Learning Restricted Probabilistic Link Grammars

Eva Wai-man Fong **Dekai Wu**
HKUST
Department of Computer Science
University of Science & Technology
Clear Water Bay, Hong Kong
{wmfong,dekai}@cs.ust.hk

Abstract

We describe a language model employing a new *headed-disjuncts* formulation of Lafferty *et al.*'s (1992) probabilistic link grammar, together with (1) an EM training method for estimating the probabilities, and (2) a procedure for learning some simple lexicalized grammar structures. The model in its simplest form is a generalization of n-gram models, but in its general form possesses context-free expressiveness. Unlike the original experiments on probabilistic link grammars, we assume that no hand-coded grammar is initially available (as with n-gram models). We employ *untyped* links to concentrate the learning on lexical dependencies, and our formulation uses the *lexical* identities of heads to influence the structure of the parse graph. After learning, the language model consists of grammatical rules in the form of a set of simple disjuncts for each word, plus several sets of probability parameters. The formulation extends cleanly toward learning more powerful context-free grammars. Several issues relating to generalization bias, linguistic constraints, and parameter smoothing are considered. Preliminary experimental results on small artificial corpora are supportive of our approach.

1 Introduction

Although finite-state language models, in particular the linguistically simplistic trigram model, have occupied a predominant position in statistical language modeling, a context-free grammar formalism, with its enhanced ability to model center embeddings and long-distance dependencies, has the potential of outperforming the simpler models, given sufficiently constrained learning techniques for estimating the parameters reliably. We are studying *probabilistic link grammars*, a relatively new context-free grammar formalism with the desirable property of being highly lexicalized (Lafferty *et al.* 1992). In our models, the grammar rules (in the form of *disjuncts*) are induced in addition to their probabilities. We have extended the original probabilistic link grammar model to suit our purpose, and also made a number of revisions to better capture word associations and resolve the sparse data problem.

There have been attempts in learning stochastic context-free grammars (Lari & Young 1990; Jelinek *et al.* 1992), but the perplexities of these language models have so far been well above that of trigram language models (Brown *et al.* 1992). Newer approaches have focused on the more constrained context-free formalisms, such as dependency grammars (Carroll & Charniak 1992) and link grammars (Lafferty *et al.* 1992). Link grammar has the additional attraction of being highly lexical. Constraints in a grammatical theory reduce the number of possible production rules, and restrict the number of possible analyses for the data (Starosta 1991). Lexicality is a desirable feature since it enables the grammar to predict which words can go together, in addition to the grammatical rules in terms of parts of speech. (Lexicality accounts for the relative efficacy of n-gram language models compared with traditional context-free grammars.)

In this paper, we shall describe a novel approach to induce a restricted class of probabilistic link grammars which employs maximum-likelihood estimation from incomplete data. The training algorithm employed falls in the general class of EM algorithms (Baum 1972; Dempster *et al.* 1977). We shall also address issues such as appropriate parameter initialization, use of a suitable bias, and linguistic constraints to guide generalization.

2 Background

2.1 Link Grammar

Our point of departure is Lafferty *et al.*'s (1992) probabilistic model of link grammar. Link grammar is a relatively new context-free formalism for natural language proposed by Sleator & Temperley (1991). It does not have explicit constituents and possesses a high degree of lexicalization. In link grammar, a sequence of words is in the language if there exists at least one valid *linkage* for the words. Words (vertices) are connected by *links* (labelled arcs), and a linkage is valid if the resulting graph is connected and planar with arcs drawn above the words, and at most one arc connects a given pair of words. Figure 1 shows a link grammar parse (linkage) for an English sentence.

Rules in link grammar are in the form of *disjuncts*, where each disjunct corresponds to a usage of the word, or a particular way of satisfying its requirements. A disjunct comprises two ordered lists of connector names: the *left list* (connectors that link to the left of the current word) and the *right list* (connectors that link to the right of the current

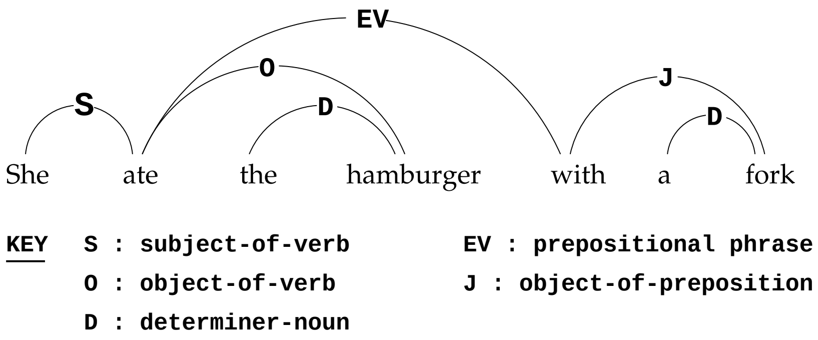

Figure 1: A link grammar parse

word). A disjunct is denoted as:

$$((L_1, L_2, \cdots, L_m)\ (R_n, R_{n-1}, \cdots, R_1))$$

which is equivalent to:

$$(L_1\ \&\ L_2\ \&\ \cdots\ \&\ L_m\ \&\ R_1\ \&\ R_2\ \&\ \cdots\ \&\ R_n)$$

There exists an efficient top-down link grammar parsing algorithm, presented in (Sleator & Temperley 1991). The generative probabilistic model of link grammar proposed in (Lafferty *et al.* 1992) was based on this top-down parsing algorithm.

2.2 Probabilistic Link Grammar

The basic operation of link grammar is *linking*. Here we use the term *link* to refer to a 7-tuple (W, d, O, L, R, l, r), where the word W with its disjunct d is linked to (at least one of) the words L and R via the connectors l and r. Thus a link depends on two connectors, a right connector and a left connector, say l and r, of two words L and R (L to the left of R) respectively. These connectors are roughly analogous to a nonterminal which is to be rewritten for a phrase structure grammar. A word W with its disjunct d is produced given the previously generated words and disjuncts. An *orientation* is also chosen for the link by deciding whether d links to l, to r, or to both l and r. (Hereafter, we use the random variable O to represent the orientation of a link. The possible values for O are $\leftarrow$, $\rightarrow$ and $\leftrightarrow$.)

The particular model in Lafferty *et al.* (1992) makes the simplifying assumption that a (W, d, O)-tuple is generated only based on the words nearest to the left and right that have already appeared, L and R, together with the associated connectors l and r. Thus the set of model parameters is

$$P(W, d, O|L, R, l, r).$$

Since each word in a sentence is generated exactly once, there is only one link probability $P(W, d, O|L, R, l, r)$ for each word. In an N-word sentence $W_1, W_2, ..., W_N$, a *linkage* (or a parse) is a set of N links $\{(W, d, O, L, R, l, r)\}$, and the probability of a linkage Γ is a product of N factors, together with an initial disjunct d_1 of word W_1:

$$(1)\quad P(S, \Gamma) \quad = \quad P(W_1, d_1) \prod P(W, d, O|L, R, l, r)$$

The probability of a sentence S is the sum of the probabilities of all its linkages, and thus we have:

$$P(S) = \sum_{\Gamma} P(S, \Gamma)$$

Under this model, the probability of a linkage does not depend on the order of traversal of the links – a given linkage only maps to one order in which the links are traversed, that order according to the top-down link grammar parsing algorithm. In this way, the derivation of a parse is not hidden and so does not need to be trained.

Note that the set of model parameters $P(W, d, O|L, R, l, r)$ may be rewritten as

$$P(W|L, R, l, r)P(d|W, L, R, l, r)P(O|d, W, L, R, l, r).$$

As this implies a huge set of conditional probabilities which may be impossible in practice for any training algorithm to form reliable estimates of them, these probabilities were approximated in (Lafferty *et al.* 1992) as

$$P(W, d, O|L, R, l, r) \approx \\ P(W|L, R, l, r)P(d|W, l, r)P(O|d, l, r).$$

An EM-based training algorithm, in the spirit of the Inside-Outside algorithm (Baker 1979), was used for estimating the model parameters. The set of rules (disjuncts) need to be coded manually.

3 A Framework for Learning Untyped-Link Grammars

We now describe our framework for inducing probabilistic link grammars, assuming that no *a priori* hand-coded grammar is available. We discuss first our representational constraints, then the learning procedure.

3.1 Expressiveness Constraints

An *a priori* grammar would constrain the legal disjuncts for each word to some predetermined set. In the absence of such constraints, it is necessary to restrict the space of allowed disjuncts by other means. We imposed the following two conditions.

Bounded number of connectors If no limit is placed on the number of connectors in a disjunct, the search space would be infinite, and learning would not be possible. On studying linkage patterns of sentences by means of Sleator and Temperley's (non-probabilistic) link grammar (Sleator & Temperley 1991), we observe that most words would not be linked to many words in a given sentence. In particular, it is rare for words to need disjuncts with more than three connectors on either of the left and right connector lists (with the exception of very complex sentences and disjuncts having multi-connectors [1], which are not addressed in this basic model). Therefore, we postulate that a large portion of the sentences can be parsed using only the relatively simple disjuncts, that is, those disjuncts with a small number of connectors.

To make learning feasible, we need to restrict the number of possible disjuncts under our framework. We use the number of connectors in disjuncts to form our framework, and restrict our attention to those disjuncts with a maximum number of N left and N right connectors, where N is specified as a parameter to the learning algorithm.

Untyped links To further reduce the space of possible disjuncts, we made the simplification of allowing for only one type of link. Since the number of disjuncts is exponential in the order of the number of link types, learning would be extremely difficult without this restriction. A more complex model, with a large number of link types corresponding to different relationships between the connected words, generally possesses greater power in modeling the data at its full potential. However, the complexity of the model would increase the dimension of learning, making it easy for the generalization process to get stuck in local maxima, and hard to obtain reliable estimates of the model parameters.

Notice that our restriction is somewhat analogous to context-free bracketing grammars which possess only a single type of nonterminal, except that the link grammar formulation still allows lexical constraints to be expressed. The resulting language model supports more abstract structural dependencies than pure n-gram models, but without losing the lexicality advantages of n-grams.

3.2 Learning Procedure

Learning proceeds in three stages:

1. Initialize probabilities
2. Train on corpus
3. Prune disjuncts

Initialization Since EM only guarantees convergence to local maxima instead of a globally maximal solution, it is important that the parameter estimation process starts with good initial estimates. We used windowed trigram probabilities for initializing the parameters $P(W|L, R, l, r)$.

[1] connectors that are endpoints of more than one link

Training We have studied two variants of EM training. The original method is described in Lafferty *et al.* (1992). Our new formulation for the headed-disjuncts model is described in Section 6.

Pruning At the end of the EM estimation process, disjuncts with very low probabilities are pruned and the probability values are renormalized. Manual analysis of the parses produced by the induced grammar confirmed that pruning improves the quality of the generalization by reducing the number of parses for a sentence and boosting probabilities of the more promising linkages. Disjunct pruning can be applied to each iteration of the parameter estimation process to speed up generalization, provided that the threshold is set at a suitable level. The success of this greedy strategy depends on the choice of the pruning threshold and the sensitivity of the results to this value. To avoid omitting useful disjuncts, we have chosen to prune the low probability disjuncts only at the end of the training process.

4 Problems with Lafferty *et al.*'s Formulation

For our initial experiments with the learning procedure, we employed the same probabilistic approximations as in Lafferty *et al.*'s (1992) original formulation, which we refer to hereafter as the LST model. However, under these approximations, we were unable to obtain satisfactory results (see Sections 8–10). Inspection revealed several factors that contributed to the low performance.

Dependence on link type information With the untyped links that make learning feasible, the model parameters—$P(W|L, R, l, r)$, $P(d|W, l, r)$ and $P(O|d, l, r)$—become less predictive, since the type of word association usually gives some clue about the word that is being generated next. For instance, the probability of linking the noun `dog` to the preceding article `the` would be high if it is expecting a determiner-noun relationship in its disjunct being currently considered, but would be extremely low if it is looking for a word to fulfill the object-of-verb relationship. However, an untyped-link model does not distinguish between these two cases, and both are assigned the same parameter values. Thus it would be more difficult for the learning process to generalize well without information about the types of the links.

Sensitivity to sparse data Results for the LST model have not been reported in literature, and they would also depend on how the link types are classified. However, on analyzing the model parameters, we find that the model is extremely prone to the *sparse data problem*, due to its huge number of parameters some of which are not normalized and thus contain redundancies. We are particularly concerned with the set of probabilities $P(W|L, R, l, r)$, which may not be apt for modeling word correlations.

As an illustration, consider the association between the words `the` and `ball`. Instead of parameters involving just the two associated words, we also need a third word to make up the parameter $P(W|L,R,l,r)$. Temporarily ignoring the connector types, we need parameters of the form $P(\texttt{the}\ |L\ \texttt{ball})$, where L is a word in the vocabulary, to model the determiner-noun relationship between `the` and `ball`, instead of just the single parameter $P(\texttt{the}\ |\ \texttt{ball})$. Suppose we are able to deduce from the training corpus the probabilities

$P(\texttt{the}\ |\ \texttt{hit ball})$,
$P(\texttt{the}\ |\ \texttt{kick ball})$,
$P(\texttt{the}\ |\ \texttt{throw ball})$,
$P(\texttt{the}\ |\ \texttt{have ball})$,
$P(\texttt{the}\ |\ \texttt{to ball})$.

But these are not going to be of use when we are given sentences with phrases containing `the ball` but are not covered in the above, e.g. `The new chairman is on the ball`. It would be rare that a corpus contains all the possible words that can occur with a binary association between two other words. Thus the problem of insufficient training data is exaggerated.

Insensitivity to orientations Furthermore, we noticed that under this probabilistic framework it is difficult to model the exact associations between words. Given two words, L and R, there are three ways, i.e. *orientations*, in which they generate a word W, where ($L < W < R$): W may be linked to L, to R, or to both of them. All three orientations will be assigned the same values for $P(W|L,R,l,r)$ and $P(d|W,l,r)$, though perhaps a different $P(O|d,l,r)$. However, the first two parameters, namely the word probabilities and disjunct probabilities, are exactly the more influential ones, from which we observe a much greater variance and also predictive power.

Whereas the deficiency of link types can be avoided by finding the right link types and then the disjuncts either manually or by some clustering method, before or after the link structures are induced, the other two problems may be tackled by reformulating the probabilistic model.

5 Formulation of the Headed-disjuncts Model

In reformulating the probabilistic model, our objective is to incorporate the orientation factor into the probabilities for choosing a word and a disjunct in the linking process, while at the same time reducing the size of the parameter space.

Recall that the parameters of probabilistic link grammar are the set of *link probabilities*

$$P(W,d,O|L,R,l,r) = P(W|L,R,l,r)P(d|W,L,R,l,r)P(O|d,W,L,R,l,r)$$

which were approximated by

$$P(W|L,R,l,r)P(d|W,l,r)P(O|d,l,r).$$

On the other hand, we can also rewrite $P(W,d,O|L,R,l,r)$ as

$$P(O|L,R,l,r)P(W|L,R,l,r,O)P(d|W,L,R,l,r,O)$$

which makes it easier to capture the true association between W, L and R.

The first of these parameters, $P(O|L,R,l,r)$, may be approximated by $P(O|l,r)$, since knowing the identities of the words L and R may not be of much help in predicting the orientation for linking them to a word, without knowing which word they are linking to.

Therefore, we have

$$P(W,d,O|L,R,l,r) = P(O|l,r)P(W|L,R,l,r,O)P(d|W,L,R,l,r,O). \quad (2)$$

For our untyped-link model, a connector is either present or nil. This has implications on the values of $P(O|l,r)$, as shown in Table 1. The case for $l = r \neq$ NIL needs a bit of explanation. In the parsing algorithm, linking to the left and linking to the right are intentionally made assymetric to avoid overcounting. Since the $\rightarrow$ orientation is allowed only when l is NIL, we have only two possible orientations when both l and r are not NIL. The only orientation probabilities that need to be estimated are $P(\leftrightarrow|l,r)$ and $P(\leftarrow|l,r)$ where both l and r are not NIL.

Table 1: Relationship between orientation and connectors l and r

l	r	Orientation allowed
present	NIL	$\leftarrow$
NIL	present	$\rightarrow$
present	present	$\leftrightarrow, \leftarrow$

By including the orientation into the conditional probabilities $P(W|L,R,l,r,O)$ and $P(d|W,L,R,l,r,O)$, it is possible to capture details on the choice of words and disjuncts associated with the different orientations, while at the same time reducing the size of the parameters.

In deriving a formula for P_w, we consider three cases, each corresponding to an orientation. For the orientations where W is linked with one of L and R, we may simply omit the unlinked word from the conditional probability, as it does not play a role in generating W. For the orientation $\leftrightarrow$, whereby W is linked to both L and R (we may think of this as W is generated by both L and R), we can derive the following expression, assuming L, l and R, r are independent:

$$P(W|L,R,l,r) = \frac{P(W|L,l)P(W|R,r)}{P(W)}$$

Thus we may approximate $P(W|L,R,l,r,O)$ by

$$P(W|L,R,l,r,O) \approx \begin{cases} P(W|L,l) & \text{if } O = \leftarrow \\ P(W|R,r) & \text{if } O = \rightarrow \\ \frac{P(W|L,l)P(W|R,r)}{P(W)} & \text{if } O = \leftrightarrow \end{cases} \quad (3)$$

Similarly, we can derive an expression for P_d as follows:

$$(4)\quad P(d|W,L,R,l,r,O) \approx \begin{cases} P(d|W,L,l) & \textit{if } O = \leftarrow \\ P(d|W,R,r) & \textit{if } O = \rightarrow \\ \dfrac{P(d|W,L,l)P(d|W,R,r)}{P(d|W)} & \textit{if } O = \leftrightarrow \end{cases}$$

The prime parameters that we need to estimate for the model are thus: $P(W|L,l)$, $P(W|R,r)$, $P(d|W,L,l)$, $P(d|W,R,r)$, $P(d|W)$, and $P(W)$, where $P(W)$ is equivalent to the unigram probabilities which are independent of our model and can be computed before the parameter estimation process.

We note that the parameters in the revised model, are in the order of bigrams, whereas the parameters $P(W|L,R,l,r)$ in the original probabilistic model are in the order of trigrams. We may achieve further economy in the number of model parameters by using $P(d|W,l)$ and $P(d|W,r)$ instead of $P(d|W,L,l)$ and $P(d|W,R,r)$, but we believe that the latter set would be more representative of the underlying probabilistic events, especially for an untyped-link model.

For initializing the parameters $P(W|L,l)$ and $P(W|R,r)$, we can make use of *windowed* bigram probabilities, distorted by *mutual information* scores so that links between highly correlated words are preferred.

We call this formulation the *headed-disjuncts* model (henceforth HD), because the disjunct preferences of each word in the sentence are influenced by its lexical heads.

6 EM Estimation for the Headed-disjuncts Model

The parameter estimation process of the revised model is similar to that of the LST model. The inside and outside probabilities are computed in the same top-down manner, but with the link probabilities $P(W,d,O|L,R,l,r)$ decomposed in a different way, as described in Equations 2, 3, and 4.

After calculating the inside and outside probabilities of a sentence, the training algorithm proceeds with updating the counts for the parameters. The complexity of the training algorithm is $O(D^3N^3)$, where D is an upper bound on the number of disjuncts of an arbitrary word in the grammar, and N the number of words in the sentence. The counts that we need to obtain for estimating the model parameters are shown in Table 2.

Since this revised model separates the treatment of the word and disjunct probabilities for the different orientations, the calculations for the associated counts also differ. For instance, we only increment $Count(W,L,l)$ for linkages in which W and L are connected, that is, the orientation is either $\leftarrow$ or $\leftrightarrow$ when we link W to L and some other word R to the right.

Table 2: List of Counts to be computed for HD Model

Parameter estimated	Counts needed
$P(W|L,l)$	$Count(W,L,l), Count(L,l)$
$P(W|R,r)$	$Count(W,R,r), Count(R,r)$
$P(d|W,L,l)$	$Count(d,W,L,l), Count(W,L,l)$
$P(d|W,R,r)$	$Count(d,W,R,r), Count(W,R,r)$
$P(d|W)$	$Count(d,W), Count(W)$

In the formulas for counts that follow, we let

$$\begin{aligned} P_L &= P(\leftarrow|l,r)\ P(W|L,l)\ P(d|W,L,l) \times \\ &\quad P_I(L,W,l\triangleright,\triangleleft left[d])\ P_I(W,R,right[d],r) \\ P_R &= P(\rightarrow|l,r)\ P(W|R,r)\ P(d|W,R,r) \times \\ &\quad P_I(L,W,l,left[d])\ P_I(W,R,right[d]\triangleright,\triangleleft r) \\ P_{LR} &= \\ &\quad P(\leftrightarrow|l,r) \times \\ &\quad \frac{P(W|L,l)P(W|R,r)}{P(W)} \frac{P(d|W,L,l)P(d|W,R,r)}{P(d|W)} \times \\ &\quad P_I(L,W,l\triangleright,\triangleleft left[d])\ P_I(W,R,right[d]\triangleright,\triangleleft r) \end{aligned}$$

$$\begin{aligned} Count(W,L,l) &= P(S)^{-1} \sum_{R>W,r} P_O(L,R,l,r) \times \\ &\quad \sum_{d\in D(W)} \{P_L + P_{LR}\} \\ Count(W,R,r) &= P(S)^{-1} \sum_{L<W,l} P_O(L,R,l,r) \times \\ &\quad \sum_{d\in D(W)} \{P_R + P_{LR}\} \\ Count(d,W,L,l) &= P(S)^{-1} \sum_{R,r} P_O(L,R,l,r) \times \\ &\quad \{P_L + P_{LR}\} \\ Count(d,W,R,r) &= P(S)^{-1} \sum_{L,l} P_O(L,R,l,r) \times \\ &\quad \{P_R + P_{LR}\} \\ Count(d,W) &= P(S)^{-1} \sum_{L,R,l,r} P_O(L,R,l,r) \times \\ &\quad \{P_L + P_R + P_{LR}\} \\ Count(L,l) &= \\ &\quad P(S)^{-1} \sum_{R,r} P_O(L,R,l,r)\ P_I(L,R,l,r) \\ Count(R,r) &= \\ &\quad P(S)^{-1} \sum_{L,l} P_O(L,R,l,r)\ P_I(L,R,l,r) \end{aligned}$$

7 Regularization Issues

7.1 Generalization Bias

We would like our grammar induction process to produce generalizations that favor simpler linkages, following Occam's razor. We implement this preference by adding to the counts computed from the original corpus a set of virtual counts which follow a predetermined distribution favoring simpler disjunct categories in the complexity hierarchy (Table 3). This is analogous to augmenting the input space with a set of input vectors containing a specified distribution of certain features. In our case, these features are the counts for disjuncts used in valid linkages. The bias helps guide the learning process toward the correct solution space.

Table 3: Disjunct complexity in ascending order, for a maximum of 2 connectors

Rank	Disjunct configurations	
1	$((l_1)$ NIL)	(NIL $(r_1))$
2	$((l_1)\ (r_1))$	
3	$((l_1, l_2)$ NIL)	(NIL $(r_1, r_2))$
4	$((l_1, l_2)\ (r_1))$	$((l_1)\ (r_1, r_2))$
5	$((l_1, l_2)\ (r_1, r_2))$	$((l_1, l_2)\ (r_1, r_2))$

7.2 Linguistic Constraints

On analyzing the preliminary experiment results, we found that determiners (such as `a`, `an`, `the`, `his`, `her`, etc.) have been causing confusion to the learning process. Traditional linguistic schools usually regard determiners as *modifiers* of their *head* nouns, which may in turn be modifiers of some other words in the sentence, such as the main verb.[2] Since in English a determiner always appears to the left of its associated noun, it can only be connected with a noun to the right. Obviously a determiner in English does not have disjuncts with one or more left connectors. This is an example of using linguistic knowledge to limit the number of disjuncts to be considered when constructing hypothetical linkages in the grammar induction process.

We remark that the linguistic constraints we may introduce into the learning process are language dependent; different languages generally require different sets of applicable linguistic constraints. A small number of constraints is usually effective in pruning a large number of invalid hypotheses and helps the induction process concentrate on the remaining hypotheses which are likely to lead to a grammar that possesses a greater amount of linguistic flavor. Of course, this is based on our hypothesis that linkages conforming to linguistic theories tend to yield better language models.

[2] Whether a determiner or the associated noun should be the head is still controversial. But for the purpose of predicting word co-occurrences and grammar induction, we consider the traditional approach to be more suitable.

7.3 Smoothing

Obtaining reliable estimates of the parameters has always been a fundamental issue for probabilistic language models. We have used windowed N-grams to smooth our model parameters. The smoothed estimates are given by

$$\tilde{P}(W|L,l) = \gamma^{-1}\delta_l(W)[\lambda P(W|L) + (1-\lambda)P(W|L,l)]$$

$$\tilde{P}(W|R,r) = \gamma^{-1}\delta_r(W)[\lambda P(W|R) + (1-\lambda)P(W|R,r)]$$

where

$$\delta_l(W) = \begin{cases} 1, & W \textit{ has a disjunct that can link to } l \\ 0, & \textit{otherwise} \end{cases}$$

$$\delta_r(W) = \begin{cases} 1, & W \textit{ has a disjunct that can link to } r \\ 0, & \textit{otherwise} \end{cases}$$

and γ is a normalizing constant. λ takes on values between 0 and 1, and it can be estimated from the training data to maximize the predictive power of the resulting distribution. For many natural language tasks, simply assigning a reasonable constant value to λ can yield satisfactory performance.

8 Experimental Setup

The aim of the experiments is to test the viability of our approach and facilitate analysis of the behaviour of the learning procedure, and hence corpora comprising relatively short and simple sentences are suitable. As we were unable to locate available simple English corpora, we have constructed two experimental English corpora (see Table 4). We generated parameters for both the LST and HD models, as well as for the bigram model.

Since the number of hypotheses the training algorithm needs to consider grows exponentially with the maximum number of connectors allowed in a disjunct, we set the limit of two left and two right connectors in the disjuncts for experimental purpose. We expect this would be adequate for most of our training and testing data which contain relatively short English sentences and few adjectives[3]. The threshold for pruning disjuncts has been set to 0.001.

We have also used the following linguistic constraints to guide the induction process:

- *determiners* have no left connectors and have only one right connector in their disjuncts;
- *verbs* have at least one left connector (for connecting to the subject). This left connector should link to a special WALL word in imperative sentences;
- *prepositions*, when they appear in lowercase, have exactly one left connector in their disjuncts.

[3] We have purposely selected sentences without two consecutive adjectives modifying a noun, since that would necessitate disjuncts with greater than two left connectors for the noun. Sleator and Temperley introduced the notion of *multiconnectors* which can be linked to more than one connector (Sleator & Temperley 1991), but this has not been incorporated into any probabilistic model we are aware of.

Table 4: Particulars of corpora used in experiments

Corpus	# Sentences	Vocabulary Size	Max. Length	Training	Testing
A	320	57	8	80%	20%
B	600	92	8	80%	20%

These constraints are simple to formulate, and can largely reduce the number of possibilities needed to be considered during the induction process.

9 Parsing Results

We first evaluated our probabilistic models by comparing the resulting parse structures and their derivation probabilities. These parses (or linkages) were generated by a probabilistic link grammar parser, which read in a set of probabilistic grammar rules, and output for each input sentence all its valid linkages together with their derivation probabilities.

Our evaluation criterion is the *correctness of parsing*. For each sentence in the test set, linkages (parse structures) that score the highest probabilities in each of the models are compared. A linkage which resembles the parse by humans is preferred to one which does not.

In our evaluation, we classify the mostly likely parse (i.e. one with the highest derivation probability) to be *correct* if it is the same as a parse (in link grammar) which a human would give for the sentence. Since a linkage which is not exactly correct may still contribute to good probability evaluations, we allow for a class of *acceptable* linkages, which contains the correct linkages as well as those with only minor deviations from the correct one which the human evaluator regards as tolerable. We restrict an acceptable linkage to differ from a correct parse by at most one misplaced, or extraneous, link. An example of an acceptable, but not exactly correct, linkage is shown in Figure 2. Linkages that are not acceptable are classified as *wrong*.

Acceptable linkage:

```
                        +---L--+
    +---L---+--L-+-L-+-L+  +-L-+
    |       |    |   |  |  |   |
Yesterday they went to see a film
```

Correct linkage:

```
    +------L-----+      +---L--+
    |       +--L-+-L-+-L+  +-L-+
    |       |    |   |  |  |   |
Yesterday they went to see a film
```

Figure 2: Examples of acceptable and correct linkages.

We also carried out the above evaluation for the top three scoring linkages for each sentence in the test set. In this case, a sentence is regarded as having been parsed correctly if at least one among these three is a correct linkage for the sentence.

We observed that the HD approximation yields superior performance to the LST approximation under this evaluation criterion.

Table 5: Parsing correctness (measured using the most likely parse)

Model	% correct parses	% acceptable parses
LST	46.0	60.0
HD	68.75	85.9

Table 6: Parsing correctness (measured using the 3 most likely parses)

Model	% correct parses	% acceptable parses
LST	47.6	63.4
HD	85.9	92.2

10 Perplexity Results

The predictive accuracy of language models is often compared by evaluating their *perplexities* with respect to some common test data. For the same data set, a smaller perplexity measure implies a better model.

Perplexities with respect to our two corpora are computed for each of our probabilistic link grammar models, as well as the bigram model. The results are shown in Table 7.

Table 7: Model comparison using perplexity measures

Corpus	Bigram	LST	HD
A	25.09	48.90	28.41
B	32.64	39.94	17.89

Since corpus B contains a larger number of sentences, the perplexity values computed are more reliable. The HD model is superior to the bigram model according to the perplexity measures for corpus B and comparable for corpus A. Note however that the bigram perplexities are supplied for reference and are not directly comparable, because the corpora are too small to compensate for the larger number of parameters in the link grammar models. More significantly, for both corpora, the HD approximation yields significantly better predictive measures than the LST approximation.

11 Conclusion and Future Work

We believe that trend for probabilistic grammar models is towards increased *expressiveness* (as in the context-free formalism), *constrainedness* (as in dependency grammar and link grammar), and *lexicality* (as in link grammar and lexicalized tree-adjoining grammar).

In our work, we studied the viability of inducing probabilistic link grammars from corpora. The original model probabilistic link grammar proposed by Lafferty *et al.* (1992) was simplified to the untyped-link case with a limit on the number of connectors in the disjuncts, to make learning the grammar rules feasible. Later on, we also made some revisions to the probabilistic model in order to better capture word associations and to address the sparse data problem. We have also addressed a number of issues in improving the generalization process.

To test the viability of our approach, we have conducted experiments using English corpora of limited vocabulary and relatively simple sentences. Though our preliminary experiments are not on a very large scale, they are supportive of our approach. We are planning to conduct experiments for corpora of substantial sizes, to yield more representative perplexity measures.

The next step is to extend our grammar induction approach to deal with real corpora with larger vocabularies and more complex sentences. Learning can start with the short and simple sentences and proceed incrementally to the more complex sentence structures, coupled with the introduction of more complex disjuncts.

Acknowledgements

We are grateful to John Lafferty for explaining details of his probabilistic link grammar model; to Li Deng and Dit-Yan Yeung for valuable comments.

References

BAKER, JAMES K. 1979. Trainable grammars for speech recognition. In *Speech Communication Papers for the 97th Meeting of the Acoustic Society of America*, ed. by D. H. Klatt & J. J. Wolf, 547–550.

BAUM, LEONARD E. 1972. An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. *Inequalities*, 3(1):1–8.

BROWN, PETER F., STEPHEN A. DELLAPIETRA, VINCENT J. DELLAPIETRA, JENNIFER C. LAI, & ROBERT L. MERCER. 1992. An estimate of an upper bound for the entropy of English. *Computational Linguistics*, 18(1):31–40.

CARROLL, GLENN & EUGENE CHARNIAK. 1992. Learning probabilistic dependency grammars from labelled text. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, 25–32, Cambridge, MA. Proceedings to appear as AAAI technical report.

DEMPSTER, A. P., N. M. LAIRD, & D. B. RUBIN. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B*, 39:1–38.

JELINEK, FREDERICK, JOHN D. LAFFERTY, & ROBERT L. MERCER. 1992. Basic methods of probabilistic context-free grammars. In *Speech Recognition and Understanding: Recent Advances, Trends, and Applications*, Nato ASI Series, Vol. F75, 345–360. Springer Verlag.

LAFFERTY, JOHN D., DANIEL SLEATOR, & DAVY TEMPERLEY. 1992. Grammatical trigrams: A probabilistic model of link grammar. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, 89–97, Cambridge, MA. Proceedings to appear as AAAI technical report.

LARI, K. & S. J. YOUNG. 1990. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, 4:35–56.

SLEATOR, DANIEL & DAVY TEMPERLEY. 1991. Parsing English with a link grammar. Technical Report CMU-CS-91-196, Dept. of Computer Science, Carnegie Mellon University.

STAROSTA, STANLEY. 1991. Dependency grammar and monostratal transfer. In *2nd Japan-Australia Joint Symposium on Natural Language Processing*.