Training of Probabilistic Context-Free Grammar

ZHOU GuoDong LUA KimTeng
Dept. of Computer Science
School of Computing
National University of Singapore
Singapore 119260
Email: {zhougd, luakt}@comp.nus.edu.sg

Abstract

Probabilistic context-free grammar(PCFG) has been successfully used in natural language processing. However, there exists a seroius problem in PCFG training. In this paper, a new mixed PCFG training approach is proposed and compared with the commonly used supervised and unsupervised training approaches. Compared with supervised training, it does not need to construct a very large corpus of disambiguated parse trees. Additionally, it can guide the probability estimates away from not only bad initial values but also bad values in each of subsequent iterations. Different PCFGs trained by different training approaches are evaluated in the applications of Part-of Speech(POS) tagging and sentence parsing. It is found that the mixed training approach has much better performance than the supervised or unsupervised approach alone.

Introduction

Language modeling is crucial to natural language processing. While statistical language modeling approaches such as N-gram modeling are effective and have been successfully used in the applications such as speech recognition, part-of-speech tagging and others, it has been proven to be difficult to approximate language phenomena precisely enough when context dependencies exist over a structure.

An appealing alternative is the grammar-based language modeling approach. Grammar has long been used in linguistics and natural language processing to represent languages. Such models intuitively capture properties of natural language that N-gram models cannot. It has been shown that grammatical language models can express the structure dependency directly by means of syntax [Lari+90][Resnik92][Schabes+93]. Not only can such models capture the same short-distance dependency as N-gram models, but they also have the potential to model the structure dependency over a long distance beyond the scope of N-gram models. Therefore, they have the potential for superior performance. Furthermore, grammatical models have the potential to be more compact while achieving equivalent performance as N-gram models[Brown+92], because grammars can express the classing or grouping of similar words. However, constructing a grammar to parse sentences is a difficult task. One of the most serious problems is the large number of ambiguities. Pure syntactic analysis based on only syntactic knowledge will sometimes result in many ambiguous parses, even for a written sentence.

A reasonable way to do this job is to employ probability as a device to quantify language ambiguities, in other words, to combine linguistic expertise (e.g. grammar knowledge) with its probabilistic augmentation for approximating natural language. Within this framework, semantic and pragmatic constraints are expected to be captured implicitly in the probabilistic augmentation while syntactic constraints are captured explicitly in some sense. An good example of this framework is the Probabilistic Context-Free Grammar (PCFG).

There have been several works on parsing with PCFGs in recent years. [Fujisaki+91] described an experiment in which a PCFG was used as a disambiguation method. The grammar contained 7550 rules in Chomsky Normal Form (CNF) and the rule probabilities were determined using the iterative Inside-Outside algorithm on a corpus of 4206 sentences. [Corazza+91] discussed how to compute the probabilities of sub-strings of sentences derived from a PCFG. They had also described how such probabilities can be used in conjunction with an island-driven probability parser to score alternative acoustic hypotheses produced by a speech recognition system. [Schabes+93] introduced stochastic lexicalized CFGs(SLCFGs), a context-free version of stochastic lexicalized tree-adjoining grammars, and presented algorithms for parsing, training of the probabilities and recovering the most probable parse of a given input for these kinds of grammars. The main advantage of using SLCFGs is their lexical sensitivity, which provides them with a better basis for capturing distributed information about words.

The main problem of PCFG is training. In this paper, we will study different training approaches and their characteristics. Meanwhile, a new mixed PCFG training approach is proposed and compared with existing training approaches.

The organization of the rest is as follows: the concept of PCFG is described in Section 1. Section 2 discusses different approaches for PCFG training while the experiments in the applications of POS tagging and sentence parsing are done in Section 3. Finally, a conclusion of this paper is given.

1 Probabilistic Context-Free Grammar

A probabilistic context-free grammar (PCFG) consists of

a) a set of non-terminal symbols N
b) a set of terminal symbols V
c) a start non-terminal symbol $S \in \mathrm{N}$, from which the grammar generates the sentences
d) a set of rules $\Re$
e) a set of rule probabilities $\{P(r)$ for all $r \in \Re\}$

The rules (or productions) are of the form $X \to \lambda$, where $X \in \mathrm{N}$ and $\lambda \in (\mathrm{N} \bigcup V)^*$. X is called the left-hand side (LHS) of the rule, whereas λ is called the right-hand side (RHS) of the rule.

The rule probability $P(X \to \lambda)$ denotes the probability that a non-terminal X, having appeared during the top-down sentence derivation process, will be replaced with the RHS λ. Obviously,

$$\sum_{\lambda} P(X \to \lambda) = 1 \qquad (1)$$

holds.

A PCFG is thus exactly like a standard CFG, except that rules are assigned with probability parameters. If this process is done in a consistent way, it induces a probability on the sentences of the language defined by the grammar. More importantly, it induces a probability distribution on the various derivation trees of a particular grammatical sentence, which enables us to quantify sentence ambiguities.

The probability of a derivation tree t can be computed as the product of the probabilities of the rules which are employed for deriving t.

$$P(t) = \prod_{r \in T(t)} P(r) \qquad (2)$$

Here r denotes a rule, and $T(t)$ denotes the ordered set of rules which are employed for deriving the tree t. Figure 1 illustrates how the probability of a derivation tree can be computed as a product of rule probabilities.

An ambiguous grammar allows many different derivation trees for a given sentence. From the viewpoint of sentence parsing, we say that a sentence is ambiguous when more than one tree, say $t_1, t_2, \ldots$, can be derived from the parsing process. By having a device to compute derivation tree probabilities as shown by Equation 2, we can handle sentence ambiguity in a quantitative way. Namely, when a sentence s is parsed ambiguously to derive trees $t_1, t_2, \ldots$, and a probability $P(t_i)$ is computed for each derivation tree t_i, the sum of the probabilities $\sum_i P(t_i)$ can be regarded as the probability that a particular sentence s will be

generated among many other possibilities. More interesting is the ratio denoting relative probabilities among ambiguous derivation trees: $\frac{P(t_i)}{\sum_i P(t_i)}$.

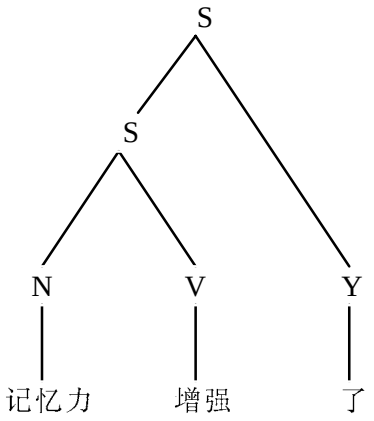
It is obvious that above relative probability also gives measure of likelihood for each derivation tree t_i.

Notation: In this paper, capital letters $A, B, C, \ldots$ denote non-terminal symbols and lowercase letters $a, b, c, \ldots$ denote terminal syllables.

normalized into rule probabilities by Equation 1. This method produces accurate probability estimates when trained on a sufficiently large corpus of disambiguated parse trees. The major problem with this supervised training method is that the process of constructing a sufficiently large corpus of disambiguated parse trees can be labor intensive.

2.2 Unsupervised Training

As an alternative to the supervised training method, an efficient unsupervised version of the Baum-Welch algorithm has been developed



(a) A derivation tree t

$$P(t) = P(S \rightarrow SY) \times P(S \rightarrow NV) \times P(N \rightarrow 记忆力) \times P(V \rightarrow 增强) \times P(Y \rightarrow 了) \times$$

S: the start non-terminal symbol
N: noun
V: verb

(b) The probability of the left derivation tree

Figure 6.1: A derivation tree of Chinese sentence "记忆力增强了" and its probability

2 PCFG Training

Given a CFG, there are two commonly used approaches for estimating the rule probabilities: supervised training and unsupervised training. In this paper, a third approach - mixed training- will be proposed and compared with the other two approaches.

2.1 Supervised Training

Supervised training uses a corpus of disambiguated parse trees to estimate the rule probabilities of a PCFG. The occurrence frequencies of the rules in the correct (disambiguated) parse trees can be determined from the corpus. These frequencies can then be

for PCFGs[Jelinek+90][Lari+90]. This iterative training algorithm can be divided into six steps as follows[Fujisaki+91]:

1. Make an uniform initial guess of rule probabilities $\{P_{old}(X \rightarrow \lambda)\}$ such that $\sum_{\lambda} P_{old}(X \rightarrow \lambda) = 1$ holds
2. Parse each sentence and all possible parse trees are produced. We denote as T_j^i the j-th parse tree for the i-th sentence S^i.
3. Compute the probability of each parse tree T_j^i using the rule probabilities from the previous iteration in the following way:

$$P(T_j^i) = \prod_{r \in T_j^i} P_{old}(r) \qquad (3)$$

This computes $P(T_j^i)$ as a product of the probabilities of the rules which are employed to generate the parse tree T_j^i.

4. The occurrence frequency $C^i(X \to \lambda)$ of the rule $X \to \lambda$ in the sentence S^i is estimated by counting its occurrence in all parse trees, each weighted by the probability of the parse tree.

$$C^i(X \to \lambda) = \sum_j \left(\frac{P(T_j^i)}{\sum_k P(T_k^i)} \times n_j^i(X \to \lambda)\right) \qquad (4)$$

where $n_j^i(X \to \lambda)$ denotes the number of times the rule $X \to \lambda$ is used in the parse tree T_j^i.

5. The estimated frequencies are then normalized into a new set of rule probabilities.

$$P_{new}(X \to \lambda) = \sum_i \frac{C^i(X \to \lambda)}{\sum_\beta C^i(X \to \beta)} \qquad (5)$$

6. The old set of rule probabilities $\{P_{old}(X \to \lambda)\}$ is replaced with the new set $\{P_{new}(X \to \lambda)\}$, and the entire process is then repeated from Step 2 until a convergence threshold is reached.

After a number of iterations, this algorithm is guaranteed to converge to a local optimum that maximizes the probability of the training corpus. However, there is no guarantee that a global optimum will be found, and the initial probabilities chosen for the rules of the grammar have much influence on the converged probabilities. Although the resulting rule probability estimates can be skewed due to bad initial values as well as having been trained on incorrect as well as correct parse analyses, this iterative algorithm has the advantage of being unsupervised in the sense that it does not require a disambiguated training corpus.

2.3 Mixed Training

Considering the advantages and disadvantages of supervised and unsupervised training approaches, a new mixed training approach which combines the supervised training and unsupervised training is proposed in this paper. In this approach, the initial probability of each rule is derived using supervised training with a (small) disambiguated parse tree corpus, after which the probabilities are trained using unsupervised training. In the unsupervised training stage, the disambiguated parse tree corpus used in the initial supervised training stage is also included in the re-estimation of every iteration. From another viewpoint, this mixed training can also be considered as guided unsupervised training, which can guide the probability estimates away not only from bad initial values but also from bad values in each of subsequent iterations.

3 Experiments and Analysis

In order to evaluate different PCFG training approaches, we experiment on two core topics in the Chinese language processing: Part-of-Speech(POS) tagging and sentence parsing.

Two corpora are used in the experiments: the SPSTB TreeBank and the CKIP-TSINGHUA TagBank. The SPSTB TreeBank contains 8367 disambiguated parse sentences which are from Singapore primary school Chinese text books. The CKIP-TSINGHUA TagBank is a combination of two corpora: the CKIP[CKIP89][CKIP93][CKIP95] tagged corpus and the TSINGHUA tagged corpus. The CKIP corpus (version one) was developed by the Taiwan Chinese Knowledge Information Processing Group in 1995. It contains about 280,000 tagged sentences(about 2 million words) and has about 180 POS classes. The TSINGHUA tagged corpus was developed by TsingHua University of P.R.China in 1995. It contains about 25,000 tagged sentences(about 200,000 words) and has about 90 POS classes. The two systems of POS classes are merged together into 30 POS classes as in the SPSTB TreeBank. Thus the SPSTB TreeBank and CKIP-TSINGHUA TagBank have the same 30 POS classes.

In this paper, 453 PCFG parsing rules are derived from the SPSTB TreeBank. Additionally, the Chinese Word to POS conversion rules are derived from the CKIP-

TSINGHUA TagBank. There are about 28,000 words and about 38,000 conversion rules.

Training Data	POS Tagging	Sentence Parsing
1,000	90.1%	79.2%
2,000	91.7%	80.9%
3,000	93.0%	81.7%
4,000	93.9%	82.1%
5,000	94.2%	82.5%
6,000	94.5%	82.8%
7,000	94.7%	83.1%
8,000	94.9%	83.4%
8,367	95.1%	83.5%

Table 1: Supervised training

Tables 1 and 2 shows the POS tagging and sentence parsing accurate rates using the PCFGs of different numbers of training data with supervised and unsupervised training respectively. It is clear from Tables 1 and 2 that supervised training has better performance than unsupervised training even if the number of training senetences in the supervised training is only about 10% of that in the unsupervised training.

Training Data	POS Tagging	Sentence Parsing
10,000	91.2%	79.3%
20,000	92.9%	81.0%
30,000	93.4%	81.8%
40,000	93.8%	82.4%
50,000	94.1%	82.8%
60,000	94.5%	83.0%
70,000	94.7%	83.1%
80,000	94.8%	83.2%
90,000	94.8%	83.2%
100,000	94.9%	83.3%

Table 2: Unsupervised training

In order to compare the mixed training with supervised training and unsupervised training, following three sets of PCFGs are tested:

1) supervised training. All 8367 disambiguated parse trees from the SPSTB TreeBank are used.
2) unsupervised training. 80,000 tagged sentences from the CKIP-TSINGHUA TagBank are used to estimate the probabilities of PCFG parsing rules using the iterative unsupervised algorithm with uniform initial probabilities.
3) mixed training. In this method, 8367 disambiguated parse trees from the SPSTB TreeBank and 80,000 tagged sentences same as above from the CKIP-TSINGHUA TagBank are used to estimate the probabilities of PCFG parsing rules.

Table 3 shows the POS tagging and sentence parsing accurate rates using the PCFGs with different training approaches.

It is found from Table 3 that the PCFG using supervised training has slightly better performance than using unsupervised training while the PCFG using mixed training has much better performance than the PCFG using supervised training or unsupervised training alone.

Training Approach	POS Tagging	Sentence Parsing
Supervised	95.1%	83.5%
Unsupervised	94.8%	83.2%
Mixed	96.3%	86.4%

Table 3: Comparision of supervised, unsupervised and mixed training

To deep into the characteristics, Tables 4 and 5 show the POS tagging and sentence parsing accurate rates of various numbers of parsed sentences given consistent number of tagged sentences and various numbers of tagged sentences given consistent numbers of parsed sentences, respevtively, in the mixed training approach. Table 4 shows that the accurate rates improves greatly when the number of parsed sentences increases from 0 to 5,000 and improves slowly and stably afterwards. Table 5 shows that the accurate rates improves greatly when the number of targged sentences increases from 0 to 80,000 and improves slowly and stably afterwards.

Number of	POS	Sentence

Parsed Sentences	Tagging	Parsing
0	94.8%	83.2%
1,000	95.3%	84.1%
2,000	95.5%	84.7%
3,000	95.7%	85.3%
4,000	95.8%	85.7%
5,000	96.0%	86.0%
6,000	96.1%	86.2%
7,000	96.2%	86.3%
8,000	96.3%	86.4%
8,367	96.3%	86.4%

Table 4: Mixed training: various numbers of parsed sentences + 80,000 tagged sentences

Number of Parsed Sentences	POS Tagging	Sentence Parsing
0	95.1%	83.5%
10,000	95.5%	84.4%
20,000	95.7%	84.9%
30,000	95.8%	85.3%
40,000	96.0%	85.7%
50,000	96.1%	86.0%
60,000	96.1%	86.2%
70,000	96.2%	86.3%
80,000	96.3%	86.4%
90,000	96.3%	86.4%
100,000	96.4%	86.4%
110,000	96.4%	86.5%

Table 4: Mixed training: various numbers of tagged sentences + 8,367 parsed sentences

Conclusion

This paper studies and compares different PCFG training approaches. Considering the advantages and disadvantages of supervised and unsupervised training, a new mixed PCFG training approach is proposed. Compared with supervised training, it does not need to construct a very large corpus of disambiguated parse trees which is labor intensive. Compared with unsupervised training, it can guide the probability estimates away not only from bad initial values but also from bad values in each of subsequent iterations. Different PCFG trained by different training approaches are evaluated in the applications of Part-of Speech(POS) tagging and sentence tagging in the Chinese language. It is found that the mixed training approach has much better performance than supervised or unsupervised alone.

References

[Brown+92] Brown P.F. et al. "Class-based N-gram Models of Natural Language". *Computational Linguistics*, Vol.18, No.4, pp.467-479, 1992.

[CKIP89] Chinese Knowledge Information Processing Group (CKIP). "国语的词类分析（修订版）". *Technical Report*. Institute of Information Science. Academia Sinica, Taiwan. 1989.

[CKIP93] Chinese Knowledge Information Processing Group (CKIP). "词库小组技术报告". *Technical Report 93-05*. Institute of Information Science. Academia Sinica, Taiwan. 1993.

[CKIP95] Chinese Knowledge Information Processing Group(CKIP). "中央研 吭浩胶庥镱峡獾哪谌莺说明". *Technical Report 95-02*. Institute of Information Science. Academia Sinica, Taiwan. 1995.

[Corazza+91] Corazza A. et al. "Stochastic Context-Free Grammars for Island-Driven Probability Parsing". *Proceedings of second International Workshop on Parsing Technologies (IWPT' 91)*, Cancun, Mexico, pp.210-217, 1991.

[Fujisaki+91] Fujisaki T. et al. "A Probabilistic Parsing Method for Sentence Disambiguation". *Current Issues in Parsing Technology*, pp.139-152, 1991.

[Jelinek+90] Jelinek J.D. et al. "Basic Methods of Probabilistic Context Free Grammar". *Research Report RC 16374(#72684)*, IBM Research Division, T.J.Watson Research Center, 1990.

[Lari+90] Lari K. et al. "The Estimation of Stochastic Context-Free Grammars using the Inside-Outside Algorithm". *Computer, Speech and Language*, Vol. 4, pp.35-56, 1990.

[Resnik92] Resnik P. "Probabilistic Tree-Adjoining Grammar as a Framework for Statistical Natural Language Processing". *Proceedings of the 14th International Conference on Computational Linguistics*, 1992.

[Schabes+93] Schabes Y. et al. "Stochastic Lexicalized Context-Free Grammar". *Proceedings of the 3rd International Workshop on Parsing Technologies (IWPT'93)*, pp.257-266, Tilburg, Netherlands, 1993.