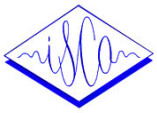
ISCA Archive
http://www.isca-speech.org/archive



International Symposium on Chinese Spoken Language Processing (ISCSLP 2004)
Hong Kong
December 15-18, 2004

EXPLOITING SYNTACTIC, SEMANTIC AND LEXICAL REGULARITIES IN LANGUAGE MODELING VIA DIRECTED MARKOV RANDOM FIELDS

Shaojun Wang, Shaomin Wang**, Russell Greiner*, Dale Schuurmans* and Li Cheng**
University of Alberta*, Massachusetts Institute of Technology**

ABSTRACT

We present a directed Markov random field (MRF) model that combines n-gram models, probabilistic context free grammars (PCFGs) and probabilistic latent semantic analysis (PLSA) for the purpose of statistical language modeling. The composite directed MRF model has potentially exponential number of loops and becomes context sensitive grammar, nevertheless we are able to estimate its parameters in cubic time using an efficient modified EM method, *the generalized inside-outside algorithm*, which extends inside-outside algorithm to incorporate the effects of the n-gram and PLSA language models.

1. INTRODUCTION

The goal of statistical language modeling is to accurately model the probability of naturally occurring word sequences in human natural language. The dominant motivation for language modeling has traditionally come from the field of speech recognition [7], however statistical language models have recently become more widely used in many other application areas, such as information retrieval, machine translation and bio-informatics.

There are various kinds of language models that can be used to capture different aspects of natural language regularity. The simplest and most successful language models are the Markov chain (n-gram) source models, first explored by Shannon in his seminal paper [11]. These simple models are effective at capturing local lexical regularities in text. However, many recent approaches have been proposed to capture and exploit different aspects of natural language regularity, sentence-level syntactic structure [3] and document-level semantic content [2, 6], with the goal of outperforming the simple n-gram model. Unfortunately each of these language models only targets some specific, distinct linguistic phenomena. The key question we are investigating is how to model natural language in a way that simultaneously accounts for the lexical information inherent in a Markov chain model, the hierarchical syntactic structure captured in a stochastic branching process, and the semantic content embodied by a bag-of-words mixture of log-linear models—all in a unified probabilistic framework.

Several techniques for combining language models have been investigated. The most commonly used method is simple linear interpolation [3, 10], where each individual model is trained separately and then combined by a weighted linear combination. The weights in this case are trained using held out data. Even though this technique is simple and easy to implement, it does not generally yield effective combinations because the linear additive form is too blunt to capture subtleties in each of the component models. Another approach is based on Jaynes' maximum entropy (ME) principle [8, 10] and first applied in language modeling a decade ago, and ever since it has become dominant technique in statistical natural language processing. In fact, the ME principle is nothing but maximum likelihood estimation (MLE) for undirected MRF models where ME is the primal problem formulation and MLE is the dual problem formulation. The major weakness with ME methods, however, are that they can only model distributions over explicitly observed features, whereas in natural language we encounter hidden semantic [2, 6] and syntactic information [3]. Recently we [12] proposed the latent maximum entropy (LME) principle, which extends standard ME estimation by incorporating hidden dependency structure. However, we have been unable to incorporate PCFGs in this framework, because the tree-structured random field component create intractability in calculating the feature expectations and global normalization over infinitely large configuration space. Previously we had envisioned that MCMC sampling methods [12] would have to be employed, leading to enormous computational expense.

In this paper, instead of using an undirected MRF model, we present a unified generative *directed Markov random field model* framework that combines n-gram models, PCFG and PLSA. Unlike undirected MRF models where there is a global normalization factor over infinitely large configuration space which often causes computational difficulty, the directed MRF model representation for the composite n-gram/syntactic/semantic model only requires local normalization constraints. More importantly it satisfies certain factorization property which greatly reduces the computational burden and makes the optimization tractable. To learn the composite model, by exploiting the factorization properties of the composite model, we use a simple yet efficient EM iterative optimization method, *the generalized inside-outside algorithm*, which enhances the well known inside-outside

algorithm [1] to incorporate the effects of the n-gram and PLSA language models. Given that n-gram, PCFG and PLSA models have each been well studied, it is striking that this procedure has gone undiscovered until now.

2. A COMPOSITE TRIGRAM/SYNTACTIC/ SEMANTIC LANGUAGE MODEL

Natural language encodes messages via complex, hierarchically organized sequences. The local lexical structure of the sequence conveys surface information, while the syntactic structure, encoding long range dependencies, carries deeper semantic information.

Assume that we use a trigram Markov chain to model local lexical information, and a PCFG to model the syntactic structure and PLSA [6] to model its semantic content of natural language, see Figure 1. Each of these models can be represented as a directed MRF model. If we combine these three models, we obtain a composite model that is represented by a rather complex chain-tree-table directed MRF model.

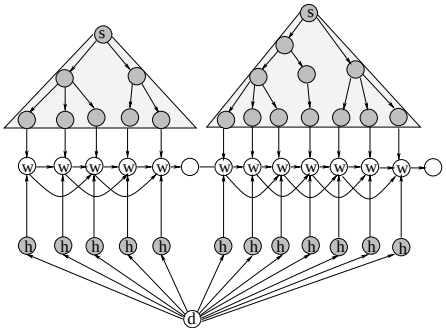


Fig. 1. The observables in natural language consist of words, sentences, and documents; whereas the hidden data consists of sentence-level syntactic structure and document-level semantic content. The figure illustrates a composite chain/tree/table model incorporating these aspects, where light nodes denote observed information and dark nodes/triangles denote hidden information.

A context free grammar (CFG) [1] G is a 4-tuple $(\Sigma, \mathcal{V}, \mathcal{R}, S)$ that consists of: a set of non-terminal symbols Σ whose elements are grammatical phrase markers; a vocabulary of $\mathcal{V} = \{v_1, \cdots, v_M\}$ whose elements, words v_i, are terminal symbols of the language; a sentence "start" symbol $S \in \Sigma$; and a set of grammatical production rules $\mathcal{R}$ of the form: $A \to \gamma$, where $A \in \Sigma$ and $\gamma \in (\Sigma \cup \mathcal{V})^*$. A PCFG is a CFG with a probability assigned to each rule, such that the probabilities of all rules expanding a given nonterminal sum to one. A PCFG is a branching process and can be treated as a directed MRF model, although the straightforward representation as a complex directed MRF is problematic.

PLSA [6] is a generative model of word-document co-occurrences by bag-of-words assumption as follows: (1) choose a document d with probability $\theta(d)$, (2) select a semantic class h with probability $\theta(d \to h)$, (3) pick a word w with probability $\theta(h \to w)$. The joint probability model for pair of (d, w) is a mixture of log-linear model with the expression $p(d, w) = \theta(d) \sum_h \theta(h \to w)\theta(d \to h)$. The latent class variables function as bottleneck variables to constrain word occurrences in documents.

When a PCFG is combined with a trigram model and PLSA, the grammar becomes context sensitive. If we view each uvw trigram as $uv \to w$, where $u, v, w \in \mathcal{V}$, then the composite trigram/syntactic/semantic language model can be represented as a directed MRF model, where the generation of nonterminals remains the same as in PCFG, but the generation of each terminal depends additionally on its surrounding context; i.e. not only its parent nonterminal but also the preceding two words as well as its semantic content node h.

3. TRAINING ALGORITHM FOR THE COMPOSITE MODEL

We are interested in learning a composite trigram/syntactic/ semantic model from data. We assume we are given a training corpus $\mathcal{W}$ consisting of a collection of documents $\mathcal{D}$, where each document contains a collection of sentences, and each sentence W is composed of a sequence of words from a vocabulary $\mathcal{V}$. For simplicity, but without loss of generality, we assume that the PCFG component of the composite model is in Chomsky normal form. That is, each rule is either of the form $A \to BC$ or $A \to w$ where $B, C \in \Sigma, w \in \mathcal{V}$. After combined with trigram and PLSA models, the terminal production rule $A \to w$ becomes $uvAh \to w$. By examining Figure 1, it should be clear that the likelihood of the observed data under this composite model can be written as below:

$$L(\mathcal{W}, \theta) = \prod_{d \in \mathcal{D}} \left(\prod_l p_\theta(d, W_l) \right) = \prod_{d \in \mathcal{D}} \left(\prod_l \left(\sum_{h \in \mathcal{H}} \theta(d \to h)^{n(d, W_l, h)} \right. \right.$$
$$\left. \sum_t \Big(\prod_{u,v \in \mathcal{V}, A \to w \in \mathcal{R}, h \in \mathcal{H}} \theta(uvAh \to w)^{n(uvAh \to w; d, W_l, t, h)} \right. \quad (1)$$
$$\left. \left. \prod_{A \to BC \in \mathcal{R}} \theta(A \to BC)^{n(A \to BC; d, W_l, t)} \Big) \right) \right)$$

where $p_\theta(d, W_l)$ is the probability of generating sentence W_l in document d, $n(d, W_l, h)$ is the count of semantic content h in sentence W_l of the document d, $n(uvAh \to w; d, W_l, t, h)$ is the count of trigrams uvw, the non-terminal symbol A and semantic content h in sentence W_l of document d with parse tree t and $n(A \to BC; d, W_l, t)$ is the count of nonterminal production rule $A \to BC$ in sentence W_l of document d with parse tree t. The parameters $\theta(d \to h), \theta(uvAh \to w), \theta(A \to BC)$ are normalized so that

$$\sum_{w \in \mathcal{V}} \theta(uvAh \to w) = 1$$
$$\sum_{BC \in \Sigma} \theta(A \to BC) = 1 \quad (2)$$
$$\sum_{h \in \mathcal{H}} \theta(d \to h) = 1$$

Thus we have a constrained optimization problem, and there will be a Lagrange multiplier for $uvAh$, nonterminal A and document d.

At a first glance, it seems that estimating parameters of the composite model is intractable since the composite directed MRF model has potentially exponential number of

loops, which suggests that loopy belief propagation and/or variational approximation methods have to be used. It turns out that this is not the case and there is an efficient and exact recursive EM iterative optimization procedure to perform this task.

Following Lafferty's [9] derivation of the inside-outside formulas for updating the PCFG parameters from a general EM [5] algorithm, we derive the generalized inside-outside algorithm for the composite language model. To apply the EM algorithm, we consider the auxiliary function

$$Q(\theta',\theta)=\sum_d\sum_l\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)\log\frac{p_{\theta'}(d,W_l,H_l,t)}{p_\theta(d,W_l,H_l,t)} \quad (3)$$

where H_l is the semantic content sequence of the sentence W_l.

Because of the normalization constraints (2), the reestimated parameters of the composite model are then the normalized conditional expected counts:

$$\begin{aligned}
&\theta'(A\to BC)\\
&=\frac{\sum_{d\in\mathcal{D}}\sum_l\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(A\to BC;d,W_l,t)}{\text{normalization over } BC}\\
&\theta'(uvAh\to w) \qquad (4)\\
&=\frac{\sum_{d\in\mathcal{D}}\sum_l\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(uvAh\to w;d,W_l,t,h)}{\text{normalization over } w}\\
&\theta'(d\to h)\\
&=\frac{\sum_{d\in\mathcal{D}}\sum_l\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(d\to h;d,W_l,t)}{\text{normalization over } h}
\end{aligned}$$

This looks very similar as the PCFG model. Thus we need to compute the conditional expected counts:

$$\sum_{d\in\mathcal{D}}\sum_l\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(A\to BC;d,W_l,t)$$

$$\sum_{d\in\mathcal{D}}\sum_l\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(uvAh\to w;d,W_l,t)$$

$$\sum_{d\in\mathcal{D}}\sum_l\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(d\to h;d,W_l,t)$$

In general, the sum requires summing over an exponential number of parser trees. However, just as for standard PCFGs, it is easy to check that the following equations still hold

$$\begin{aligned}
&\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(A\to BC;d,W_l,t)\\
&=\frac{\theta(A\to BC)}{p_\theta(d,W_l)}\frac{\partial p_\theta(d,W_l)}{\partial\theta(A\to BC)}\\
&\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(uvAh\to w;d,W_l,t)\\
&=\frac{\theta(uvAh\to w)}{p_\theta(d,W_l)}\frac{\partial p_\theta(d,W_l)}{\partial\theta(uvAh\to w)}\\
&\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(d\to h;d,W_l,t)\\
&=\frac{\theta(d\to h)}{p_\theta(d,W_l)}\frac{\partial p_\theta(d,W_l)}{\partial\theta(d\to h)}
\end{aligned}$$

and it turns out that there is an efficient way of computing the partial derivative on the righthand side, *the generalized inside-outside algorithm*.

Let $A\Rightarrow\gamma$ denote that, beginning with a nonterminal A, we can derive a string γ of words and nonterminals by applying a sequence of rewrite rules from the grammar *with the flowing-in trigrams and PLSA nodes*, where flowing-in trigrams and PLSA nodes are those which induce the words of the string γ.

Suppose the position of a rule $A\to BC$ within a tree t for sentence $W_l=(w_1,...,w_N)$ in document d can be specified by a triple $(i,j,k), i\le j\le k$. The partial derivative of the probability $p_\theta(S\to W_l \text{ in } d)=p_\theta(d,W_l)$ with respect to the parameter $\theta(A\to BC)$ only involves those parse trees which use the rule $A\to BC$. Consider the event "$S\to W_l$ in d using $A\to BC$ in position (i,j,k)". Because of the Markov property of the directed MRF model, the probability of this event can be written as a product of four terms, i.e. *the factorization property*, as follows:

$$\begin{aligned}
&p_\theta(S\to W_l \text{ in } d;\ \text{using } A\to BC \text{ in position } (i,j,k))\\
&=\ \theta(A\to BC)p_\theta(B\Rightarrow w_i\cdots w_j;W_l \text{ in } d)\\
&\quad p_\theta(C\Rightarrow w_{j+1}\cdots w_k;W_l \text{ in } d)\\
&\quad p_\theta(S\Rightarrow w_1\cdots w_{i-1}Aw_{k+1}\cdots w_N;W_l \text{ in } d)
\end{aligned}$$

See Figure 2 (a) for an illustration. The *key insight* toward a solution for the composite model is that, in comparison with the PCFG model, there are additional trigrams which connect the decomposition in position (i,j,k). These dependencies encode additional information from the trigram model, and significantly influence the parameter estimation of the non-terminal grammatical production rules (the impact of the PLSA model is implicitly considered, this will become clear when we derive the estimation formula for the terminal grammatical production rules). The factorization property is the crucial constituent for the success to derive an efficient and exact recursive algorithm.

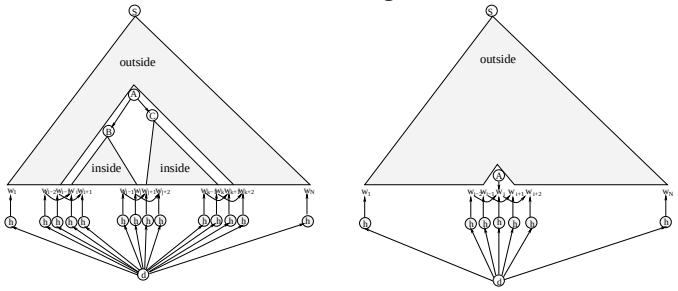


Fig. 2. Inside and outside probabilities in the composite trigram/syntactic/semantic model, where each component is influenced by the injected trigram and PLSA nodes.

From this it is not difficult to see that

$$\begin{aligned}
&\frac{\partial p_\theta(S\to W_l \text{ in } d)}{\partial\theta(A\to BC)}\\
&=\sum_{i\le j\le k}p_\theta(B\Rightarrow w_i\cdots w_j;W_l \text{ in } d)p_\theta(C\Rightarrow w_{j+1}\cdots w_k;W_l \text{ in } d)\\
&\qquad p_\theta(S\Rightarrow w_1\cdots w_{i-1}Aw_{k+1}\cdots w_N;W_l \text{ in } d)
\end{aligned}$$

Thus, the conditional expected number of times that the rule $A\to BC$ is used in generating the sentence $W_l\in\mathcal{W}$ in document d using the model θ is given by

$$\sum_{H_l}\sum_t p_\theta(H_l,t|d,W_l)n(A\to BC;d,W_l,t)=\frac{\theta(A\to BC)}{p_\theta(W_l \text{ in } d)}\Big(\sum_{i\le j\le k}\beta_{ik}(A;W_l \text{ in } d)\alpha_{ij}(B;W_l \text{ in } d)\alpha_{j+1k}(C;W_l \text{ in } d)\Big)$$

where $\alpha_{ij}(A; W_l \text{ in } d) = p_\theta(A \Rightarrow w_i \cdots w_j; W_l \text{ in } d)$

i.e., the inside probability that the nonterminal A, trigram parent nodes of w_i, w_{i+1} and document node d derive the word subsequence $w_i \cdots w_j$ in the sentence W_l of document d; and

$\beta_{ik}(A; W_l \text{ in } d) = p_\theta(S \Rightarrow w_1 \cdots w_{i-1} A w_{k+1} \cdots w_N; W_l \text{ in } d)$

i.e., the outside probability that beginning with the start symbol S, trigram parent nodes of w_{k+1}, w_{k+2} and document node d, we can derive the sequence $w_1 \cdots w_{i-1} A w_{k+1} \cdots w_N$ in the sentence W_l of document d.

Similarly consider the event "$S \rightarrow W_l$ using $uvAh \rightarrow w$ in d in position (i)". Because of the Markov property of the directed MRF model, the probability of this event can be written as a product of four terms, again *the factorization property*, as follows:

$$\begin{aligned} &p_\theta(S \rightarrow W_l \text{ in } d;\ \text{using } uvAh \rightarrow w \text{ in position } (i)) \\ &= \delta_{uvw}(w_{i-2}w_{i-1}w_i)\Big(\theta(d \rightarrow h)\theta(w_{i-2}w_{i-1}Ah \rightarrow w_i)\Big) \\ &\quad p_\theta(S \Rightarrow w_1 \cdots w_{i-1} A w_{i+1} \cdots w_N; W_l \text{ in } d) \end{aligned}$$

See Figure 2 (b) for illustration. The *key insight* toward a solution for the composite model is that comparing with the PCFG model, there are additional trigram and PLSA nodes which connect the decomposition in position (i) to encode the information of both trigram and PLSA nodes and make influencial impact for parameter estimation of the grammatical production rules $uvAh \rightarrow w$. Again, the factorization property is the crucial constituent for the success to derive an effi cient and exact recursive algorithm.

Thus we have

$$\sum_{H_l} \sum_t p_\theta(H_l, t|d, W_l) n(uvAh \rightarrow w; d, W_l, t) = \frac{\theta(uvAh \rightarrow w)}{p_\theta(W_l \text{ in } d)} \sum_{1 \le i \le N} \delta_{uvhw}(w_{i-2}w_{i-1}hw_i)\theta(d \rightarrow h)\beta_{ii}(A; W_l \text{ in } d)$$

where δ is the indicator function.

Now consider the event "$S \rightarrow W_l$ in d using $d \rightarrow h$ in position (i)". Because of the Markov property of the directed MRF model, the probability of this event can be written as sums of products of three terms as follows:

$$\begin{aligned} &p_\theta(S \rightarrow W_l \text{ in } d;\ \text{using } d \rightarrow h \text{ in position } (i)) \\ &= \sum_{A \in \Sigma} p_\theta(S \Rightarrow w_1 \cdots w_{i-1} A w_{i+1} \cdots w_N; W_l \text{ in } d) \\ &\qquad \Big(\theta(d \rightarrow h)\theta(w_{i-2}w_{i-1}Ah \rightarrow w_i)\Big) \end{aligned}$$

Thus we have

$$\sum_{H_l} \sum_t p_\theta(H_l, t|d, W_l) n(d \rightarrow h; d, W_l, t) = \frac{\theta(d \rightarrow h)}{p_\theta(W_l \text{ in } d)} \sum_{1 \le i \le N} \sum_{A \in \Sigma} \theta(w_{i-2}w_{i-1}Ah \rightarrow w_i)\beta_{ii}(A; W_l \text{ in } d)$$

Similar as in the PCFG case, there is an effi cient recursive method for computing the α's and β's using the CYK chart-parsing algorithm [13]. The only modifi cation is to the defi nition of α and β so that they incorporate additional information from the trigram and PLSA nodes. The method for doing this is almost the same as for PCFG and is implicit in the following recursive formulas:

$$\begin{aligned} \alpha_{ij}(A; W_l \text{ in } d) &= \sum_{BC} \sum_{i \le k \le j} \theta(A \rightarrow BC)\alpha_{ik}(A; W_l \text{ in } d) \\ &\qquad \alpha_{k+1j}(C; W_l \text{ in } d) \\ \alpha_{ii}(A; W_l \text{ in } d) &= \sum_h \theta(d \rightarrow h)\theta(w_{i-2}w_{i-1}Ah \rightarrow w_i) \\ \beta_{ij}(A; W_l \text{ in } d) &= \sum_{B,C} \sum_{k<i} \theta(B \rightarrow CA)\alpha_{ki-1}(C; W_l \text{ in } d) \\ &\qquad \beta_{kj}(B; W_l \text{ in } d) \\ &\quad + \sum_{B,C} \sum_{k>j} \theta(B \rightarrow AC)\alpha_{j+1k}(C; W_l \text{ in } d) \\ &\qquad \beta_{ik}(B; W_l \text{ in } d) \\ \beta_{1N}(A; W_l \text{ in } d) &= \delta_S(A; W_l \text{ in } d) \end{aligned}$$

To combat the sparse data problem in language modeling, we are able to generalize various smoothing techniques [4] to alleviate the sparseness of trigram counts in (4) eventhough there exist hidden variables A and h. We can also derive analogous algorithms to fi nd the most likely parse of a sentence and to calculate the probability of initial subsequence of a sentence, all generated by the composite language model. We will report experimental results somewhere else. If fully labelled data are available, then the composite model can be trained discriminatively, for the purpose of statistical parsing, either by maximizing conditional likelihood or maximizing margin of the directed MRFs, and the generalized inside-outside algorithm plays an important role in this task.

4. REFERENCES

[1] J. Baker. Trainable grammars for speech recognition. *Proceedings of the 97th Meeting of the Acoustical Society of America*, 547-550, 1979.

[2] J. Bellegarda. Exploiting latent semantic information in statistical language modeling. *Proceedings of IEEE*, 88(8):1279-1296, 2000.

[3] C. Chelba and F. Jelinek. Structured language modeling. *Computer Speech and Language*, 14(4):283-332, 2000.

[4] S. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 13(4): 319-358, 1999.

[5] A. Dempster, N. Laird and D. Rubin. Maximum likelihood estimation from incomplete data via the EM algorithm. *Journal of Royal Statistical Society*, 39:1-38, 1977.

[6] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1):177-196, 2001.

[7] F. Jelinek. *Statistical Methods for Speech Recognition*. MIT Press, 1998.

[8] S. Khudanpur and J. Wu. Maximum entropy techniques for exploiting syntactic, semantic and collocational dependencies in language modeling. *Computer Speech and Language*, 14(4):355-372, 2000.

[9] J. Lafferty. A derivation of the inside-outside algorithm from the EM algorithm. *IBM Research Report* 21636, 2000.

[10] R. Rosenfeld. A maximum entropy approach to adaptive statistical language modeling. *Computer Speech and Language*, 10(2):187-228, 1996.

[11] C. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(2):379-423, 1948.

[12] S. Wang, D. Schuurmans, F. Peng and Y. Zhao. Combining statistical language models via the latent maximum entropy principle. *Machine Learning*, to appear.

[13] D. Younger. Recognition and parsing of context free languages in time N^3. *Information and Control*, 10:198-208, 1967.