ELSEVIER

Cognitive Science 26 (2002) 239–253

COGNITIVE SCIENCE

http://www.elsevier.com/locate/cogsci

Statistical models of syntax learning and use

Mark Johnson [a,*], Stefan Riezler [b,1]

[a] *Department of Cognitive and Linguistic Sciences, Brown University, P.O. Box 1978, Providence, RI 02912, USA*
[b] *Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, CA 94304, USA*

Accepted 22 March 2002

Abstract

This paper shows how to define probability distributions over linguistically realistic syntactic structures in a way that permits us to define language learning and language comprehension as statistical problems. We demonstrate our approach using lexical-functional grammar (LFG), but our approach generalizes to virtually any linguistic theory. Our probabilistic models are maximum entropy models. In this paper we concentrate on statistical inference procedures for learning the parameters that define these probability distributions. We point out some of the practical problems that make straightforward ways of estimating these distributions infeasible, and develop a "pseudo-likelihood" estimation procedure that overcomes some of these problems. This method raises interesting questions concerning the nature of the data available to a language learner and the modularity of language learning and processing.

Keywords: Statistical language learning; Statistical parsing; Maximum entropy modeling; Discriminative parameter estimation

1. Introduction

This paper summarizes our recent work in developing statistical models of syntax which are compatible with the kinds of syntactic structures posited by current linguistic theories. In a series of papers we have developed tools for estimating or "learning" such models from data (Johnson, Geman, Canon, Chi, & Riezler, 1999; Johnson & Riezler, 2000; Riezler, Prescher, Kuhn, & Johnson, 2000) and this paper provides a high-level overview of both the general approach and the methods we developed.

* Corresponding author. Tel.: +1-401-863-1670; fax: +1-401-863-2255.
E-mail addresses: mark_johnson@brown.edu (M. Johnson), riezler@parc.com (S. Riezler).
[1] Tel.: +1-650-812-4663; fax: +1-650-812-4374.

0364-0213/02/$ – see front matter
PII: S0364-0213(02)00068-X

Turning to theoretical results on learning, it seems that statistical learners may be more powerful than non-statistical learners. For example, while Gold's famous results showed that neither finite state nor context-free languages can be learnt from positive examples alone (Gold, 1967), it turns out that *probabilistic* context-free languages can be learnt from positive examples alone (Horning, 1969).[1] Statistics provides the theory of optimal learners and optimal comprehenders (optimal in an information-theoretic sense) which serve as idealizations of, and upper bounds to, human performance. If an optimal statistical learner fails to learn a language given certain kinds of inputs (say, phonological forms alone) under certain assumptions about universal grammar, then we can be fairly certain that human beings either have access to richer data or have stronger biases that restrict the class of possible grammars.

An immediate goal of this research is to find a way of defining probability distributions over linguistically realistic structures in a way that permits us to define language learning and language comprehension as statistical problems, and the rest of this paper concentrates on these questions. The next section describes the linguistic theory, lexical-functional grammar (LFG), which defines the linguistic structures used in this research, and Section 3 explains how we define a probability distribution over these structures. Section 4 describes how one can learn the parameters that define probability distributions over these structures in principle, and points out some of the practical problems that make straightforward ways of estimating these distributions infeasible. This leads us to the "pseudo-likelihood" estimation methods described in Section 5, which also raise interesting questions concerning the nature of the data available to the child and modularity of language learning and processing.

2. Lexical-functional grammar

This research differs from most work in statistical computational linguistics in that it is compatible with and builds on the results of modern linguistic theory. While our approach is compatible with virtually all existing theories of grammar (including transformational grammar and minimalist grammars), we have adopted the framework and structures of LFG in our research. LFG has several properties that make it especially well suited for research involving linguistically-oriented probabilistic grammars. The formal definition of LFGs and the structures they generate is clear and precise (Kaplan, 1995), and LFG provides simple, clean descriptions of a wide range of typologically diverse linguistic phenomena (Bresnan, 1982). There is also a substantial amount of existing computational research on LFG, including efficient parsing with large grammars (Maxwell & Kaplan, 1993), which we exploit in our research.

An LFG representation of a sentence consists of a small number of distinct components, such as the phonological structure, the syntactic structure, the semantic interpretation, etc. To keep things simple in this paper, however, we will only use a subset of these components and simplify them where appropriate. For example, we take the phonological component of a sentence to be just a string of words, and ignore prosody and other phonological details. Similarly, we take the semantic interpretation of a sentence to be its predicate-argument structure (roughly, "who did what to whom"), and ignore mood, tense, etc. We make extensive use of two components in this paper. The constituent or *c-structure* of a sentence shows the temporal arrangement of words, phrases and clauses organized as a tree structure. The functional or

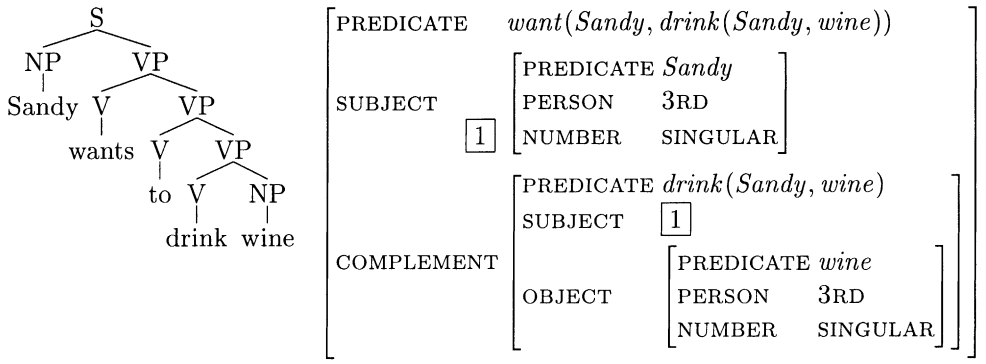

Fig. 1. The c- and the f-structure for the English sentence *Sandy wants to drink wine.*

f-structure of a sentence is an attribute-value structure that shows the grammatical function relationships between the phrases and clauses of a sentence, abstracting away from details of linear order. The particular grammatical function relationship involved (e.g., subject, object, etc.) is represented by the attribute name, and f-structures also encode the argument-adjunct distinction. Although it probably deserves to be a component in its own right, for simplicity we follow early work in LFG that encodes the predicate-argument structure of a phrase or sentence as the value of the PREDICATE attribute in an f-structure. Fig. 1 depicts the c- and f-structure of the English sentence *Sandy wants to drink wine*.

One of the reasons for adopting an attribute-value representation of f-structure in LFG is that such structures can describe the multiple functional roles that a single constituent can play in a single sentence. For example, in *Sandy wants to drink wine* the NP *Sandy* functions both as the subject of the verb *wants* and the verb *drink* (cf. *Sandy wants Sam to drink wine*). This is indicated by a re-entrancy in the f-structure, depicted by the shared index "1" in Fig. 1.

Similar re-entrancies are used to indicate the functional roles played in relative clauses and WH-questions, where a functional dependency may span an unbounded distance in the constituent structure. For example, in the question *Which bottle did Sandy want Sam to open?* The WH-phrase *which bottle* functions as the object of the verb *open* even though the two elements are discontinuous in c-structure terms. As explained in Section 3, lexical dependencies between governor–governee pairs play an important role in our probabilistic model, and their explicit representation in LFGs f-structure makes the construction of our probabilistic model much easier.

The f-structures also make explicit other important linguistic information. For example, the f-structure in Fig. 1 encodes person and number features on noun phrases (important for subject-verb and pronoun agreement); although not shown here, f-structures also encode verb tense and other semantically important information. Notice that the f-structure makes explicit dependencies that may be non-local or only indirectly marked in the c-structure, and represents these in a relatively language-independent way. This gives LFG the power to provide simple descriptions of phenomena such as crossed serial dependencies, which cannot be described using context-free grammars (Bresnan, Kaplan, Peters, & Zaenen, 1982; Shieber, 1985; Kaplan & Zaenen, 1995).

An account of language acquisition should explain how the properties that differentiate the language being learnt from other possible human languages are acquired. Since one of the goals of this research is to determine the extent to which language learning can be viewed as a statistical parameter estimation problem, the restrictions or constraints imposed on possible linguistic structures should be universal, i.e., satisfied by all possible human languages. Thus, the set of candidate linguistic structures (which we call Ω below) should include all structures possible in any human language. Unfortunately, such "universal grammars" are not yet available: indeed, there are still major conceptual issues to be resolved before such a universal grammar (for any linguistic theory) can be constructed. Because of the lack of any reasonable candidate for a universal grammar, our computational experiments to date have utilized grammars for specific languages such as English (Johnson et al., 1999; Johnson & Riezler, 2000) and German (Riezler et al., 2000). Thus, the statistical models developed in these experiments in effect learn how likely each grammatical linguistic structure of the particular language are: they are capable of interpreting and disambiguating phonological forms (strings of words in our case), but do not fully model language learning *per se*.

3. Probability distributions over linguistic structures

This section explains how we define a probability distribution over a set of possible linguistic structures Ω. In a model of language learning, Ω should be the set of all structures that could appear in any human language, but for models of parsing of a single language we take Ω to be the set of grammatical structures of that language. In either case, Ω is a countably infinite set, even if it is highly constrained by innate or languagespecific constraints.

The probability distribution over Ω is defined in terms of a finite vector of *features* $\boldsymbol{f} = (f_1, \ldots, f_m)$, where each f_j is a function mapping a linguistic structure $x \in \Omega$ to a real number $f_j(x)$. (The term 'feature' is used both in statistics and linguistics; we follow the standard usage in statistics here, and use the term 'attribute' to refer to components of attribute-value structures or node labels.) While the mathematics impose few constraints on what the features can be, we generally take $f_j(x)$ to be the number of times that a given construction appears in the linguistic structure $x \in \Omega$, which means that $f_j(x)$ is a non-negative integer.

The features can be *lexicalized*, i.e., they can make reference to a specific words or word classes, but they need not be. For an example of a non-lexicalized feature, let $f_1(x)$ be the number of times that a direct object immediately precedes its governing verb in x; this is presumably almost always zero for sentences of English, but is often non-zero for a head-final language like German. For an example of a lexicalized feature, let $f_2(x)$ be the number of times the verb *eat* appears with a direct object in the structure x; if this is close to number of times *eat* appears in x then presumably *eat* is a primarily transitive verb. The selection of features is presumably an empirical linguistic issue (just as the selection of constraints in an Optimality Theory grammar or of parameters in a Principles and Parameters model are empirical issues).

In our experiments with probabilistic LFGs we use a wide variety of features (see Johnson et al., 1999 for a more detailed description). Inspired by probabilistic context-free grammars, we introduced a feature f_A for each category A that can label a c-structure node, and define

$f_A(x)$ to be the number of times a node labeled A appears in the c-structure of x. Additionally, the probabilistic LFGs evaluated below used the following kinds of features, whose selection was guided by the principles proposed by Hobbs and Bear (1995). Adjunct and argument features indicate adjunct and argument attachment, respectively, and permit the model to capture a general argument attachment preference. In addition, there are specialized adjunct and argument features corresponding to each grammatical function used in LFG (e.g., SUBJECT, OBJECT, COMPLEMENT, ADJUNCT, etc.). There are features indicating both high and low attachment. Another feature indicates non-right branching non-terminal nodes. There is a feature for non-parallel coordinate structures (where parallelism is measured in constituent structure terms). Each f-structure attribute-atomic value pair which appears in any feature structure is also used as a feature. We also use a number of features identifying syntactic structures that seem particularly important in the particular corpora we used in our experiments, such as a feature identifying NPs that are dates (it seems that date interpretations of NPs are preferred if they are available).

Ideally we would like to include lexical features directly in our experiments to capture the dependencies between governors and the heads of the phrases that they govern, but we did not have enough training data to estimate these directly in our experiments. However, probabilistic models of such dependencies can be constructed by other means, and we can include information from such "auxiliary" models in our model as follows (Johnson & Riezler, 2000; Riezler et al., 2000). Suppose we have an auxiliary model R which assigns a positive numerical preference score $R(x)$ to each $x \in \Omega$ (R might define a probability distribution over Ω, but need not). Then we define a new feature $f_R(x) = \log R(x)$, and treat it otherwise just as another feature in our model. In effect, the preference information from the auxiliary model R is treated as another source of information that will be taken into account in the model we construct. This provides a general mechanism whereby a range of complex preferences (possibly including innate ones) can be included in a statistical model, which generalizes the "reference distribution" approach described in Jelinek (1997).

We now explain how the probability of a particular linguistic structure x is defined in terms of its feature values $\boldsymbol{f}(x) = (f_1(x), \ldots, f_m(x))$. While there are many ways in which this can be done, we use the class of *log-linear models* in our research (Abney, 1997). We justify our choice of log-linear models after we have explained how they are defined.

Given a set of linguistic structures Ω and a feature vector $(f_1, \ldots, f_m)$, a log-linear model is defined by a *parameter vector* $\theta = (\theta_1, \ldots, \theta_m)$, where each θ_j is a real number. Informally, θ_j is the "weight" assigned to the corresponding feature f_j. If θ_j is positive then higher values of $f_j(x)$ increase the probability of x, and if θ_j is negative then higher values of $f_j(x)$ decrease the probability of x (assuming that the values of $f_{j'}(x)$, $j' = j$ stay the same).

Mathematically, the probability $P_\theta(X = x)$ of x given the parameter vector θ is defined as follows. We define the *weight* $V_\theta(x)$ of x as the exponential of a linear combination of the feature values of x, weighted according to the parameter vector. (Thus, the logarithm of $V_\theta(x)$ is a linear combination of the feature values, hence the name log-linear model.)

$$V_\theta(x) = \exp\left(\sum_{j=1}^{m} \theta_j f_j(x)\right)$$

A probability distribution over linguistic structures Ω must satisfy the normalization constraint that the sum of probability of the structures in Ω is 1, i.e., $\sum_{x\in\Omega} P_\theta(X = x) = 1$. We cannot set $P_\theta(X = x) = V_\theta(x)$ because in general V_θ does not satisfy the normalization constraint. However, we can make $P_\theta(X = x)$ proportional to $V_\theta(x)$ by dividing the latter by a normalization factor known as the *partition function* Z_θ (the name comes from statistical physics, which was the first major application of log-linear models).

$$Z_\theta = \sum_{x\in\Omega} V_\theta(x) \tag{1}$$

$$P_\theta(X = x) = \frac{V_\theta(x)}{Z_\theta} \tag{2}$$

Unlike probabilistic context-free grammars and related models, log-linear models permit essentially arbitrary dependencies between features, which makes them ideal for defining probability distributions over linguistically realistic structures (Abney, 1997). Additionally, there are information-theoretic reasons for preferring log-linear models over other model classes. The class of log-linear models is also the class of *maximum entropy* models; roughly speaking, these are the models which contain the minimum additional information over and above the information contained in the training data (see Jelinek, 1997 for a textbook introduction). Virtually all of the well known probabilistic models of language are subclasses of the class of log-linear models (e.g., probabilistic context-free grammars, hidden Markov models, etc.). Finally, even though one might suspect that the restriction to *linear* combinations of the feature values is unduly restrictive, because no restrictions are placed on the features themselves, we can define a feature which is a non-linear combination of other features, so the class of log-linear models is much less restrictive than it may first seem.

4. Learning grammars

The previous section described how we define a log-linear probability distribution over linguistic structures Ω. We now turn to the problem of determining the parameter vector θ from some observational data D. In our experiments we use *maximum likelihood* estimators (but see the discussion of regularization in Section 5). A maximum likelihood estimator selects a parameter vector θ which makes the data D as likely as possible. Under very general conditions, maximum likelihood estimation is *unbiased* (the expected value of the parameter estimate is its true value), *consistent* (as the size of the data grows, the estimated parameters converge on the true value) and *asymptotically efficient* (there is no other estimation procedure whose parameter estimates have uniformly lower variance). Further, given the independence assumptions below the maximum likelihood estimator for a log-linear model selects the closest model to the training data distribution in terms of Kullback–Leibler divergence (an information-theoretic measure of the distance between two distributions).

More formally, suppose that D consists of a sequence of *fully observed* parses $D = (x_1, \ldots, x_n)$, $x_i \in \Omega$. ("Fully observed" means that the learner has access to the complete linguistic structures; we consider the problem of learning from phonological forms alone

Fig. 2. Maximum likelihood estimation from fully observed (parsed) data.

below.) We make the standard statistical assumptions that each observation x_i is independent of the other observations $x_{i'}$, $i' \neq i$, and that each x_i is identically distributed according to P_θ for some unknown θ (these assumptions are undoubtedly incorrect, but we hope that they are approximately true). Given these assumptions, the likelihood L_D of the data D and the corresponding maximum likelihood estimate $\hat{\theta}$ of θ are

$$L_D(\theta) = \prod_{i=1}^{n} P_\theta(X = x_i), \qquad \hat{\theta} = \operatorname*{argmax}_\theta L_D(\theta) \tag{3}$$

Fig. 2 graphically depicts this maximum likelihood estimation. Informally, maximum likelihood estimation adjusts θ to make the weight $V_\theta(x_i)$ of each training datum as large as possible relative to the partition function Z_θ (the sum of the weights of all linguistic structures Ω).

It is straightforward to show that L_D has a unique maximum value, and at this maximum the expected value $E_{\hat{\theta}}(f_j)$ of each feature under the distribution $P_{\hat{\theta}}$ is equal to its expected value under the "empirical distribution" of the training data D, i.e.

$$E_{\hat{\theta}}(f_j) = \frac{1}{n}\sum_{i=1}^{n} f_j(x_i), \quad j = 1, \ldots, m$$

Thus, maximum likelihood estimation selects a parameter vector $\hat{\theta}$ so that the expected value of each feature under the estimated distribution $P_{\hat{\theta}}$ is the same as the average value of that feature in the training data, which intuitively seems to be a reasonable thing to do.

Now we turn to the case where the training data is *partially hidden* and consists of phonological forms alone, i.e., $D' = (w_1, \ldots, w_n)$, where each w_i is a phonological form (here taken to be a string of words). In this situation the training data does not uniquely identify the linguistic structure corresponding to each phonological form w_i; all we know is that it lies somewhere in the set $\Omega(w_i) = \{x : W(x) = w_i\}$ of linguistic structures whose phonological form is w_i. Making the same independence assumptions as before, the likelihood $L'_{D'}$ of the data D' is now a product of the marginal probability of each w_i, where the marginal probability of w is the sum of the probability of each $x \in \Omega(w)$.

$$P_\theta(W = w) = \sum_{x \in \Omega(w)} P_\theta(X = x),$$

$$L'_{D'}(\theta) = \prod_{i=1}^{n} P_\theta(W = w_i), \qquad \hat{\theta} = \operatorname*{argmax}_\theta L'_{D'}(\theta) \tag{4}$$

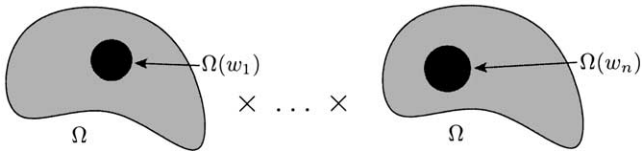

Fig. 3. Maximum likelihood estimation from partially visible (phonological form) data.

Fig. 3 graphically depicts the quantity being maximized during estimation from phonological forms alone. Notice that the maximum likelihood estimator selects the θ that places maximum weight on the $\Omega(w_i)$ as compared to the whole of Ω.

There is a standard technique known as the "expectation-maximization (EM)" algorithm which reduces the optimization required in maximum likelihood estimation from partially hidden data to a series of optimizations of the kind involved in maximum likelihood estimation from fully visible data (Dempster, Laird, & Rubin, 1977). The technique requires an initial guess $\theta^{(0)}$ of the parameter vector as well as the partially observed data D', and it produces a sequence of estimates $\theta^{(1)}, \theta^{(2)}, \ldots$. This sequence has the property that each additional estimate typically increases, and provably does not decrease, the likelihood of the partially observed data, i.e., $L'_{D'}(\theta^{(k+1)}) \geq L'_{D'}(\theta^{(k)})$.

Informally, the technique involves treating each partial observation w_i as a set of fully observed data consisting of each $x \in \Omega(w_i)$, with each full observation x weighted according to $P_{\theta^{(k)}}(x)$, where $\theta^{(k)}$ is the estimate of θ at the kth iteration. Thus, EM "pays most attention to" the $x \in \Omega(w_i)$ that its current estimate of θ assigns the highest probability to.

Unlike the fully visible case, there is no guarantee that the likelihood function for partially hidden data has only a single local maximum, and the EM algorithm can get "trapped" in such local maxima. Indeed, there is no guarantee that estimation is possible at all: the parameter vector θ may simply be *non-identifiable* from the kind of data available. For example, it is logically possible that universal grammar permits two different languages with exactly the same marginal distribution over phonological forms, even though the two languages associate each phonological form with different semantic interpretations.

5. Pseudo-likelihood estimation

The previous section introduced maximum likelihood estimation of θ for both fully visible and partially hidden data. Unfortunately, it seems that maximizing the likelihood (3) is computationally infeasible even for fully visible data (and since the EM technique reduces the partially hidden data case to the fully visible data case, it too is infeasible). The standard algorithms for maximizing this likelihood are iterative, and require the calculation of the expected value of each feature $E_\theta(f_j)$ for a variety of different parameter vectors θ (see Berger, Della Pietra, & Della Pietra, 1996; Jelinek, 1997 for an introduction to these algorithms). Informally, the cause of the computational infeasibility is that maximum likelihood estimation requires us to select the parameter vector θ that maximizes the weight $V_\theta(x_i)$ on the observed datum x_i relative to the sum Z_θ of the weights on *all* possible linguistic structures $x \in \Omega$ (see

Fig. 2 and Eqs. (1)–(3)). Because Ω is infinite, we cannot calculate the partition function Z_θ or the feature expectations $E_\theta(f_j)$ by explicitly enumerating Ω. Indeed, even calculating the probability $P_\theta(X = x)$ of a single linguistic structure x seems infeasible, since it too crucially involves Z_θ (see Eq. (2)).

If Ω and the feature vector $\boldsymbol{f}$ have a suitably simple structure, then it may be the case that Z_θ and $E_\theta(f_j)$ can be calculated in an analytically and computationally tractable way. For example, if Ω is the set of trees generated by a context-free grammar and the feature f_i maps an $x \in \Omega$ to the number of times the ith production is used in a derivation of x, then Z_θ and $E_\theta(f_j)$ can be calculated without an explicit enumeration of Ω (Abney, McAllester, & Pereira, 1999; Chi, 1999). However, this calculation depends crucially on the context-free or Markovian independence properties of Probabilistic Context-Free Grammars. It seems that such context-free systems cannot describe the true set Ω of possible linguistic structures (Shieber, 1985), yet these context-free properties are what makes the computation of Z_θ and $E_\theta(f_j)$ feasible. Indeed, precisely because the LFGs used in this research are capable of capturing the non-local, context-sensitive dependencies of natural language, the methods that can be used to calculate Z_θ and $E_\theta(f_j)$ for PCFGs do not extend to LFGs.

Never the less, we believe that there may be techniques for calculating or approximating Z_θ for LFGs that avoid explicit enumeration. Abney (1997) points out that $E_\theta(f_j)$ can be approximated using Monte Carlo sampling techniques that do not enumerate all of Ω. While this is in principle correct, a "back of the envelope" calculation suggests that the particular Hastings Metropolis sampling scheme that Abney proposes is computationally impractical for all but small grammars (see Johnson et al., 1999 for further discussion).

However, note that the full joint distribution over phonological forms and their parses is not actually required for natural language processing tasks. For example, as explained above, comprehension and parsing only requires the *conditional* distribution $P(X|W)$ of linguistic structures given their phonological forms. Crucially, estimating these conditional distributions is often computationally feasible, even though estimation of the joint distribution is infeasible.

Consider the case where the data is fully observed: D consists of parses $D = (x_1, \ldots, x_n)$, $x_i \in \Omega$ as above. Each parse is associated with a phonological form $w_i = W(x_i)$. Making the same independence assumptions as before, the conditional likelihood or *pseudo-likelihood* PL_D of the data D and the corresponding maximum likelihood estimate $\hat{\theta}$ of θ are

$$\mathrm{PL}_D(\theta) = \prod_{i=1}^{n} P_\theta(X = x_i | W = w_i), \qquad \hat{\theta} = \operatorname*{argmax}_\theta \mathrm{PL}_D(\theta) \tag{5}$$

where

$$P_\theta(X = x | W = w) = \frac{V_\theta(x)}{Z_\theta(w)}, \qquad Z_\theta(w) = \sum_{x \in \Omega(w)} V_\theta(x)$$

Whereas the likelihood L_D is a product of (unconditional) probabilities (3), the PL_D is a product of conditional probabilities (5). Ultimately, pseudo-likelihood differs from likelihood in that pseudo-likelihood only involves $Z_\theta(w)$ in place of the infeasible Z_θ in the likelihood. It is straightforward to show that at the maximum of (5), the sum of the conditional expectations of

each feature must be same as the sum of their empirical values, where $E_\theta(f|W)$ is the expectation of f with respect to the conditional distribution $P_\theta(X|W)$

$$\sum_{i=1}^{n} E_{\hat{\theta}}(f_j|W = w_i) = \sum_{i=1}^{n} f_j(x_i), \quad j = 1, \ldots, m.$$

Moving to pseudo-likelihood makes a crucial difference in the kinds of expectations that must be computed in the standard algorithms for maximizing θ; they now involve the generally feasible conditional expectations $E_\theta(f_j|W)$ rather than the infeasible unconditional expectations $E_\theta(f_j)$.

It turns out that this idea of estimating a conditional distribution (rather than the joint) has been independently discovered at least twice. Besag (1975), who coined the name 'pseudo-likelihood,' uses it in a computational vision setting in which one part of an image serves as the conditioning environment for another part of the image (here, the phonological form corresponds to one part of the linguistic structure, and everything else in the structure corresponds to the other part). Berger et al. (1996) and Jelinek (1997) both describe optimizations in their algorithms which replace joint probabilities with conditional probabilities in exactly the manner described here (but they do not acknowledge that this means they are estimating a conditional rather than a joint distribution).

Fig. 4 graphically depicts maximum pseudo-likelihood estimation (compare Fig. 3). Informally, maximum pseudo-likelihood estimation adjusts θ to make the weight $V_\theta(x_i)$ of each training datum as large as possible relative to $Z_\theta(w_i)$, i.e., the sum of the weights of all parses $\Omega(w_i)$ of the phonological form w_i. As remarked earlier, $\Omega(w_i)$ is finite and of manageable size for LFGs, so $Z_\theta(w_i)$ and the conditional expectations required for maximizing the pseudo-likelihood can be calculated using explicit enumeration of $\Omega(w_i)$.

While pseudo-likelihood estimation is consistent *for the conditional distribution*, it is not hard to see that maximizing PL_D will not always correctly estimate the joint $P_\theta(X)$ (Chi, 1998). Suppose there is a feature f_j which depends solely on the phonological form $W(x)$ of a linguistic structure x, i.e., $f_j(x') = f_j(x)$ for all $x \in \Omega$ and $x' \in \Omega(W(x))$; we call such features *pseudo-constant*. (For an example of a pseudo-constant feature, let $f_j(x)$ be the number of times the word *eat* occurs in x.) If f_j is pseudo-constant, then it is easy to show that the pseudo-likelihood does not depend on the value of the parameter θ_j associated with f_j, so maximum pseudo-likelihood estimation provides no basis for choosing a value for θ_j. In fact, in this case *any* value of θ_j gives the *same* conditional distribution $P_\theta(X|W)$, so θ_j is irrelevant to the problem of choosing good parses.

Informally, the relationship between maximum likelihood and pseudo-likelihood estimation is the same as the relationship between the joint $P(X, W)$ and the conditional $P(X|W)$, which

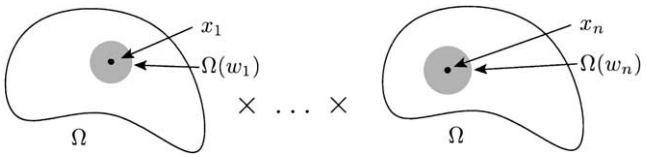

Fig. 4. Maximum pseudo-likelihood estimation from fully observed data.

are related by the marginal $P(W)$:

$$P(X, W) = P(X|W)P(W).$$

The parameter vectors estimated by maximum likelihood estimation model the joint; they describe both the conditional distribution of parses given phonological forms as well as the marginal distribution of phonological forms $P(W)$, while pseudo-likelihood estimation focuses on the conditional $P(X|W)$ and ignores the marginal.

Interestingly, from a cognitive modularity perspective, the conditional and the marginal distributions seem to correspond to two different kinds of information. As noted above, the conditional distribution $P(X|W)$ is precisely the information required for disambiguation in sentence comprehension, which seems to be purely linguistic knowledge. The marginal distribution $P(W)$, on the other hand, describes the distribution of phonological forms, which seems to involve world knowledge and contextual information at least as much as it involves linguistic knowledge. Thus, pseudo-likelihood estimation may be more compatible with a modular view of language, since it seems to focus on more purely linguistic knowledge than does maximum likelihood estimation.

We now briefly describe some of the more practical details of pseudo-likelihood estimation. Despite the assurance of consistency, pseudo-likelihood estimation is prone to over fitting when a large number of features is matched against a modest-sized corpus of training data. One particularly troublesome manifestation of over fitting results from the existence of features which, relative to the training data, we call "pseudo-maximal." A feature f is *pseudo-maximal* for a phonological form w if and only if for all $x' \in \Omega(w)$ $f(x) \geq f(x')$, where x is any correct parse of w, i.e., the feature's value on every correct parse x of w is greater than or equal to its value on any other parse of w. Pseudo-minimal features are defined similarly. It is easy to see that if f_i is pseudo-maximal on *each sentence* of the training corpus then the parameter assignment $\theta_j = \infty$ maximizes the corpus pseudo-likelihood. (Similarly, the assignment $\theta_j = -\infty$ maximizes pseudo-likelihood if f_j is pseudo-minimal over the training corpus.) Such infinite parameter values indicate that the model treats pseudo-maximal features categorically, i.e., any parse with a non-maximal feature value is assigned a zero conditional probability.

Of course, a feature which is pseudo-maximal over a finite training corpus is not necessarily pseudo-maximal for all phonological forms in those language. This is an instance of over fitting, and it can be addressed, as is customary, by adding to the objective function a *regularization* term that promotes small values of θ. In Johnson et al. (1999), we added a quadratic to the log pseudo-likelihood, which corresponds to multiplying the pseudo-likelihood itself by a normal distribution. Specifically, we multiplied the pseudo-likelihood by a zero-mean normal in θ with diagonal covariance and with standard deviation σ_j for Ω equal to seven times the maximum value of f_j found in any parse in the training data. Thus, instead of choosing $\hat{\theta}$ to maximize the pseudo-likelihood (5), in the experiments reported in Johnson et al. (1999) and Johnson and Riezler (2000) we actually selected $\hat{\theta}$ to maximize:

$$\log \mathrm{PL}_D(\hat{\theta}) - \sum_{j=1}^{m} \frac{\theta_j^2}{2\sigma_j^2} \tag{6}$$

Interestingly, this way of regularizing has a Bayesian interpretation. In Bayesian estimation one seeks a parameter vector θ that maximizes the *posterior probability* $P(\theta|D)$ of the parameter vector θ given the training data D. According to Bayes theorem, this can be done by maximizing the product of the *prior probability* $P(\theta)$ of the parameter vector and the likelihood $P(D|\theta)$ of the data given the parameter vector. If one sets the prior probability $P(\theta)$ to be proportional to $\exp(-\sum_{j=1}^{m}\theta_j^2/2\sigma_j^2)$ and makes the same independence assumptions concerning the data as above, then it is possible to show that the Bayesian estimate for θ is precisely the θ that maximizes (6).

In these experiments, the set of possible linguistic structures Ω was defined by a hand-written LFG for English, which was specifically designed at Xerox Parc to generate the sentences in two corpora of business appointment dialogs and "Homecenter" printer/copier documentation, consisting of 500 and 1,000 parsed sentences, respectively. Even though the grammar included all standard linguistic constraints, the sentences in the corpora were often highly ambiguous, with an average of eight parses per sentence. The training data consisted of the correct parse for each sentence (which was identified manually) together with the set of all alternative (i.e., incorrect) parses of the sentence generated by the grammar. Using a cross-validation framework, we showed that a model trained by maximum pseudo-likelihood correctly disambiguated approximately 58% of the ambiguous test sentences, whereas a model that treated each parse as equally likely would correctly disambiguate only 25% of the ambiguous test sentences.

We now turn to the more realistic situation (in terms of language acquisition) where the training data consists of phonological forms alone. Whereas maximum likelihood estimation from partially visible data is conceptually straightforward—one adjusts θ to maximize the likelihood of the phonological forms that constitute the training data D—it turns out that a similar approach based on pseudo-likelihood fails. Specifically, conditioning the marginal $P(W = w_i)$ in the likelihood (4) on the phonological form results in a constant-valued likelihood that does not vary with θ or D, so estimation fails.

Intuitively, the problem is that we are trying to maximize the sum of the weights $V_\theta(x)$ placed on the $x \in \Omega(w_i)$ relative to the sum of the weights of exactly the same $\Omega(w_i)$, set as depicted in Fig. 5. Standard maximum likelihood estimation from partially visible data (as performed by the EM algorithm) maximizes the sum of the weights placed on $\Omega(w_i)$ relative to the sum Z_θ of the weights placed on all possible linguistic structures Ω.

We noted earlier that maximum likelihood estimation is infeasible because the partition function Z_θ and the expectations $E_\theta(f_j)$ involve summing over all possible linguistic structures Ω. In Riezler et al. (2000) we developed a method for maximum likelihood estimation from partially visible data that exploits a conditional approximation to Z_θ and $E_\theta(f_j)$ in which we

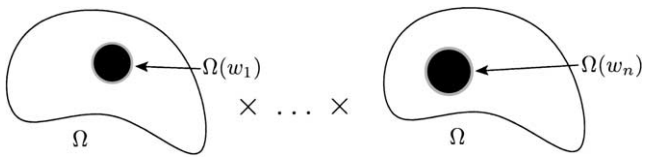

Fig. 5. A straightforward application of maximum pseudo-likelihood estimation from partially visible (phonological form) data fails.

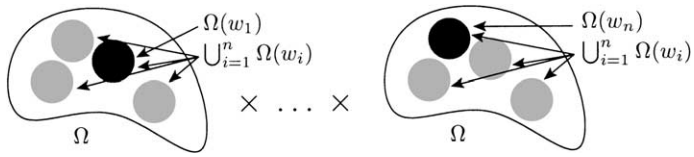

Fig. 6. A conditional approach to estimation from partially visible (phonological form) data.

replace the summation over Ω with a summation over the finite set $\Omega(D')$ consisting of all possible parses of all phonological forms that constitute the training data $D' = (w_1, \ldots, w_n)$. More precisely, the likelihood function $L''_{D'}$, maximized in these experiments is:

$$\Omega(D') = \bigcup_{i=1}^{n} \Omega(w_i), \qquad Z_\theta(D') = \sum_{x \in \Omega(D')} V_\theta(x),$$

$$P_\theta(W = w | W \in D') = \frac{Z_\theta(w_i)}{Z_\theta(D')}, \qquad L''_{D'}(\theta) = \prod_{i=1}^{n} P_\theta(W = w_i | W \in D')$$

Fig. 6 depicts the likelihood function that this conditional approach maximizes. Unlike standard maximum likelihood estimation, computation of the conditional partition function $Z_\theta(D')$ and the corresponding expectations is feasible. The conditional approach can be viewed as a version of pseudo-likelihood in the following way. Recall the key idea behind pseudo-likelihood: namely, that one can define a likelihood function by conditioning one part of the structure on another part of that structure. In pseudo-likelihood estimation from fully visible (parsed) data we take each sentence to be an observation and condition each linguistic structure on its phonological form. In this conditional approach, we take the entire dataset D' to be an observation, and condition each phonological form on the fact that it occurred in D'.

Since the above described estimation procedure does not require manually annotated data for training but merely data consisting of phonological forms alone, large sets of training data can easily be provided. In our experiments we parsed a large corpus of newspaper text with a German LFG grammar (developed in the ParGram project at the University of Stuttgart), and extracted all parses for sentences which were assigned at most 20 parses by the grammar. This resulted in a training corpus of approximately 36,000 sentences and 250,000 parses. An evaluation of disambiguation performance on LFG-parsed newspaper sentences with on average 25 parses per sentence showed the following results: the task of matching full c-/f-structure pairs to the manually selected pair could be performed correctly in over 60% of the test cases; a disambiguation of the predicate-argument structures of the parses of the test sentences (which is sufficient for many application purposes) could be performed correctly in over 90% of test cases.

6. Conclusion and further directions

Because log-linear models make no assumptions about relationships between features, they provide a general framework for defining probability distributions over linguistic structures from virtually any linguistic theory (Abney, 1997). Maximum likelihood estimation is an

optimal method for estimating the parameter vectors for such models from data, but precisely because log-linear models are so general, maximum likelihood estimation is typically computationally infeasible because it requires us to calculate expectations over all possible linguistic structures. This lead us to develop techniques based on pseudo-likelihood (Besag, 1975) for estimating parameter vectors from fully visible (parsed) data (Johnson et al., 1999; Johnson & Riezler, 2000) and partially visible (phonological form) data (Riezler et al., 2000).

This work is still in its infancy, and many interesting avenues remain to be explored. We believe there is interesting empirical linguistic research to be done in investigating the trade-off between the "hard" grammatical constraints incorporated in the grammar that determines Ω and the "soft" preferences that can be encoded using features f_j in the statistical model. The grammars we used in our experiments were not written with our statistical models in mind, and we might obtain a more robust system with broader coverage by removing some of the grammatical constraints from the grammar and re-expressing them as features in the statistical model.

Turning to more mathematical issues, it would be valuable to investigate other ways for estimating the partition function and the expectations required for maximum likelihood estimation from both parsed and phonological form data. Techniques for approximating these quantities have been developed in other fields (e.g., mean field approximations), and it may be possible to apply them in computational linguistics as well (Saul & Jordan, 1999).

A problem left unaddressed in our applications is efficient searching for most probable parses. This question becomes crucial if higher coverage is desired and traded in for more superficial parses and for higher ambiguity. Clearly, for such cases it is desirable to adapt techniques such as Viterbi's algorithm (Viterbi, 1967) to searching efficiently for most probable parses in probabilistic LFG grammars. Here a closer look at generalized dynamic programming techniques as developed for graphical models (Frey, 1998) seems promising.

Finally, we believe that there may be other ways of applying pseudo-likelihood to language learning besides the ways described in this paper. The pseudo-likelihood estimation approach from visible (parsed) data it seems highly unrealistic in one respect: such a learner *learns nothing from unambiguous sentence* in its training data, even though such sentences are intuitively most informative of all. This is because the pseudo-likelihood we used conditioned on phonological form, i.e., $P(X|W)$. Suppose instead we adopt a "generation-oriented" pseudo-likelihood, where we condition on the semantic interpretation $S(x)$ of each linguistic structure x, so the likelihood is the product of terms $P(X = x_i | S = s_i)$. Such a learner would learn from each sentence in its training data whose semantic interpretation can be expressed in more than one way universally.

Note

1. That is, a class of probabilistic languages may be statistically learnable even though its categorical counterpart is not. Informally this is because the statistical learning framework makes stronger assumptions about the training data (i.e., that it is distributed according to some probabilistic grammar from the class) and accepts a weaker criterion for successful learning (convergence in probability).

References

Abney, S. (1997). Stochastic attribute-value grammars. *Computational Linguistics*, *23*(4), 597–617.

Abney, S., McAllester, D., & Pereira, F. (1999). Relating probabilistic grammars and automata. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics* (pp. 542–549). San Francisco: Morgan Kaufmann.

Berger, A. L., Della Pietra, V. J., & Della Pietra, S. A. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, *22*(1), 39–71.

Besag, J. (1975). Statistical analysis of non-lattice data. *The Statistician*, *22*, 179–195.

Bresnan, J. (1982). *The mental representation of grammatical relations*. Cambridge, MA: MIT Press.

Bresnan, J., Kaplan, R. M., Peters, S., & Zaenen, A. (1982). Cross-serial dependencies in Dutch. *Linguistic Inquiry*, *13*, 613–635.

Chi, Z. (1998). *Probability models for complex systems*. Unpublished doctoral dissertation, Brown University.

Chi, Z. (1999). Statistical properties of probabilistic context-free grammars. *Computational Linguistics*, *25*(1), 131–160.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data *via* the EM algorithm. *Journal of the Royal Statistical Society, Series B*, *39*, 1–38.

Frey, B. J. (1998). *Graphical models for machine learning and digital communication*. Cambridge, MA: MIT Press.

Gold, E. (1967). Language identification in the limit. *Information and Control*, *10*, 447–474.

Hobbs, J. R., & Bear, J. (1995). Two principles of parse preference. In A. Zampolli, N. Calzolari, & M. Palmer (Eds.), *Linguistica computazionale: Current issues in computational linguistics in honor of Don Walker* (pp. 503–512). Dordrecht: Kluwer Academic Publishers.

Horning, J. J. (1969). *A study of grammatical inference*. Unpublished doctoral dissertation, Stanford.

Jelinek, F. (1997). *Statistical methods for speech recognition*. Cambridge, MA: MIT Press.

Johnson, M., Geman, S., Canon, S., Chi, Z., & Riezler, S. (1999). Estimators for stochastic "unification-based" grammars. In *Proceedings of the 37th Annual Conference of the Association for Computational Linguistics* (pp. 535–541). San Francisco: Morgan Kaufmann.

Johnson, M., & Riezler, S. (2000). Exploiting auxiliary distributions in stochastic unification-based grammars. In *Proceedings of the North American Chapter of the Association for Computational Linguistics* (pp. 154–161). San Francisco: Morgan Kaufmann.

Kaplan, R. M. (1995). The formal architecture of LFG. In M. Dalrymple, R. M. Kaplan, J. T. Maxwell III, & A. Zaenen (Eds.), *Formal issues in lexical-functional grammar* (pp. 7–28). CSLI Publications.

Kaplan, R. M., & Zaenen, A. (1995). Long-distance dependencies, constituent structure and functional uncertainty. In M. Dalrymple, R. M. Kaplan, J. T. Maxwell III, & A. Zaenen (Eds.), *Formal issues in lexical-functional grammar* (pp. 137–165). CSLI Publications.

Maxwell, J. T., III, & Kaplan, R. M. (1993). The interface between phrasal and functional constraints. *Computational Linguistics*, *19*(4), 571–590.

Riezler, S., Prescher, D., Kuhn, J., & Johnson, M. (2000). Lexicalized stochastic modeling of constraint-based grammars using log-linear measures and EM training. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*.

Saul, L., & Jordan, M. (1999). A mean field learning algorithm for unsupervised neural networks. In M. Jordan (Ed.), *Learning in graphical models* (pp. 541–554). Cambridge, MA: MIT Press.

Shieber, S. M. (1985). Evidence against the context-freeness of natural language. *Linguistics and Philosophy*, *8*(3), 333–344.

Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, *IT-13*, 260–269.