for $w_1$ and $w_2$, so that their probabilities are constant. Then, using the HMM independence assumptions and (37),

$$\begin{aligned}
\mathcal{P}(t_1 \ldots t_n, w_1 \ldots w_n) &= \mathcal{P}(w_n t_n|t_{n-1} t_{n-2}) \ldots \mathcal{P}(w_3 t_3|t_2 t_1) && (38)\\
&= \mathcal{P}(w_n|t_n t_{n-1} t_{n-2}) \mathcal{P}(t_n|t_{n-1} t_{n-2}) \ldots \mathcal{P}(w_3|t_3 t_2 t_1) \mathcal{P}(t_3|t_2 t_1) && (39)\\
&= \mathcal{P}(w_n|t_n) \mathcal{P}(t_n|t_{n-1} t_{n-2}) \ldots \mathcal{P}(w_3|t_3) \mathcal{P}(t_3|t_2 t_1) && (40)\\
&= \prod_{i=3}^{n} \mathcal{P}(t_i|t_{i-1} t_{i-2}) \mathcal{P}(w_i|t_i). && (41)
\end{aligned}$$

Bayes formula states that $\mathcal{P}(w_i|t_i) = \frac{\mathcal{P}(t_i|w_i)\mathcal{P}(w_i)}{\mathcal{P}(t_i)}$. Thus, (41) can be expressed as

$$\prod_{i=3}^{n} \frac{\mathcal{P}(t_i|t_{i-1} t_{i-2}) \mathcal{P}(w_i) \mathcal{P}(t_i|w_i)}{\mathcal{P}(t_i)}. \tag{42}$$

[11] The example used here is a detailed description the part-of-speech model proposed by Church [5]. It is similar to the models discussed in [17] and [16] as well.

[12] The lexical probability of a word is defined to be the percentage of time a word is used as a given part-of-speech.

Factoring out the $\mathcal{P}(w_i)$ terms and applying the multiplication rule,

$$\prod_{i=3}^{n} \mathcal{P}(w_i) \prod_{i=3}^{n} \mathcal{P}(t_i|w_i) \frac{\mathcal{P}(t_i t_{i-1} t_{i-2})}{\mathcal{P}(t_{i-1} t_{i-2}) \mathcal{P}(t_i)} \tag{43}$$

The first term in (43) can be factored out and eliminated, since it is a constant function of the word string, and will not affect the selection of the most probable part-of-speech sequence. So, finally, we have

$$\mathcal{P}(t_1 \ldots t_n, w_1 \ldots w_n) = \prod_{i=2}^{n} \mathcal{P}(t_i|w_i) \frac{\mathcal{P}(t_i t_{i-1} t_{i-2})}{\mathcal{P}(t_{i-1} t_{i-2}) \mathcal{P}(t_i)}. \tag{44}$$

This quantity can be calculated for each possible part-of-speech sequence in a sentence using a dynamic programming algorithm. Based on the derivation of this formula, when the HMM independence assumptions hold, the part-of-speech sequence for which (44) achieves its highest value is the correct part-of-speech sequence.

More importantly, the components of the equation represent probabilities of events which are feasible to empirically estimate with available corpora of examples. That is the idea behind HMMs: to break apart a problem into *relatively* independent components whose distributions are empirically available.

The part-of-speech HMM model is an example of an effective application of joint probability estimation which is accomplished by linguistically motivated assumptions. There *is* reason to believe that a decision such as part-of-speech disambiguation can be handled locally in many cases, so the independence assumptions upon which HMM models are based are justifiable in this model.

While HMMs have been successful in part-of-speech modeling, as well as in speech recognition, HMMs cannot be used to model all types of events. Recent attempts at developing HMM models for probabilistic context-free parsing have been less successful, precisely because the independence assumptions are far less motivated.

# 9 What Else?

There are many other theorems, properties and concepts relating to probability theory that haven't been mentioned in this paper. I didn't discuss random variables or probability density functions, and I only referred in passing to the parameterized families of distributions, such as normal, Bernoulli, and Poisson. And all but one of the examples involved finite discrete distributions, while a majority of the discussion in probability textbooks concerns continuous distributions.

These omissions are deliberate. The goal of this paper was to present the topics in probability theory which are most relevant to computational linguists, while employing the bare minimum of mathematical machinery. These omitted topics are important to probability theory, but are not as meaningful to the probabilistic models of CL.

If you feel cheated, as though you have only gotten half of the picture, then this paper has achieved its purpose. Having a "working knowledge" of probability theory is useful for understanding the recent work in probabilistic models; but, a more thorough coverage of probability theory can only enhance your ability to contribute to that body of work. If you are interested in a more formal view of probability theory, hopefully this paper has sufficiently motivated and prepared you to pick up a probability textbook ([8], [2]) and begin learning.

## References


[1] Ayuso, D., Bobrow, R., et. al. 1990. Towards Understanding Text with a Very Large Vocabulary. In Proceedings of the June 1990 DARPA Speech and Natural Language Workshop. Hidden Valley, Pennsylvania.

[2] Bickel, P. J. and Doksom, K. A. 1977. *Mathematical Statistics: Basic Ideas and Selected Topics,* Holden Day, Inc., Oakland, California.

[3] Brill, E., Magerman, D., Marcus, M., and Santorini, B. 1990. Deducing Linguistic Structure from the Statistics of Large Corpora. In Proceedings of the June 1990 DARPA Speech and Natural Language Workshop. Hidden Valley, Pennsylvania.

[4] Chitrao, M. and Grishman, R. 1990. Statistical Parsing of Messages. In Proceedings of the June 1990 DARPA Speech and Natural Language Workshop. Hidden Valley, Pennsylvania.

[5] Church, K. 1988. A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text. In Proceedings of the Second Conference on Applied Natural Language Processing. Austin, Texas.

[6] Church, K. and Gale, W. 1990. Enhanced Good-Turing and Cat-Cal: Two New Methods for Estimating Probabilities of English Bigrams. *Computer, Speech and Language.*

[7] Church, K. and Hanks, P. 1989. Word Association Norms, Mutual Information, and Lexicography. In Proceedings of the 27th Annual Conference of the Association of Computational Linguistics.

[8] DeGroot, M. H. 1986. *Probability and Statistics, Second Edition,* Addison-Wesley. Reading, Massachusetts.

[9] Francis, W. N. and Kucera., H. 1982. *Frequency Analysis of English Usage: Lexicon and Grammar,* Houghton Mifflin Company.

[10] Gale, W. A. and Church, K. 1990. Poor Estimates of Context are Worse than None. In Proceedings of the June 1990 DARPA Speech and Natural Language Workshop. Hidden Valley, Pennsylvania.

[11] Jelinek, F. and Mercer, R. 1980. Interpolated Estimation of Markov Source Parameters from Sparse Data. *Pattern Recognition in Practice.* North Holland Publishing Co., Amsterdam.

[12] Jelinek, F., Lafferty, J. D. and Mercer, R. L. 1990. Basic Methods of Probabilistic Context Free Grammars. IBM Report RC 16374 (#72684).

[13] Katz, S. M. 1987. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-35, No. 3.*

[14] Magerman, D. M. and Marcus, M. P. Parsing a Natural Language Using Mutual Information Statistics. In *Proceedings, Eight National Conference on Artificial Intelligence,* August, 1990. Boston, Massachusetts.

[15] Magerman, D. M. and Marcus, M. P. $\mathcal{P}$earl: A Probabilistic Chart Parser. In *Proceedings, European ACL,* April, 1991. Berlin, Germany.

[16] Merialdo, Bernard. Tagging Text With a Probabilistic Model. IBM Report.

[17] Meteer, M., Schwartz, R. and Weischedel, R. 1991. Studies in Part of Speech Labelling (sic). In Proceedings of the February 1991 DARPA Speech and Natural Language Workshop. Asilomar, California.

[18] Sharman, R. A., Jelinek, F., and Mercer, R. 1990. Generating a Grammar for Statistical Training. In Proceedings of the June 1990 DARPA Speech and Natural Language Workshop. Hidden Valley, Pennsylvania.