Structural Zeros versus Sampling Zeros

Mehryar Mohri
Courant Institute - NYU
719 Broadway
New York, NY 10003
mohri@cs.nyu.edu

Brian Roark
CSLU - OGI
20000 NW Walker Road
Beaverton, Oregon 97006
roark@cslu.ogi.edu

Abstract

Probabilistic sequence models estimated from large corpora typically require smoothing techniques to reserve some probability mass for unobserved events. These techniques fail to distinguish between events unobserved due to sampling limitations, *sampling zeros*, and those unobserved due to structural reasons such as syntactic constraints, *structural zeros*. We investigate the use of statistical tests to determine structural zeros, to avoid assigning them probability mass and thereby improve model accuracy. Experimental results on a context-free parsing task demonstrate the usefulness of these techniques.

1 Motivation

The design of accurate probabilistic models for sequences is a key problem in a variety of applications in computational biology and natural language processing. Probabilistic models for letters, phonemes, words, or word classes (e.g. part-of-speech tags) are crucial components of information extraction, speech recognition and synthesis, or handwriting recognition systems [7, 9, 4, 11]. Similar models for DNA or protein sequences are also of considerable importance in bioinformatics [6, 12].

Probabilistic models for sequences are typically derived from large datasets. In natural language processing applications, the corpora used often contain tens or hundreds of million of tokens. Even so, one of the key problems faced in the design of these models is that of data sparsity: some sequences may not appear, even in very large samples. Numerous *smoothing* techniques have been introduced to deal with this sparsity problem by reserving some probability mass for unobserved sequences (see, e.g., [9]). A feature common to all of these techniques is that they do not differentiate between sequences that were unobserved due to the limited size of the sample, which we refer to as *sampling zeros*, from sequences that were unobserved due to their being grammatically forbidden or otherwise illicit, which we call *structural zeros*. Smoothing techniques reserve some probability mass both for sampling and structural zeros.

If some or all structural zeros were known in advance, they could be excluded from the model, or equivalently assigned no probability mass. The probability mass thereby freed up could be non-negligible. It could be redistributed among possible sequences in a way that could improve the quality of the overall model. For any given sequence modeling task, we cannot expect to have existing lists of such ill-formed sequences. Instead, we propose to use a large corpus to infer that some sequences are structurally impossible using statistical criteria and to use that information to improve the model derived from that corpus.

Note that our detection of structural zeros and the changes it implies to the design of a statistical model are not related to the so-called *zero-inflated models* [10]. The purpose of zero-inflated models is to account for excess zeros (zero counts greater than expected), typically by increasing a model's probability of zero, regardless of their being structural or sampling zeros.

To illustrate the benefit of our approach, we investigate modeling sequences of grammatical categories on the right-hand side (RHS) of rules in a probabilistic context-free grammar (PCFG). PCFGs induced from commonly used treebanks, such as the Penn Wall St. Journal (WSJ) Treebank, contain many productions with a lengthy sequence of categories on the RHS, causing both a sparse-data problem (many possible productions are unobserved) and a processing efficiency problem. Certain grammar factorizations address both issues, but allocate significant probability mass to ungrammatical structures, leading to large reductions in parsing accuracy. We show that detecting and removing structural zeros from a factored PCFG provides the benefits of the factorization with greatly improved parsing accuracy.

The paper is organized as follows. We first discuss several statistical criteria for detecting structural zeros and compare them by applying them to a large corpus. Section 3 describes the application of our method to context-free grammar factorization for chart parsing. Section 4 reports in detail the results of our experiments with a probabilistic context-free parser trained on a large corpus of about one million words.

2 Statistical Criteria

The main idea behind our method for detecting structural zeros is to search for events that are very frequent but that do not co-occur. For example, to create a statistical language model for English, i.e. a probabilistic model estimating the probability of any sequence of English words, we may wish to rule out some ungrammatical sequences. We can use a large corpus to count the number of occurrences of any sequence of words. If the counts of two sequences x and y are very large but the count of their co-occurrence is zero, then the co-occurrence of x and y can be viewed as a candidate for the list of events that are structurally inadmissible.

In the simplest case, we can choose x and y to be single words and count how often x is followed by y. For example, the words "*of*" and "*the*" typically occur extremely frequently in a large sample of English text, but if the count of "*the of*" is zero, then we can view "*the of*" as a candidate structural zero. In general, we may wish to relax this condition by allowing the count of sequence xy to be non-zero and only require it to be very low compared to the counts of x and y. This is because corpora may contain noisy data causing illicit sequences to infrequently appear. But, to simplify matters, in what follows we will consider strictly unobserved sequences

We can similarly view a trigram xyz as a structural zero, when it does not appear, or very seldom appears in the data while the bigrams xy and yz appear very frequently, e.g. "*brand new york*". More generally, we can distinguish an n-gram sequence as a structural zero if the counts of its subsequences are relatively high.

While the examples just discussed were with sequences of words, the same approach applies to other sequence models. For example, a context-free grammar induced from the Penn Treebank may have a rule of the form NP→DT JJ JJ NN NN NNS to handle such noun phrases as "*the hot tasty duck beak soups*." In a treebank, a similar rule, such as NP→DT RB JJ JJ NN NN NNS, which would handle such noun phrases as "*the very hot tasty duck beak soups*," might be unobserved. Smoothing techniques can be applied to reserve some probability mass for such unobserved rules. But, these techniques would then similarly assign some probability mass to many truly ungrammatical rules as well. In this case, we can look at the co-occurrence of categories on the right-hand side of productions with a particular left-hand side, to infer which combinations are illicit.

Different statistical criteria can be used to compare the counts of two events with that of their co-occurrence. This section briefly introduces several criteria and compares them by applying them to the same corpus.

2.1 Notation

This section describes several statistical criteria to determine if a sequence of two words or categories should be viewed as a structural zero. These tests can be generalized to longer and more complex sequences, and to different types of events.

Given a corpus $\mathcal{C}$, and a vocabulary Σ, we denote by c_x the number of occurrences of x in $\mathcal{C}$. Let n be the total number of observations in $\mathcal{C}$. We will denote by $\bar{x}$ the set $\{y \in \Sigma : y \neq x\}$. Hence $\mathrm{c}_{\bar{x}} = n - \mathrm{c}_x$. Let $\mathrm{p}(x) = \frac{\mathrm{c}_x}{n}$, and for $a \in \Sigma$, let $\mathrm{p}(a|x) = \frac{\mathrm{c}_{xa}}{\mathrm{c}_x}$. Note that $\mathrm{c}_{\bar{x}a} = \mathrm{c}_a - \mathrm{c}_{xa}$.

2.2 Mutual information

The mutual information between two random variables X and Y is defined as

$$I(X;Y) = \sum_{x,y} \mathrm{p}(x,y) \log \frac{\mathrm{p}(x,y)}{\mathrm{p}(x)\mathrm{p}(y)}. \tag{1}$$

For a particular word sequence of length two ab, this suggests the following statistic:

$$\begin{aligned} I(ab) &= \log \mathrm{p}(ab) - \log \mathrm{p}(a) - \log \mathrm{p}(b) \\ &= \log \mathrm{c}_{ab} - \log \mathrm{c}_a - \log \mathrm{c}_b + \log n \end{aligned} \tag{2}$$

Unfortunately, for $\mathrm{c}_{ab} = 0$, $I(ab)$ is not finite. If we assume, however, that all unobserved sequences are given some ϵ count, then

$$I(ab) \quad = \quad K - \log \mathrm{c}_a - \log \mathrm{c}_b, \tag{3}$$

where K is a constant. We need these statistics only for ranking purposes, thus we can ignore the constant factor.

2.3 Log odds ratio

Another statistic that, as with mutual information, is ill-defined with zeros, is the *log odds ratio*:

$$\log(\hat{\theta}) \quad = \quad \log \mathrm{c}_{ab} + \log \mathrm{c}_{\bar{a}\bar{b}} - \log \mathrm{c}_{\bar{a}b} - \log \mathrm{c}_{a\bar{b}}. \tag{4}$$

Here again, if $\mathrm{c}_{ab} = 0$, $\log(\hat{\theta})$ is not finite. But, if we give all unobserved bigrams a small count ϵ, the expression becomes

$$\log(\hat{\theta}) \quad = \quad K + \log \mathrm{c}_{\bar{a}\bar{b}} - \log \mathrm{c}_b - \log \mathrm{c}_a. \tag{5}$$

2.4 Pearson chi-squared

For any $i, j \in \Sigma$, define $\hat{\mu}_{ij} = \frac{\mathrm{c}_i \mathrm{c}_j}{n}$. The Pearson chi-squared test of independence is then defined as follows:

$$\begin{aligned} \mathcal{X}^2 &= \frac{(\mathrm{c}_{ab} - \hat{\mu}_{ab})^2}{\hat{\mu}_{ab}} + \frac{(\mathrm{c}_{\bar{a}b} - \hat{\mu}_{\bar{a}b})^2}{\hat{\mu}_{\bar{a}b}} + \frac{(\mathrm{c}_{a\bar{b}} - \hat{\mu}_{a\bar{b}})^2}{\hat{\mu}_{a\bar{b}}} + \frac{(\mathrm{c}_{\bar{a}\bar{b}} - \hat{\mu}_{\bar{a}\bar{b}})^2}{\hat{\mu}_{\bar{a}\bar{b}}} \\ &= \frac{(n\mathrm{c}_{ab} - \mathrm{c}_a \mathrm{c}_b)^2}{n\mathrm{c}_a \mathrm{c}_b} + \frac{(n\mathrm{c}_{\bar{a}b} - \mathrm{c}_{\bar{a}} \mathrm{c}_b)^2}{n\mathrm{c}_{\bar{a}} \mathrm{c}_b} + \frac{(n\mathrm{c}_{a\bar{b}} - \mathrm{c}_a \mathrm{c}_{\bar{b}})^2}{n\mathrm{c}_a \mathrm{c}_{\bar{b}}} + \frac{(n\mathrm{c}_{\bar{a}\bar{b}} - \mathrm{c}_{\bar{a}} \mathrm{c}_{\bar{b}})^2}{n\mathrm{c}_{\bar{a}} \mathrm{c}_{\bar{b}}}. \end{aligned}$$

In the case of interest for us, $\mathrm{c}_{ab} = 0$ and the statistic simplifies as follows:

$$\begin{aligned} \mathcal{X}^2 &= \frac{\mathrm{c}_a \mathrm{c}_b}{n} + \frac{(n\mathrm{c}_b - \mathrm{c}_{\bar{a}} \mathrm{c}_b)^2}{n\mathrm{c}_{\bar{a}} \mathrm{c}_b} + \frac{(n\mathrm{c}_a - \mathrm{c}_a \mathrm{c}_{\bar{b}})^2}{n\mathrm{c}_a \mathrm{c}_{\bar{b}}} + \frac{(n(n - \mathrm{c}_a - \mathrm{c}_b) - \mathrm{c}_{\bar{a}} \mathrm{c}_{\bar{b}})^2}{n\mathrm{c}_{\bar{a}} \mathrm{c}_{\bar{b}}} \\ &= \frac{\mathrm{c}_a \mathrm{c}_b}{n} + \frac{\mathrm{c}_b \mathrm{c}_a^2}{n\mathrm{c}_{\bar{a}}} + \frac{\mathrm{c}_a \mathrm{c}_b^2}{n\mathrm{c}_{\bar{b}}} + \frac{\mathrm{c}_a^2 \mathrm{c}_b^2}{n\mathrm{c}_{\bar{a}} \mathrm{c}_{\bar{b}}} \quad = \quad \frac{n\mathrm{c}_a \mathrm{c}_b}{\mathrm{c}_{\bar{a}} \mathrm{c}_{\bar{b}}}. \end{aligned} \tag{6}$$

2.5 Log likelihood ratio

Pearson's chi-squared statistic assumes a normal or approximately normal distribution, but that assumption typically does not hold for the occurrences of rare events [5]. It is then preferable to use the likelihood ratio statistic which allows us to compare the null hypothesis, that $\mathrm{p}(b) = \mathrm{p}(b|a) = \mathrm{p}(b|\bar{a}) = \frac{\mathrm{c}_b}{n}$, with the hypothesis that $\mathrm{p}(b|a) = \frac{\mathrm{c}_{ab}}{\mathrm{c}_a}$ and $\mathrm{p}(b|\bar{a}) = \frac{\mathrm{c}_{\bar{a}b}}{\mathrm{c}_{\bar{a}}}$. These discrete conditional probabilities are a binomial distribution, hence the likelihood ratio is

$$\begin{aligned} \lambda &= \frac{\mathrm{p}(b)^{\mathrm{c}_{ab}}(1 - \mathrm{p}(b))^{\mathrm{c}_a - \mathrm{c}_{ab}} \binom{\mathrm{c}_a}{\mathrm{c}_{ab}} \mathrm{p}(b)^{\mathrm{c}_{\bar{a}b}}(1 - \mathrm{p}(b))^{\mathrm{c}_{\bar{a}} - \mathrm{c}_{\bar{a}b}} \binom{\mathrm{c}_{\bar{a}}}{\mathrm{c}_{\bar{a}b}}}{\mathrm{p}(b|a)^{\mathrm{c}_{ab}}(1 - \mathrm{p}(b|a))^{\mathrm{c}_a - \mathrm{c}_{ab}} \binom{\mathrm{c}_a}{\mathrm{c}_{ab}} \mathrm{p}(b|\bar{a})^{\mathrm{c}_{\bar{a}b}}(1 - \mathrm{p}(b|\bar{a}))^{\mathrm{c}_{\bar{a}} - \mathrm{c}_{\bar{a}b}} \binom{\mathrm{c}_{\bar{a}}}{\mathrm{c}_{\bar{a}b}}} \\ &= \frac{\mathrm{p}(b)^{\mathrm{c}_{ab}}(1 - \mathrm{p}(b))^{\mathrm{c}_a - \mathrm{c}_{ab}} \mathrm{p}(b)^{\mathrm{c}_{\bar{a}b}}(1 - \mathrm{p}(b))^{\mathrm{c}_{\bar{a}} - \mathrm{c}_{\bar{a}b}}}{\mathrm{p}(b|a)^{\mathrm{c}_{ab}}(1 - \mathrm{p}(b|a))^{\mathrm{c}_a - \mathrm{c}_{ab}} \mathrm{p}(b|\bar{a})^{\mathrm{c}_{\bar{a}b}}(1 - \mathrm{p}(b|\bar{a}))^{\mathrm{c}_{\bar{a}} - \mathrm{c}_{\bar{a}b}}} \end{aligned} \tag{7}$$

In the special case where $\mathrm{c}_{ab} = 0$, $\mathrm{p}(b|\bar{a}) = \mathrm{p}(b)$, and this expression can be simplified as follows

$$\begin{aligned} \lambda &= \frac{(1 - \mathrm{p}(b))^{\mathrm{c}_a} \mathrm{p}(b)^{\mathrm{c}_{\bar{a}b}}(1 - \mathrm{p}(b))^{\mathrm{c}_{\bar{a}} - \mathrm{c}_{\bar{a}b}}}{\mathrm{p}(b|\bar{a})^{\mathrm{c}_{\bar{a}b}}(1 - \mathrm{p}(b|\bar{a}))^{\mathrm{c}_{\bar{a}} - \mathrm{c}_{\bar{a}b}}} \\ &= (1 - \mathrm{p}(b))^{\mathrm{c}_a}. \end{aligned} \tag{8}$$

The log likelihood ratio, denoted G^2, is known to be asymptotically $\mathcal{X}^2$ distributed. In this case

$$G^2 \quad = \quad -2\mathrm{c}_a \log(1 - \mathrm{p}(b)) \tag{9}$$

and with the binomial, this has 1 degree of freedom, hence the distribution will asymptotically have a mean of 1 and a standard deviation of $\sqrt{2}$.

	Top 50 list			Top 200 list			Top 500 list		
Statistic	$\mathcal{X}^2$	I	$\log(\hat{\theta})$	$\mathcal{X}^2$	I	$\log(\hat{\theta})$	$\mathcal{X}^2$	I	$\log(\hat{\theta})$
G^2	3	1	3	9	4	9	28	13	28
$\mathcal{X}^2$	-	3	0	-	8	0	-	25	0
I		-	3		-	8		-	25

Table 1: Maximum difference in rank between different statistics for top 50, 200 and 500 scoring unseen bigrams in 4M word Switchboard corpus.

2.6 Ranking differences

Although all of these statistics are measuring nearly the same thing, i.e., the frequency of the individual events, each statistic is slightly different. To get a sense of how these differences affect the ranking, we generated the lists of the most surprising zero bigrams according to each statistic, and compared their rank order. For each list of k bigrams, we look for the largest difference in rank between two statistics. Table 1 shows the differences found, when the bigram statistics are gathered from a 4 million-word Switchboard corpus. The log odds ratio and Pearson chi-squared statistics give identical rankings. The log likelihood ratio and mutual information statistics are closer together, but overall the lists given by all of the statistics are quite similar. In view of this similarity, we chose to use in our experiments the log likelihood ratio test.

3 Application to Statistical Parsing

We chose to illustrate these techniques within the context of probabilistic grammar estimation because smoothing techniques are widely used in this domain, but also because adjacent categories on the right-hand side (RHS) of a rule are by definition strongly constrained by grammaticality. With a relatively small non-terminal vocabulary (about 100), this makes it for a nice test-bed for detection and use of structural zeros.

3.1 Definitions

A context-free grammar (CFG) $G = (V, T, S^\dagger, P)$ consists of a set of non-terminal symbols V, a set of terminal symbols T, a start symbol $S^\dagger \in V$, a set of rule productions P of the form: $A \longrightarrow \alpha$, where $A \in V$ and $\alpha \in (V \cup T)^*$. A PCFG is a CFG with a probability assigned to each rule, such that the probabilities of all rules expanding a given non-terminal sum to one; specifically, each RHS has a probability given the left-hand side of the rule. For all of the trials reported here, we trained a PCFG on sections 2-21 of the Penn WSJ Treebank (40k sentences, 936k words), and evaluated on section 24 (1346 sentences, 32K words). True part-of-speech tags are taken as terminals, and words are ignored.

3.2 Grammar smoothing and factorization

PCFGs induced from the Penn Treebank have many productions with very long sequences of non-terminals on the RHS. Probability estimates of the RHS given the left-hand side are often smoothed by making a Markov assumption regarding the conditional independence of a category on those more than k categories away [4, 2].

$$\begin{aligned} \mathrm{p}(X \rightarrow Y_1 \ldots Y_n) &= \mathrm{p}(Y_1|X) \prod_{i=2}^{n} \mathrm{p}(Y_i|X, Y_1 \ldots Y_{i-1}) \\ &\approx \mathrm{p}(Y_1|X) \prod_{i=2}^{n} \mathrm{p}(Y_i|X, Y_{i-k} \ldots Y_{i-1}) \qquad (10) \end{aligned}$$

This Markov assumption provides probability mass to unobserved productions, whether those productions are sampling or structural zeros.

Making a Markov assumption on productions is closely related to grammar transformations required for certain efficient parsing algorithms. For example, the CYK parsing algorithm [8, 13] takes as input a binarized PCFG, i.e. a grammar with only binary productions[1]. PCFGs are induced from a treebank, which has been

[1] Our implementation has been extended to allow for unary productions in the PCFG.

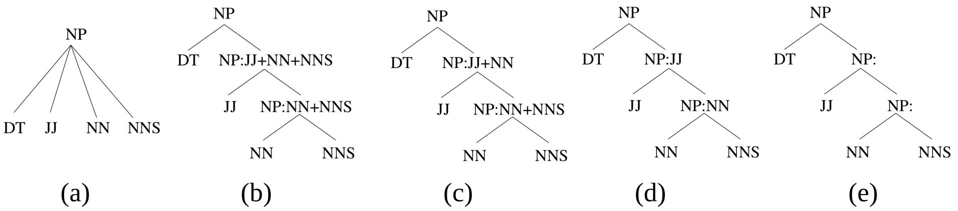

Figure 1: Five representations of an n-ary production, $n = 4$. (a) Original production (b) Right-factored production (c) Right-factored Markov-2 (d) Right-factored Markov-1 (e) Right-factored Markov-0

PCFG	Time (s)	Words/s	|NT|	LR	LP	F-measure
Right-factored	4171	7.8	10105	69.4	73.9	71.6
Right-factored, Markov-2	1145	28.3	2492	69.2	74.0	71.5
Right-factored, Markov-1	281	115.1	564	68.3	73.3	70.7
Right-factored, Markov-0	120	269.6	99	61.4	65.7	63.5

Table 2: Baseline results of CYK parser using different probabilistic context-free grammars. Grammars are trained from sections 2-21 of the Penn WSJ Treebank and tested on section 24, given the true POS-tags. The second and third columns report the total parsing time in second and the number of words parsed per second. The number of non-terminals, |NT|, is indicated in the next column. The last three columns show the precision, recall and F-measure.

factored so that n-ary productions with $n > 2$ become sequences of $n - 1$ binary productions. Full right-factorization involves creating composite non-terminals which group together the final $n-1$ categories from the RHS of an n-ary production. For example, the original production NP → DT JJ NN NNS shown in figure 1(a) is factored into three binary rules, as shown in figure 1(b). Note that a PCFG induced from such right-factored trees is weakly equivalent to a PCFG induced from the original treebank, i.e. it describes the same language.

From such a factorization, one can make a Markov assumption for estimating the production probabilities by simply recording only the labels of the first k children dominated by the composite factored label. Figure 1 (c), (d), and (e) show right-factored trees of Markov orders 2, 1 and 0 respectively. In addition to a smoothing benefit as mentioned above, these factorizations reduce the size of the non-terminal set, which in turn improves CYK efficiency. The efficiency benefit of making a Markov assumption in factorization can be substantial, since the CYK algorithm has complexity O($n^3|V_0|(|V_0|^2 + |V_f|)$), where n is the length of the string, $|V_0|$ is the size of the original, non-factored non-terminal set, and $|V_f|$ is the size of the set of new, factored non-terminals[2]. With standard right-factorization, as in figure 1 (b), the non-terminal set for the PCFG induced from sections 2-21 of the Penn WSJ Treebank grows from its original size of 72 to 10105. With a Markov factorization of orders 2, 1 and 0 we get non-terminal sets of size 2492, 564, and 99, respectively.

These reductions in the size of the non-terminal set from the original factored grammar results in an order of magnitude reduction in complexity of the CYK algorithm. One common strategy in statistical parsing is to first build a chart with a simple PCFG, which is then pruned prior to evaluating parses with richer, higher complexity models [4, 2]. As a result, producing such a chart as efficiently as possible is very important [3, 1], making these factorizations particularly useful.

Table 2 shows baseline results for standard right-factorization and factorization with Markov orders 0-2. Training consists of applying a particular grammar factorization to the treebank prior to inducing a PCFG using maximum likelihood (relative frequency) estimation. Testing consists of CYK parsing of the evaluation set with the induced grammar, then de-transforming the maximum likelihood parse back to the original format for evaluation against the reference parse. Evaluation includes the standard PARSEVAL measures labeled precision (LP) and labeled recall (LR), plus the harmonic mean (F-measure) of these two scores.

From these results, we can see the large efficiency benefit of the Markov assumption, as the size of the

[2] Every binary production in the factored grammar must have one of the original non-terminals as the first child, hence there are $|V_0|$ possibilities for first child and $|V| = |V_0| + |V_f|$ possibilities for second child. If the second child is in V_0, then there are $2|V_0|$ possible parents. If the second child is in $|V_f|$, then there are just 2 possible parents, since the factored category encodes parent information.

Unobs. productions	G^2 score	Req. NTs	Unobs. productions	G^2 score	Req. NTs
S $\rightarrow\alpha$ VP VP γ	24938.2	S:VP	S $\rightarrow\alpha$ PP VP . γ	2549.9	S:VP+.
S $\rightarrow\alpha$ VP VP . γ	24575.9	S:VP+.	NP $\rightarrow\alpha$ DT CC NP γ	2430.5	NP:CC+NP
S $\rightarrow\alpha$ VP NP VP γ	9096.3	S:NP+VP	VP $\rightarrow\alpha$ TO NP γ	2424.6	VP:NP
S $\rightarrow\alpha$ VP , NP γ	7095.0	S:,+NP	NP $\rightarrow\alpha$ NNP , NP γ	2410.5	NP:,+NP
S $\rightarrow\alpha$, VP γ	6582.3	S:VP	S $\rightarrow\alpha$ VP ADVP γ	2331.7	S:ADVP
S $\rightarrow\alpha$, VP . γ	6486.7	S:VP+.	S $\rightarrow\alpha$ VP S γ	2324.9	S:S
NP $\rightarrow\alpha$ DT , γ	6136.9	NP:,	S $\rightarrow\alpha$ NP CC γ	2105.7	S:CC
QP $\rightarrow\alpha$ CD CD CD γ	5358.6	QP:CD+CD	NP $\rightarrow\alpha$ PRP NNP γ	2072.2	NP:NNP
NP $\rightarrow\alpha$ DT , NP γ	2973.8	NP:,+NP	NP $\rightarrow\alpha$ NNP NP γ	2048.9	NP:NP
S $\rightarrow\alpha$, , γ	2783.3	S:,	S $\rightarrow\alpha$ NP CC S γ	2024.9	S:CC+S

Table 3: Top 20 ranked unobserved production templates, using the log likelihood ratio statistic, along with the factored non-terminal required to give them zero probability.

PCFG	Time (s)	Words/s	|NT|	LR	LP	F-measure
Right-factored, Markov-0	120	269.6	99	61.4	65.7	63.5
RF, Markov-0, top 100 zeros	157	206.1	152	68.0	72.5	70.2
RF, Markov-0, top 200 zeros	173	187.0	184	68.6	73.4	70.9
RF, Markov-0, top 500 zeros	234	138.3	286	69.1	73.8	71.4
RF, Markov-0, top 1000 zeros	272	118.9	386	69.2	73.9	71.5
RF, Markov-0, top 2000 zeros	370	87.4	596	69.2	74.1	71.6

Table 4: Trials adding in categories to rule out unobserved production templates.

non-terminal set shrinks. However, the efficiency gains come at a cost, with the Markov order-0 factored grammar resulting in a loss of a full 8 percentage points of F-measure accuracy. Ideally, one would like to get the benefit of the small non-terminal set, while enforcing key grammatical constraints. We will do this by using a statistical test to find structural zeros and change the factorization to remove probability mass from them.

4 Experiments – Structural Zeros

We used the log likelihood ratio statistic G^2 to rank unobserved events ab, where $a \in (V \cup T)$ and $b \in (V \cup T)^+$ are a sequence of children in the same production $A \rightarrow \alpha\, a\, b\, \gamma$, where $\alpha, \gamma \in (V \cup T)^*$ and $b\gamma \in (V \cup T)^k$ for $k > 1$.

This corresponds to a situation where a sequence of children ab with parent A are never observed. For use in equation 9,

$$\begin{array}{ll} \mathbf{c}_a = \sum_{\alpha,b',\gamma} c(A \rightarrow \alpha\, a\, b'\, \gamma) & \mathbf{c}_b = \sum_{\alpha,a',\gamma} c(A \rightarrow \alpha\, a'\, b\, \gamma) \\ \mathbf{c}_{ab} = \sum_{\alpha,\gamma} c(A \rightarrow \alpha\, a\, b\, \gamma) & \mathrm{p}(b) = \frac{\mathbf{c}_b}{\sum_{b'} \mathbf{c}_{b'}}. \end{array} \tag{11}$$

Thus, a and b may represent, for example, two events such as DT being the ith child of an NP production, and JJ being the (i+1)th child of an NP production.

In the original or fully factored PCFGs, if $\mathbf{c}_{ab} = 0$ then all productions that fit such a production template would have probability zero using maximum likelihood estimation. With a Markov order-0 factored PCFG, however, they would be given probability mass. To remove that probability mass, we can change the Markov order-0 factorization to create the non-terminal $A\!:\!b$ when factoring a sequence of children under A beginning with b. The resulting grammar would provide zero probability mass to a non-terminal a followed by b under category A.

We looked for $b \in (V \cup T)^k$ for $k \in \{1, 2\}$ to find child sequences of length 2 and 3 that do not occur. Table 3 shows the 20 highest ranked zero occurrence rule templates given the G^2 statistic. The top ranked unobserved rule template is a sequence of two VP children in an S production. In order to provide zero probability to such productions, when factoring an S production, the S:VP factored category (or one even more specific, if needed for another zero, such as the second in the list) must be used when the first child in the factored sequence is a VP.

Table 4 shows the results of using an order-0 Markov factorization, with factored categories required to remove probability mass from the the top n ranked unobserved production templates. Using the top 100 increases the size of the non-terminal set by just 53, but improves the F-measure accuracy by 6.7 percent.

5 Conclusion

We presented simple techniques for detecting structural zeros using large natural language corpora and demonstrated their effectiveness for improving the accuracy of smoothed, factored PCFGs. The methods outlined can be used to improve model accuracy in other domains. For example, the detection of structural zeros in images could help improve the accuracy of image modeling by adding data-derived model-constraints to the recognizer. Using more complex features than co-occurrences of neighboring events may lead to more accurate techniques for detecting structural zeros. In addition to their use for model enhancement, accurate techniques for detecting structural zeros could help better understand human learning. The problem of accurate detection of structural zeros may also arise novel questions in statistical learning theory.

References

[1] Don Blaheta and Eugene Charniak. Automatic compensation for parser figure-of-merit flaws. In *Proceedings of the37th Annual Meeting of the Association for Computational Linguistics*, pages 513–518, 1999.

[2] Eugene Charniak. A maximum-entropy-inspired parser. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics*, pages 132–139, 2000.

[3] Eugene Charniak, Sharon Goldwater, and Mark Johnson. Edge-based best-first chart parsing. In *Proceedings of the Sixth Workshop on Very Large Corpora*, pages 127–133, 1998.

[4] Michael Collins. Three Generative, Lexicalised Models for Statistical Parsing. In *35th Meeting of the Association for Computational Linguistics (ACL ’96), Proceedings of the Conference, Santa Cruz, California*, 1997.

[5] Ted Dunning. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 19(1):61–74, 1994.

[6] R. Durbin, S.R. Eddy, A. Krogh, and G.J. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge UK, 1998.

[7] Frederick Jelinek. *Statistical Methods for Speech Recognition*. The MIT Press, Cambridge, Massachusetts, 1998.

[8] T. Kasami. An efficient recognition and syntax analysis algorithm for context-free languages. Technical Report, AFCRL-65-758, Air Force Cambridge Research Lab., Bedford, MA, 1965.

[9] S. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35:400–401, 1987.

[10] M.S. Ridout, C.G.B. Demetrio, and J.P. Hinde. Models for counts data with many zeros. In *Proceedings of the XIXth International Biometric Conference, Cape Town*, pages 179–192, 1998.

[11] Brian Roark. Probabilistic top-down parsing and language modeling. *Computational Linguistics*, 27(2):249–276, 2001.

[12] Michael S. Waterman. *Introduction to Computational Biology*. Chapman and Hall, New York, 1995.

[13] David H. Younger. Recognition and parsing of context-free languages in time n^3. *Information and Control*, 10(2):189–208, 1967.