8 Unsupervised Learning and Grammar Induction

ALEXANDER CLARK
AND SHALOM LAPPIN

In this chapter we consider unsupervised learning from two perspectives. First, we briefly look at its advantages and disadvantages as an engineering technique applied to large corpora in natural language processing. While supervised learning generally achieves greater accuracy with less data, unsupervised learning offers significant savings in the intensive labor required for annotating text. Second, we discuss the possible relevance of unsupervised learning to debates on the cognitive basis of human language acquisition. In this context we explore the implications of recent work on grammar induction for poverty of stimulus arguments that purport to motivate a strong bias model of language learning, commonly formulated as a theory of universal grammar (UG). We examine the second issue both as a problem in computational learning theory, and with reference to empirical work on unsupervised machine learning (ML) of syntactic structure. We compare two models of learning theory and the place of unsupervised learning within each of them. Looking at recent work on part-of-speech tagging and the recognition of syntactic structure, we see how far unsupervised ML methods have come in acquiring different kinds of grammatical knowledge from raw text.

1 Overview

1.1 Machine learning in natural language processing and computational linguistics

The machine learning methods presented in this handbook have been applied to a wide variety of problems in natural language processing. These range from speech recognition (Chapter 12) through morphological analysis (Chapter 14) and syntactic parsing (Chapter 4), to the complex text and discourse understanding applications dealt with in Part IV. ML has produced increasingly successful systems for handling a large domain of natural language engineering tasks. When evaluating different types of ML there are a variety of technological issues that arise, some of which we will consider in the context of the distinction between supervised and unsupervised learning procedures.

From an engineering perspective, the main issue to be addressed when comparing the relative merits of supervised vs. unsupervised learning, for a particular task, is the degree of accuracy that each method achieves in proportion to the cost of resources that it requires. As we will see, characterizing an optimal balance between accuracy and cost is not always straightforward. It is necessary to consider a variety of factors in calculating both of the values that determine this balance.

It is also interesting to consider if ML has implications for some of the scientific questions that animate linguistics and cognitive science. Specifically, it is worth asking if the success of ML methods in solving language engineering problems illuminates the sorts of learning processes that humans could, in principle, employ in acquiring knowledge of their language. Clearly the fact that an ML procedure is able to efficiently acquire important elements of human grammatical knowledge from corpora does not, in itself, show that human learning operates according to this procedure. However, to the extent that grammar induction through domain-general learning methods succeeds on the basis of evidence of the kind available to children, we achieve insight into the computational credibility of such methods as models of language acquisition.

1.2 Grammar induction as a machine learning problem

A machine learning system implements a learning algorithm whose output is a function from a domain of input samples to a range of output values. We divide a corpus of examples into a training and a test set. The learning algorithm is specified in conjunction with a model of the phenomenon to be learned. This model defines the space of possible hypotheses that the algorithm can generate from the input data. When the values of the model's parameters are determined through training of the algorithm on the test set, an element of the hypothesis space is selected. In the case of grammar induction the algorithm learns from the training data to construct a parser that assigns descriptions of syntactic structure to input strings from the test data. It provides a learning procedure for acquiring a grammar that parses new strings in the corpus.

If we have a gold standard of correct parses in a corpus, then it is possible to compute the percentage of correct parses that the algorithm produces when tested on an unseen subpart of this corpus. A more common procedure for scoring an ML algorithm on a test set is to evaluate its performance for *recall* and *precision*.[1]

The recall of a parsing algorithm $\mathcal{A}$ is the percentage of brackets of the test set that it correctly identifies, where these brackets specify the constituent tree structure of each sentence in the set. $\mathcal{A}$'s *precision* is the percentage of the brackets that it returns which correspond to those in the gold standard. A unified score for $\mathcal{A}$, known as an *F-score*, can be computed as an average of its recall and its precision.

The choice of parameters and their possible values defines a bias for the language model by imposing prior constraints on the set of learnable hypotheses. All learning requires some sort of bias to restrict the set of possible hypotheses for the phenomenon to be learned. This bias can express strong assumptions about the nature of the domain of learning. Alternatively, it can define comparatively weak domain-specific constraints, with learning driven primarily by domain-general procedures and conditions.

One way of formalizing a learning bias is as a *prior* probability distribution on the elements of the hypothesis space that favors some hypotheses as more likely than others. The paradigm of Bayesian learning in cognitive science implements this approach.[2] The simplicity and compactness measure that Perfors et al. (2006) use is an example of a very general prior. We can describe this measure as follows.

Let D be data, and H a hypothesis. *Maximum likelihood* chooses the H which makes the D most likely (the maximum probability value of D given H):

(1) $\arg\max_H(P(D|H))$

Posterior probability is proportional to the prior probability times the likelihood.

(2) $P(H|D) \propto P(H)P(D|H)$

The maximum a posteriori approach chooses the H which maximizes the posterior probability:

(3) $\arg\max_H(P(H)P(D|H))$

The bias of the model is explicitly represented in the prior $P(H)$. Perfors et al. (2006) define this prior to give higher values to grammars whose rule sets are of smaller cardinality, and whose rules are formulated with fewer non-terminal symbols.

1.3 Supervised learning

When the samples of the training set are annotated with the classifications and structures that the learning algorithm is intended to produce as output for the test set, then learning is described as *supervised*. Grammar induction that is supervised involves training an ML system on a corpus annotated with the parse structures that correspond to a gold standard of correct parse descriptions. The learning algorithm infers a function for assigning appropriate parse output to input sentences on the basis of a training set of sentence argument-parse value pairs.

As an example of supervised grammar induction, consider the learning of a *probabilistic context-free grammar* (PCFG).[3] Such a grammar conditions the probability of a child sequence on that of the parent non-terminal. Each of its context-free grammar (CFG) rules $N \rightarrow X_1 \ldots X_n$ expands a non-terminal N into a sequence $X_1 \ldots X_n$ of non-terminal and terminal symbols, and the rule is assigned a probability value. The grammar provides conditional probabilities of the form $P(X_1 \dots X_n \,|\, N)$ for each non-terminal N and sequence $X_1 \dots X_n$ of items from the set of non-terminals and the vocabulary of terminals in the grammar. It also specifies a probability distribution over the label of the root of the tree $P_s(N)$. For a PCFG G, the conditional probabilities $P(X_1 \dots X_n \,|\, N)$ correspond to probabilistic parameters that govern the expansion of a node in a parse-tree according to a corresponding context-free rule $N \rightarrow X_1 \dots X_n$ in G.

The probabilistic parameter values of a PCFG can be learned from a parse annotated training corpus by computing the frequency of CFG rules instantiated in the corpus, in accordance with a *maximum likelihood estimation* (MLE) condition.

(4) $$\frac{c(A \rightarrow \beta_1 \dots \beta_k)}{c(A \rightarrow \gamma)}$$

where c(R) = the number of occurrences of a rule R in the annotated corpus.

In practice, MLE does not perform as well as more sophisticated estimation methods based on distribution-free techniques (see Collins 2004).

It is possible to significantly improve the performance of a PCFG by adding additional bias to the language model that it defines. Collins (1999) constructs a *lexicalized probabilistic context-free grammar* (LPCFG) in which the probabilities of the CFG rules are conditioned on lexical heads of the phrases that non-terminal symbols represent. In Collins' LPCFG non-terminals are replaced by non-terminal/head pairs. The probability distributions of the model are of the form $P_s(N/h)$ and $P(X_1/h_1 \ \cdots \ H/h \ \cdots \ X_n/h_n \mid N/h)$ (where H is the category of the head of the phrase that expands N) . Collins' LPCFG achieves an F-measure performance of approximately 88 percent. Charniak and Johnson (2005) present an LPCFG with an F-score of approximately 91 percent.

Rather than encoding a particular categorical bias into his language model by excluding certain context-free rules, Collins allows all such rules. He incorporates bias by adjusting the prior distribution of probabilities over all lexicalized CFG rules. The model imposes the requirements that (1) sentences have hierarchical constituent structure, (2) constituents have heads that select for their siblings, and (3) this selection is determined by the headwords of the siblings.

The bias that Collins, and Charniak and Johnson, specify for their respective LPCFGs does not express the complex syntactic parameters that have been proposed as elements of a strong bias view of universal grammar (UG). So, for example, these models do not contain a parameter for head-complement directionality. However, they still learn the correct generalizations concerning head-complement order. The bias of a statistical parsing model has implications for the theory of UG. It expresses the prior constraints on the hypothesis space required for a particular learning procedure to achieve effective grammar induction from the input data that the corpus supplies.

1.4 Unsupervised learning

In unsupervised learning we do not annotate the training corpus with the structures or properties that the learning algorithm is intended to produce as its output values. Rather, the algorithm is provided with the data alone, and must learn some interesting structure through identifying distributional patterns and clustering properties of more basic features in the training data. In a machine learning sense, the most basic task of unsupervised learning is density estimation, which in NLP generally involves language modeling (see Chapter 3 of this book, STATISTICAL LANGUAGE MODELLING).

In the case of grammar induction, we are interested in recovering phrases and hierarchical constituent structure, which could be used for language modeling, machine translation, or other NLP tasks.

We will also briefly consider semi-supervised learning (see Abney 2008). This approach recognizes that in reality there will only be limited amounts of annotated data, and yet such data can be extremely useful when combined with much larger amounts of unannotated data.

2 Computational Learning Theory

One way of gaining insight into the problem of unsupervised learning is through theoretical analysis. While supervised learning has been the subject of detailed theoretical investigation that has yielded the design of efficient classification algorithms (Vapnik 1998), unsupervised learning of language offers a different kind of challenge. The initial formulations of the problem, most notably by Gold (1967), suggested that it is fundamentally intractable. Subsequent accounts within the PAC (probably approximately correct) learning framework (Valiant 1984; Kearns & Vazirani 1994) appeared to confirm this conclusion. As a result, while there have been numerous attempts over the years to learn grammars from raw data, very few have been informed by theoretical learning models. Instead, these efforts have relied primarily on heuristics. The very earliest attempts at unsupervised grammar induction (Lamb 1961) lacked any theoretical underpinnings, and most current work in this area continues to pursue a non-theoretical, heuristic approach.

In our view, the theoretical problems have been misunderstood, and, in some cases, not properly formulated. Learnability results depend on quite subtle details of the formalisms. Small changes in the modeling assumptions can produce radically different results. In this section, we will review some of the competing theoretical models for unsupervised learning of natural languages, and draw conclusions that depart substantially from the received wisdom of the field. Our goal is to use formal methods to illuminate the nature of learning through realistic assumptions. If the model trivializes the learning problem so that anything is learnable, then it is vacuous. Conversely, if it rules out efficient learning where we know that learning takes place, then it is clearly misguided.

As we have noted, the field of grammatical inference owes its origins to Gold (1967). In his paper Gold presents a number of different learning paradigms. We limit ourselves to the one in which the learner must acquire a language class only from positive data. As has been pointed out before, this model suffers from a number of serious shortcomings.[4] On one hand, it fails to place restrictions on the learner that are necessary to achieve rapid learning within the available resources of time and computation. On the other hand, it imposes excessively stringent limitations on learning by requiring that languages be acquired under far more difficult circumstances than those which children have to deal with.

We will discuss one of these problems briefly to give a sense of what is involved. In the Gold paradigm the learner is provided with an infinite sequence of examples. His/her model requires the learner to produce correct grammaticality judgments after making only a finite number of errors. The learner must do this for every possible presentation of the language. A presentation is characterized so that every string in the language (every grammatical sentence) appears at least once in the data, and no ungrammatical sentences are included. These are the minimum requirements for a presentation to fully exhibit a particular language (rather than others). But on reflection this paradigm makes absurd demands on human learning. The learner is obliged to acquire a language on every presentation, even when the sequence of data samples are chosen by an infinitely powerful adversary, with knowledge of the internal structure of the learner, who is designing the presentation in order to make learning maximally difficult. This situation does not correspond to the one in which children normally acquire their language. They are generally exposed to helpfully organized sequences of sentences from supportive adults interested in facilitating learning. It is instructive to work through the proof of Gold's most celebrated result, that no supra-finite language class is identifiable in the limit, to see the crucial role that unconstrained presentations of data play in this proof.[5]

Conversely, because sample presentations are not restricted, Gold cannot constrain either the speed or the complexity of the learning process (although subsequent researchers have tried to add constraints to control these properties, such as Pitt (1989) and de la Higuera (1997)).

In the Gold model, there are two important positive results. The first is that the class of all finite languages is learnable. The second is that any finite class of languages is learnable. The first class is infinite, but its members are all finite. The second is finite, but one or more of its elements can be infinite. Both of these results use fairly trivial learning algorithms.

To learn the class of all finite languages the learner uses rote learning. He/she does not need to generalize at all, but can simply memorize the examples that have been seen. At each point, the learner returns the maximally conservative hypothesis that the language he/she is learning consists of only those sentences that he/she has already seen. It is easy to see that this very simple process of enumeration allows for only a finite number of errors, where the number of errors is bounded by the size of the language.

To learn a finite class of languages from a presentation, the learner proceeds as follows. The learner has access to a hypothesis space of all the languages in the class, where these languages are arranged in a superset hierarchy. The smallest language appears at the lowest point of the hierarchy and the largest at the top. As we noted, every data presentation consists of the strings of a language containing at least one appearance of each string. When a learner encounters a sentence in a presentation, he/she deletes from the hypothesis hierarchy any language that does not contain that sentence. He/she returns the first language (hence the smallest) that is compatible with the strings of the presentation. It is easy to see that, as the presentation approaches the limit, the learner will return the correct language for the data.

Gold proves a negative result to the effect that no supra-finite language class can be learned in the limit from positive data samples presented arbitrarily from a corpus. However, he also demonstrates that with negative as well as positive evidence the class of primitive recursive languages can be learned in the limit. This class includes the set of context-sensitive languages as a proper subset. In this Gold learning paradigm negative evidence is provided by an informant who acts as a decision procedure, telling the learner for each data sample presented whether it is in the language to be learned, or in its complement set.

The view of learnability for language classes that is associated with Gold's theorems has provided one of the motivations for the principles and parameters (P&P) view of UG.[6] If the relevant formal results concerning grammar induction are those just cited and we assume that children do not have access to negative evidence, then, given that they do generalize, one might conclude that they can only effectively acquire their grammar if there is a finite number of possible human languages. This would seem to follow from Gold's learnability results, and the assumptions that (1) natural languages are infinite, and (2) children do not have access to negative evidence. The assertion that the class of natural languages is finite follows from the P&P claim that UG contains a finite set of parameters, with a finite set of values, ideally binary (Chomsky 1981). While both advocates and critics of linguistic nativism have, for the most part, agreed in the past that negative data does not play a significant role in language acquisition, this issue has become increasingly controversial in recent years.[7]

In fact the assumption that effective learning requires a finite hypothesis space of possible grammars is incorrect. There are many positive results even within the Gold paradigm which establish that infinite classes of infinite languages are learnable with some non-trivial algorithms (Angluin & Laird 1988; Clark & Eyraud 2007).

It is important to keep in mind that, because the Gold paradigm does not accurately reflect the situation of the child learner, any conclusions we draw from it are not likely to be reliable. Rather than trying to repair it by adding various constraints on presentations and polynomial bounds on the amount of possible computation, generating samples by a fixed distribution, etc., we take a different approach. We will construct a model based on the actual facts of language learning, rather than first starting with a model and then trying to force it onto the facts. We shall end up with a model that resembles that of Valiant (1984), but which departs from it in several key respects.

We start with some standard assumptions. The objects being learned are languages, which will normally be infinite objects, and these will be represented by finite systems. These systems can be thought of as grammars, though they might be encoded in another kind of formalism. The learner is provided with some information about the language. In the most basic case this will be examples of sentences in the language, though other sources of information may be considered. We assume that the learner is provided with the information one piece at a time, in a sequence of steps, and that at each step the learner either selects a hypothesis in the form of a representation of the language, or he/she abstains in the early phases of the algorithm. We say that the learner has successfully learned the language if, as the amount of data increases, the hypothesis converges (in a sense to be made precise) to the correct language.

This very rough outline provides a framework within which we can construct particular models of learning, through specifying precisely details like the classes of representations and languages, the sorts of information that the learner is provided with, the definition of convergence, and additional constraints one might want to place on the learner. Obviously, in order to achieve computational tractability we will need to make certain simplifying assumptions. In some cases, these assumptions will make learning more difficult, while in others they may make it easier. It is important to monitor these assumptions closely when interpreting the formal properties of each model. We need to emphasize that learning does, of course, occur in the real world. Therefore, if our model predicts that learning is impossible, it is clearly wrong.

We now proceed to develop this framework by making appropriate choices for the components that we have indicated. The first and most critical one to consider is the class of languages (or representations of them). This class corresponds to the set of possible grammars from which the child must select the grammar of his/her language. We know that it must include all of the attested natural languages, and presumably all languages that differ from them only through lexical changes, and other minor differences. The key questions are the following. How much larger can this class be while remaining effectively learnable? What are the defining properties of this class that determine its learnability?

We assume for the moment that the learner is provided only with positive examples. There are three possibilities to consider. First, the samples are provided by an adversary, as in Gold's model. Second, the samples are presented randomly. Under standard assumptions we can say that they are generated independently and identically from some fixed distribution. Third, the samples are produced helpfully, by a teacher trying to assist the learner (Goldman & Mathias 1996). While this last possibility may seem the most plausible, it is difficult to formalize in a way that does not trivialize the learning problem. Therefore, we will choose the random option as a reasonable model.

We observe that the child learns rapidly, in the sense that languages are complex objects, yet the amount of data which the child requires is only in the range of tens of millions of words. Hence we require the learner to learn in a time that is polynomially bounded in the size of the representation being learned, and we constrain the learner to be efficient, in that the computation it requires is bounded also by a polynomial function for the amount of data it sees.[8]

As for convergence, in the real world we generally do not see exact identification of a correct hypothesis: indeed we cannot directly observe the hypotheses. Instead we find that disagreements on grammaticality judgments are infrequent among members of a speech community. Generational differences do, of course, emerge as languages change. As a convergence criterion we can require that the probability of disagreement between the learner and the adult grammar tend to zero as it sees more data, and this must happen rapidly.

These conditions naturally yield a version of the PAC learning paradigm. In this framework a hypothesis (such as a grammar) is learned to within a range of error, represented by a constant ϵ, and a range of probability, expressed by a constant δ, in relation to the size of a data sample. An algorithm A PAC-learns a class of representations for languages $\mathcal{R}$, if and only if,

(1) there is a polynomial q, such that
   (a) for every $R \in \mathcal{R}$, which defines a language L
   (b) every probability distribution D on the samples of the data, and
   (c) every $\epsilon, \delta > 0$,
(2) whenever A sees a number of examples greater than $q(1/\epsilon, 1/\delta, |R|)$,
   (a) it returns a hypothesis H such that,
   (b) with probability greater than $1 - \delta$,
   (c) the error of the hypothesis $P_D((H - L) \cup (L - H)) < \epsilon$, and
(3) A runs in polynomial time in the total size of the examples seen.

These conditions require that learning be rapid for any language, that complex languages take more time than simple ones, but that the growth in time for learning in proportion to complexity of the language be slow.

Note that for a realistic model it is important to incorporate this dependency of learning time on language complexity, as removing it leads to the absurd conclusion that a rote learner cannot acquire finite languages. Thus for the class of finite grammars, where the representation is just a list of the grammatical sentences, it is unrealistic to expect the learner to be able to learn any list, no matter how long, in a fixed amount of time. A rote learner can learn lists of a restricted size within a reasonable time, but will require more time to learn longer lists. Thus it is reasonable, and standard in the machine learning literature, to allow the number of samples, as expressed by the polynomial q, to depend on the size of the representation $|R|$.

A standard PAC-model is distribution free, which entails that learning is equally rapid for all possible probability distributions on the data. From a mathematical perspective, this assumption is very convenient, and it forms the basis for the VC (Vapnik–Chervonenkis) theory of learnability (Vapnik 1998). However, it is unrealistic. The samples to which a child is exposed are generated by people in his/her environment who speak the language he/she is acquiring. The distributions of samples in the *primary linguistic data* (PLD) are not selected to make learning difficult, but rather to help it proceed.[9] Clearly, the distribution of the samples must depend on the language being learned: French children hear different sentences from English ones.

Many researchers (Li & Vitányi 1991) have noted that the distribution free assumption of the classical PAC framework is harsh, but yields powerful techniques. This approach may be mathematically desirable, and it might provide improvements over other estimation methods (Collins 2004). However, if we require learnability for any distribution, we find that learning becomes intractably hard. By contrast, if we restrict the class of distributions in some way, for example to simple distributions (Li & Vitányi 1991; Denis 2001) or to distributions generated by the stochastic variations of the representations, such as probabilistic deterministic finite state automata (PDFA) or PCFGs, then we find that efficient learning is possible (Clark & Thollard 2004; Clark 2006).

Additional problems for learnability derive from the computational complexity of the learning problem. Learning statistical models of the kinds standardly employed in current NLP work is hard. So, for example Abe and Warmuth (1992) show that training a hidden Markov model (HMM) is computationally hard under common assumptions.[10] On standard cryptographic methods, computationally hard problems can be embedded in the learning of even simple acyclic deterministic automata (Kearns & Valiant 1989; Kearns et al., 1994). The natural conclusion is that the child would not be able to learn such classes. Indeed the sorts of languages that these problems give rise to bear no relation to natural languages, as they involve computing parity functions, or multiplying large integers together. From a formal point of view this means that uniform learning over the entire class of languages is not possible.

However, Ron et al. (1998) suggest a useful strategy for dealing with these difficulties. A class of languages can be stratified by a parameter that separates it into subclasses according to how hard each one is to learn. The specific parameter for Ron is a distinguishability condition. Similar approaches can be applied to the learnability of context-free grammars (Clark 2006).

2.1 Summary

What insight can we gain from these formal results and considerations? Unsupervised learning of languages is difficult but possible. This is a favorable outcome, as it implies that the study of learnability can offer us useful guidance in dealing with both engineering and cognitive issues in grammar induction.

Under the best possible theoretical analysis, we can see that negative results rule out uniform learning from positive data of the full classes of regular languages and context-free languages, but that regular languages, represented by deterministic finite state automata, and some subclasses of context-free languages, may be learnable when the distributions of examples are benignly specified. Both of these representations are based on observable properties of languages. The non-terminals or states are identified with distributional properties of the substrings of the languages. In the case of the regular languages, these are the residual languages (Clark & Thollard 2004), and with context-free languages these are the congruence classes (Clark 2006). Conversely it seems that representations based on deep hidden structures, such as trees, especially trees with many empty nodes, where the structure is not directly detectable from the surface utterance, may be hard to learn.

We might also be able to obtain positive results for a class of languages that is very restricted or even finite, although the languages in this class may themselves be infinite. But even here we may encounter problems. Finiteness in itself does not ensure efficient computation. For example, the negative results in Kearns et al. (1994) are based on finite sets of finite languages. Despite the fact that they are finite, they are unlearnable, because the problem of identifying the correct hypothesis is too hard. Even though these families of languages are specified by a small number of binary valued parameters, the parameters are very tightly entwined in the computation of a parity function. This causes the class to be not efficiently learnable.

From a theoretical point of view, the interesting question is whether these results rule out domain-general learning approaches, and necessitate a very restricted class of languages. The answer seems to be that they do not. They clearly point to different language classes from those in the Chomsky hierarchy. The classes that we use in our learning analyses do not necessarily correspond to normal families of languages, and certainly not to the Chomsky classes, such as context-free grammars. They might include, for example, some regular languages, some context-free languages, and some context-sensitive languages, but they may not cover all the members of these classes. It is also important not to confuse the hypothesis class of the learner with the class of languages that may be learnable. As Poggio et al. (2004: 422) say:

> Thus, for example, it may be possible that the language learning algorithm may be easy to describe mathematically while the class of possible natural language grammars may be difficult to describe.

The hypothesis class could be very much larger than the class of languages for which it is guaranteed to learn. So, for example, the learner in Clark (2006) represents its hypotheses as context-free languages. All of these hypotheses lie within the (smaller) class of non-terminally separated (NTS) languages. The proof given there establishes that it will learn a PAC-learnable class of unambiguous languages under some plausible assumptions about the data sample distributions. But if the samples are generated adversarially (as in one of Gold's paradigms), then the learner is only guaranteed to acquire the still smaller class of substitutable languages (Clark & Eyraud 2007). The algorithm, however, remains unchanged.

These are rather different conclusions from other recent analyses. For example, Nowak et al. (2002) and Niyogi (2006) claim that the PAC-analysis rules out learning without specific restrictions. This is largely because their approach does not allow the size of the language representation to depend on the amount of data that the learner can have, as discussed above in Section 1.1.

Our theoretical understanding of learning is changing rapidly. Modifying Chomsky's terminology somewhat, we can say that linguistic representations may achieve varying levels of adequacy. Observational adequacy is the requirement that the representations are sufficiently powerful to express the distinction between grammatical and ungrammatical sentences. Explanatory adequacy imposes the additional requirement that the representations can be learned from the available data.

We have not yet achieved explanatory adequacy. The most descriptively adequate frameworks use very powerful systems of representation, such as tree adjoining grammar (TAG) (Joshi 1987) or head driven phrase structure grammar (HPSG) (Pollard & Sag 1994), while the grammars developed to date that can be efficiently learned are not powerful enough to cover the full complexity of natural language syntax. Whether there are observationally adequate grammars that can be learned using unsupervised learning from raw corpora remains very much an open question. Our theoretical analysis points in general towards shallower linguistic representations, regardless of whether these are conceived of in terms of parameters of a language model, formal grammars, or a more situated account of learning, which leverages extralinguistic context to a far greater extent than considered here.

3 Empirical Learning

We now turn to empirical work on unsupervised learning, where ML algorithms are applied to naturally occurring natural language corpora. We will look in detail at two NLP tasks. One is the unsupervised learning of word classes, and the other is unsupervised induction of syntactic parsing.

First, we will briefly take up the problem of evaluation, which is particularly problematic in the case of unsupervised learning.[11] Three methodologies have been used. The first is naïve. It involves having observers evaluate an algorithm's output on the basis of their intuitions concerning the property or structure that the procedure is designed to identify. This approach may offer some insight into the strengths and weaknesses of the method, but it is both subjective and imprecise.

A second evaluation technique measures the correspondence between the results that the algorithm generates and those of a gold standard for the corpus. So, for example, when evaluating induced word classes one can compare the word classes that an ML procedure generates for a corpus with the traditional lexical categories that are assigned to the corpus by a reliable part-of-speech (POS) tagger that uses these categories. This comparison can be done using standard information theoretic criteria. For example the conditional entropy of the gold standard tags with respect to the induced tags will tell you how much of the information in the gold standard tags remains unaccounted for by the induced tags. If this number is very low or zero, then the gold standard tags are predictable from the induced tags.

Table 8.1 Comparison of different tag sets on IPSM data. Conditional entropy of row given column. Blanks (–) are where the two sets have different tokenization due to differing treatment of the possessive clitic

Tag set	n	H	*Brown*	*ICE*	*LLC*	*LOB*	*Parts*	*POW*	*Sec*	*UPenn*
Brown		3.16	0.00	–	0.34	0.22	1.10	0.99	0.32	–
ICE		3.38	–	0.00	–	–	–	–	–	0.84
LLC		3.34	0.52	–	0.00	0.44	1.30	1.00	0.45	–
LOB		3.24	0.31	–	0.35	0.00	1.20	1.00	0.24	–
Parts		2.46	0.41	–	0.40	0.41	0.00	0.75	0.38	–
POW		2.72	0.55	–	0.42	0.46	1.00	0.00	0.43	–
Sec		3.24	0.40	–	0.35	0.24	1.20	0.95	0.00	–
Upenn		2.92	–	0.38	–	–	–	–	–	0.00

This comparison yields objective numerical evaluation, but the gold standard in linguistic annotation often incorporates theoretical assumptions that may not be well motivated. Alternative annotations of the text may be possible. The gold standard might simply reflect the prestige of the organization that produced the annotation, the theoretical framework it employs, the amount of data annotated, the availability of the corpus, or other factors irrelevant to a sound evaluation standard.

In part-of-speech annotations of English, for example, there are significant differences between various tag sets. Using data provided by the AMALGAM (Automatic Mapping Among Lexico-Grammatical Annotation Models) project (Atwell et al., 1995), which provided text annotated with eight different tag sets, we measured the conditional entropy of each tag set with respect to the others. Table 8.1 shows the results. We see that the conditional entropy here varies up to 1.3 for these equally valid, manually constructed tag sets,[12] and it is zero, as one would expect, down the leading diagonal. By comparing these competing gold standards against each other, we observe the range of possible outcomes that we might expect.

In unsupervised parsing this approach involves using a treebank and measuring derived trees against gold standard trees: an evaluation approach first employed by van Zaanen (2000).

The third and final evaluation technique is to invoke some objective and theoretically neutral evaluation strategy. For example, one can compute the predictive power of a derived language model for word class induction (Ney et al., 1994). This is usually defined in terms of perplexity, which measures the ability of the model to predict the next word in a string or corpus.[13] This evaluation metric has two advantages. First, it directly measures a useful property of the model. Such models can be used in speech recognition, and models with lower (better) perplexity will perform with a lower error rate. Second, the metric does not depend on linguistic annotations, which as we have noted, are not uncontroversial. It relies solely on raw, naturally occurring data.

Alternatively, we could consider performance in an end-to-end problem in which the results of one procedure are taken as input for a second application. The output of the latter provide an indirect measure for the success of the former. Bod (2007a) does this when he uses the trees of his unsupervised parser to support a machine translation system. However, it is not clear how well this approach captures the linguistic accuracy of the first algorithm.

3.1 Learning word classes

One of the earliest NLP problems to which unsupervised learning was successfully applied is the induction of parts of speech. The words in every language can be divided into lexical categories that partially correspond to traditional parts of speech. Nearly all lexical resources use some fixed categories of this type, as do syntactically annotated corpora. While for many purposes manual tagging of text is adequate, it is frequently desirable, for reasons of efficiency, to extract lexical classes from corpora automatically. Moreover, from a cognitive perspective it is important to determine the extent to which purely distributional algorithms can learn these categories, as they provide the basis for post-lexical syntactic analysis.

Corresponding to engineering and to cognitive concerns we find two strands of research. The cognitive science approach is most notably represented by Nick Chater and his co-workers (Finch et al., 1995; Redington et al., 1998). The engineering direction focuses on statistical language modeling, where lexical categories are invoked to smooth *n*-gram models by specifying conditional probabilities for strings in terms of word classes rather than individual lexical items. The basic methods of this approach are studied in detail by Ney et al. (1994), Martin et al. (1998), and Brown et al. (1992).

We assume a vocabulary of words $V = \{W_1, \ldots\}$. Our task is to learn a deterministic clustering, which we can represent as a class membership function g from V into the set of class labels $\{1, \ldots, n\}$. The clustering can be used to define a number of simple statistical models. The objective function we try to maximise will be the likelihood of some model, understood as the probability of the data with respect to that model. The simplest candidate is the class-bigram model, though this approach can be extended to class-trigram models. Suppose we have a corpus $w_1, \ldots, w_N$ of length N. We can assume an additional sentence boundary token. Then the class-bigram model defines the probability of the next word given the history as

$$(5)\quad P\left(w_i \middle| w_1^{i-1}\right) = P(w_i|g(w_i))P(g(w_{i-1})|g(w_{i-2}))$$

It is not computationally feasible to search through all of the exponentially many possible partitions of the vocabulary to find the one with the highest likelihood value. Therefore we need a search algorithm that will give us a local optimum. The standard techniques (Ney et al., 1994; Martin et al., 1998) use an exchange algorithm similar to the *k*-means algorithm for clustering. This procedure (1) iteratively improves the likelihood of a given clustering by moving each word from its current cluster to the cluster that will give the maximum increase in likelihood, or (2) leaves it in its original cluster if no improvement can be found. There are a number of different ways in which an initial clustering can be chosen. It has been found that the initialization method has little effect on the final quality of the clusters, but it can have a marked effect on the speed of convergence for the algorithm. A more important variation for our purposes is how rare words are treated. Martin et al. (1998) leave all words with a frequency of less than 5 in a particular class, from which they may not be moved.

These techniques, using purely distributional evidence, work remarkably well for frequent words. However, as Rosenfeld (2000b: 1313–14) points out, in language modeling the most important task is to cluster the infrequent words. We have sufficiently reliable information about the statistical properties of the frequent words that they do not need to be smoothed with the clusters, and so it is the *infrequent* words that are most in need of smoothing.[14] But it is these words that are most difficult to cluster.

Distributional data is of course not the only information relevant to identifying the syntactic category of a word class. Words are not atoms, but sequences of letters or phonemes, and this information can be used by a learning algorithm. Moreover, words have relative frequency, and infrequent words will exhibit different frequency patterns from frequent words. Pronouns, for example, tend to be very frequent.

Consider a trivial case of the first type from written language. If we encounter an unknown word, say *£212,000*, then merely looking at the sequence of characters that compose it may well be sufficient to allow us to reliably estimate its part of speech. Less trivially, suffixes like *-ing* or *-ly* on an English word are a strong clue as to its lexical category.

Clark (2003) presents a method for determining how frequency and morphological information can be incorporated into this approach, and tests the method on a number of different languages from different families. He uses texts prepared for the MULTEXT-East project (Erjavec & Ide 1998), which consists of data (George Orwell's novel *1984*) in seven languages: the original English together with Romanian, Czech, Slovene, Bulgarian, Estonian, and Hungarian.

Table 8.2 from Clark (2003) shows the results of the cross-linguistic evaluation of this data (to get a sense of how to interpret the values in this table it is worth consulting Table 8.1 again).

This method was also evaluated by comparing the perplexity of a class-based language model derived from these classes.

3.2 *Unsupervised parsing*

Initial experiments with unsupervised grammar induction (like those described in Carroll & Charniak 1992) were not particularly encouraging. Far more promising results have been achieved in work over the past decade. Klein and Manning (2002) propose a method that learns constituent structure from POS tagged input by unsupervised techniques. It assigns probability values to all subsequences of

Table 8.2 Cross-linguistic evaluation: 64 clusters, left all words, right $f \leq 5$. We compare the baseline with algorithms using purely distributional (D) evidence, supplemented with morphological (M) and frequency (F) information

	Base	*D0*	*D5*	*D+M*	*D+F*	*D+M+F*	*Base*	*D0*	*D+M*	*D+F*	*D+M+F*
$H(G \mid C)$	*All words*						$f \leq 5$				
English	1.52	0.98	0.95	1.00	0.97	**0.94**	2.33	1.53	1.20	1.51	**1.16**
Bulgarian	2.12	1.69	1.55	1.56	1.63	**1.53**	3.67	2.86	**2.48**	2.86	2.57
Czech	2.93	2.64	2.27	2.35	2.60	2.31	4.55	3.87	3.22	3.88	3.31
Estonian	2.44	2.31	**1.88**	2.12	2.29	2.09	4.01	3.42	**3.14**	3.42	**3.14**
Hungarian	2.16	2.04	1.76	1.80	2.01	**1.70**	4.07	3.46	**3.06**	3.40	3.18
Romanian	2.26	1.74	1.53	1.57	1.61	**1.49**	3.66	2.52	**2.20**	2.63	2.22
Slovene	2.60	2.28	**2.01**	2.08	2.21	2.07	4.59	3.72	**3.25**	3.73	3.55

tagged elements in an input string, construed as possible constituents in a tree. The model that this method employs imposes the constraint of binary branching on all non-terminal elements of a parse-tree. Klein and Manning invoke an *expectation maximization* (EM) algorithm to select the most likely parse for a sentence. Their method identifies (unlabeled) constituents through the distributional co-occurrence of POS sequences in the same contexts. The model partially characterizes phrase structure by the condition that sister phrases do not have (non-empty) intersections. Binary branching and the non-overlap requirement are biases of the model.

Evaluated against Penn Treebank parses (Marcus 1993) as the gold standard, this unsupervised parse procedure achieves an F-measure of 71 percent on *Wall Street Journal* (*WSJ*) test data. This score is achieved despite a serious limitation imposed by the gold standard. The Penn Treebank allows for non-binary branching for many constituents. A binary branching parse algorithm of the sort that Klein and Manning employ can only achieve a maximum F-score of 87 percent against this standard. As it turns out, many of the algorithm's binary constituent analyses that are excluded by the gold standard are, in fact, linguistically defensible parses. So, for example, while the treebank analyzes noun phrases as having flat structure, the iterated binary branching constituent structure that the Klein–Manning procedure assigns to NPs is well motivated on syntactic grounds.

The Klein–Manning parser is, in fact, constructed by semi-supervised, rather than fully unsupervised, learning. The input to the learning algorithm is a corpus annotated with the POS tagging of the Penn Treebank. If POS annotation is, in turn, provided by a tagger that uses unsupervised learning, then the entire parsing procedure can be construed as a sequenced process of unsupervised grammar induction.[15]

Klein and Manning (2002) report an experiment in which their parser achieves an F-score of 63.2 percent on *WSJ* text annotated by an unsupervised POS tagger. They observe that this tagger is not particularly reliable. Other unsupervised taggers, like the one presented in Clark (2003), produce good results that might well allow the Klein–Manning unsupervised constituency parser to perform at a level comparable to that which it achieves with Penn Treebank tags.

Klein and Manning (2004) present an unsupervised learning procedure for acquiring lexicalized head-dependency grammars. It assigns probabilities to possible dependency relations in a sentence *S* by estimating the likelihood that each word in *S* is a head for particular sequences of words to its left and to its right, taken as its syntactic arguments or adjuncts. The probabilities for these alternative dependency relations are computed on the basis of the context in which each head occurs. The context consists of the words (word classes) that are immediately adjacent to it on either side. The dependency structure model associated with the learning algorithm requires binary branching as a condition on dependency relations. The procedure achieves an F-measure of 52.1 percent on Penn Treebank test data.

Klein and Manning (2004) combine their dependency and constituent structure grammar induction systems into an integrated model that produces better results than either of its component parsers. The composite model computes the score for a tree as the product of the dependency and constituency structure grammars. This procedure employs both constituent clustering and head dependency relations to predict binary constituent parse structure. It achieves an F-score of 77.6 percent with Penn Treebank POS tagging, and an F-score of 72.9 percent with Schütze's (1995) unsupervised tagger.

Bod (2006a; 2007a; 2007b) proposes an alternative system for unsupervised parsing, which he refers to as *unsupervised data-oriented parsing* (U-DOP). U-DOP generates all possible binary branching subtrees for a sentence *S*. The preferred parse for *S* is the one which can be obtained through the smallest number of substitutions of subtrees into nodes in larger trees. In cases where more than one derivation satisfies this condition, the derivation using subtrees with the highest frequency in previously parsed text is selected. Bod (2006a) reports an F-score of 82.9 percent when U-DOP is combined with a maximum likelihood estimator and applied to the *WSJ* corpus on which Klein and Manning tested their parsers.

While U-DOP improves on the accuracy and coverage of Klein and Manning's (2004) combined unsupervised dependency-constituency model, it generates a very large number of subtrees for each parse that it produces. Bod (2007a) describes a procedure for greatly reducing this number by converting a U-DOP model into a type of PCFG. The resulting parser produces far fewer possible subtrees for each sentence, but at the cost of performance. It yields a reported F-score of 77.9 percent on the *WSJ* test corpus (Bod 2007a).

An important advantage that U-DOP has over simple PCFGs is its capacity to represent discontinuous syntactic structures, like subject–auxiliary inversion in questions, and complex determiners such as *more . . . than . . .*, as complete constructions.[16] U-DOP incorporates binary branching tree recursion as the main bias of its model. It can parse structures not previously encountered, either through the equivalent of PCFG rules, or by identifying structural analogies between possible tree constructions for a current input and those assigned to previously parsed strings in a test set.

ADIOS is another recent unsupervised algorithm for grammar induction (Solan et al., 2005). It is interesting not so much for the algorithmic properties that it exemplifies (these are largely taken from other models, although they are combined in a novel way), but for the extensive and original method of evaluation to which it is subjected. Solan et al. (2005) use a number of different techniques to demonstrate the robustness of ADIOS. These include a language modeling task, and application of the algorithm to test children's reading comprehension.

3.3 Accuracy vs. cost in supervised, unsupervised, and semi-supervised learning

In general supervised learning algorithms achieve greater accuracy than unsupervised procedures. So LPCFG parsers trained on *WSJ* corpora annotated with constituent structure information in the Penn Treebank obtain F-measures of 88 percent to 91 percent (Collins 1999; Charniak and Johnson 2005), while efficient unsupervised parsers currently score in the mid to high 70s (Klein & Manning 2004; Bod 2007a). However, hand annotating corpora for training supervised algorithms adds a significant cost that must be weighed against the accuracy that these procedures provide. To the extent that unsupervised algorithms do not incur these costs, they offer an important advantage, if they can sustain an acceptable level of performance in the applications for which they are designed.

Banko and Brill (2001) use a method of semi-supervised learning that combines some of the benefits of both systems. They train 10 distinct classifiers for a word disambiguation problem on an annotated test set. They then run all the classifiers on an unannotated corpus and select the instances for which there is full agreement among them. This automatically annotated data is added to the original hand annotated corpus for a new cycle of training, and the process is iterated with additional unannotated corpora. In the experiments they describe how accuracy is improved through unsupervised extensions of a supervised base corpus up to a certain phase in the learning cycles, after which it begins to decline. They suggest that this effect may be due to the learning process reaching a point at which the benefits that additional data contribute are outweighed by the distortion of sample bias imported with the new samples, which causes overfitting of the data.

Bank and Brill's approach can be generalized to grammar induction and parsing. This would involve training several supervised parsing systems on an initial parsed corpus and then optimizing these procedures through iterated parsing of text containing only POS tagging. The tagging can be done automatically using a reliable tagger.

There are, in fact, good engineering reasons for investing more research effort in the development of robust unsupervised and semi-supervised learning procedures. Very large quantities of raw natural language text are now available online and easily accessible. While supervised grammar induction has achieved a high level of accuracy, generating the necessary training corpora is an expensive and time-consuming process. The use of unsupervised and semi-supervised learning algorithms reduces much of this expense. The amount of data that hand annotated training sets provide is very limited in comparison to the corpora of unannotated text currently available at little or no cost. As the accuracy and coverage of unsupervised systems improves, they become increasingly attractive alternatives to supervised methods. It is reasonable, then, to expect a greater focus on the development of these systems in future NLP work.

4 Unsupervised Grammar Induction and Human Language Acquisition

The promising results of recent work on unsupervised procedures for grammar induction raise interesting questions for long-standing debates over the cognitive basis for human language acquisition. Theoretical linguistics has been dominated for the past 50 years by a strong version of linguistic nativism.[17] On this view, a set of rich, domain-specific biases provide the basis for language acquisition. These biases are formulated as the constraints of a universal grammar, which constitutes a biologically determined, task-specific language faculty.

The main consideration offered in support of this notion of a language faculty is the *argument from the poverty of the stimulus* (APS). According to the APS the amount and quality of the primary linguistic data available to children acquiring their first language is not sufficient to account for the grammar that expresses adult linguistic competence if acquisition of the adult grammar is mediated primarily by domain-general procedures of induction, such as those applied in machine learning. A classic instance of the APS is the use of subject–auxiliary inversion to claim that language learners have an innate bias towards learning grammatical rules formulated in terms of a hierarchical phrase structure representation of sentences.[18]

(6) a. Is the student who is in the garden hungry?
b. *Is the student who in the garden is hungry?

The rule of auxiliary inversion requires that (something like) the following structures be assigned to (6a), (6b), respectively.

(7) a. [$_{S'}$ is$_2$ [$_S$[$_{NP}$ the [$_{N'}$ student [$_{RC}$ who [$_{VP}$ is$_1$ in the garden]]]] [$_{VP}$ [$_V$ e$_2$] hungry]]]
b. [$_{S'}$ is$_1$ [$_S$[$_{NP}$ the [$_{N'}$ student [$_{RC}$ who [$_{VP}$ e$_1$ in the garden]]]] [$_{VP}$ [$_V$ is$_2$] hungry]]]

Advocates of the APS maintain that the data to which children are exposed does not provide an adequate basis for inferring a structure-dependent rule of subject–auxiliary inversion unless the children come to the task of language acquisition already equipped with a mechanism for organizing strings of words into phrasal constituents of the sort that facilitate the formulation of this rule.

The APS has recently been subject to strong challenges.[19] Both sides of this debate have tended to focus on the availability of evidence for grammar induction through data-driven methods. However, it is not possible to decide how much, and what sort of, data is required for effective language acquisition independently of a clearly specified theory of learning. This question is meaningful and interesting only when considered in relation to a particular learning theory or class of such theories. Linguistic nativists have generally argued for the paucity of data without specifying a strong bias model that will generate the class of grammars which they posit, given the set of linguistic samples which they assume as evidence. Similarly, some critics of the APS have insisted that the child has access to sufficient linguistic data to produce the grammar of his/her first language without indicating how learning is achieved.

To the extent that machine learning algorithms can acquire accurate and theoretically viable grammars of languages from corpora through unsupervised methods, employing weak rather than strong learning biases, they undermine the APS as an argument for strong linguistic nativism.[20] Specifically, they show that it is possible to implement a learning algorithm that can effectively acquire a significant element of human linguistic knowledge relying primarily on generalized information theoretic techniques for classifying data, with comparatively weak domain-specific constraints on the set of possible grammars in its hypothesis space. As we have observed, unsupervised grammar induction has recently yielded encouraging results for parsing *WSJ* text according to the gold standard given by the Penn Treebank. Moreover, Bod (2006a, 2007a), and Clark and Eyraud (2006) present systems that learn subject–auxiliary inversion rules efficiently without being exposed to sample sentences like (6a) or its full declarative counterpart.

(8) The student who is in the garden is hungry.

However, most of these unsupervised grammar-induction procedures incorporate learning biases that restrict their hypothesis spaces to constituent structure grammars of some kind.[21] An advocate of the APS can claim that these biases are precisely the sort of conditions that the argument is intended to motivate as necessary learning priors for language acquisition.

In fact it is possible to argue that a preference for hierarchical constituent structure is not, in itself, an irreducible bias on a language model. It can be derived from a more basic and general learning prior. As we have seen, Perfors et al. (2006) define a very general prior for smaller grammars with fewer rules and fewer non-terminal symbols. It does not specify a bias towards constituent structure. They apply their Bayesian posterior probability measure, given in (3) ($\arg\max_H(P(H)P(D|H))$), to a hypothesis space of three types of grammar, which they evaluate on a subset of CHILDES (MacWhinney 1995), a corpus of child directed discourse.

The three types of grammar that Perfors et al. (2006) consider are:

(1) a flat grammar that generates strings directly from S without intermediate non-terminal symbols;

(2) a probabilistic regular grammar (PRG); and
(3) a probabilistic context-free grammar (PCFG).

They compute the posterior probability of each grammar for the CHILDES sentences. The PCFG receives a higher posterior probability value and covers significantly more sentence types in the corpus than either the PRG or the flat grammar. The grammar with maximum a posteriori probability makes the correct generalization. This result suggests that it may be possible to decide among radically distinct types of grammars on the basis of a Bayesian model with relatively weak learning priors, when using a corpus that accurately reflects the linguistic data that children are exposed to in the course of first language acquisition. The prior that Perfors et al. (2006) invoke does not impose a constituent structure bias, but a general preference for smaller, more compact hypotheses.

While the success of weak bias unsupervised ML procedures in grammar induction (and related tasks) vitiates the APS case for strong domain-specific learning priors as necessary conditions for language acquisition, it does not tell us anything about the actual cognitive mechanisms that humans employ in acquiring their first language. Even discounting the APS, a strong nativist view of UG could, in principle, turn out to be correct on the basis of the psychological and biological facts of language acquisition.

Is there, then, any psycholinguistic evidence showing that ML methods play a significant role in human language learning? In fact there is. Saffran et al. (1996) report a set of experiments in which eight-month-old infants learn to identify word boundaries in continuous syllable sequences on the basis of a two-minute exposure to training data. The words are nonsense terms constructed out of three-syllable sequences. The transitional probabilities between syllables within a word are maximal (set at 1), while those between syllables crossing word boundaries are low (generally around 0.33). The transitional probability of a syllable pair XY (X followed by Y) is computed as the conditional probability $P(Y|X)$ according to its Bayesian MLE condition (where $c(\alpha)$ is the frequency count for the sequence α).

(9) $P(Y|X) = \frac{c(XY)}{c(X)}$

The infants were able to distinguish familiar words heard in the training samples from novel non-words on the basis of very limited exposure to a word set. Saffran et al. (1996) conclude that they employed the difference in transitional probabilities between word internal syllable sequences and word external pairs in order to infer word boundaries.[22]

Thompson and Newport (2007) extend this experimental approach to investigate the learning of phrasal boundaries and constituent structure. They describe a series of experiments in which English-speaking adults are exposed to training sets of samples from simple artificial languages with six word classes, each containing three words (the word number of some classes is modified for one of the experiments). Phrases consist of word pairs where each element of the pair comes from a distinct word class. The training sets contain a canonical phrasal pattern of word class sequences, and variations on these patterns involving:

(1) the presence of repeated phrases,
(2) optional constituents,
(3) permutations of phrases (moved constituents), and
(4) variation in the lexical size of two of the four phrase types.

Each of the three conditions in (1)–(3) introduces a significant difference in intra-phrasal vs. inter-phrasal transitional probabilities between word classes. The former are set at 1, while the latter are lower. For each of these four conditions a control group is exposed to a training set in which the conditions do not apply to discrete phrases, but are formulated only for word classes. As a result, there is no substantial difference in the transitional probabilities that hold between different word class pairs in the control language.

After training, both the experimental and the control groups were tested on their ability to identify well-formed sentential and phrasal patterns in the language. Thompson and Newport (2007) found that for conditions (1)–(3) the experimental group outperformed the control group in learning both sentence and phrasal structure. When all four conditions were combined in a single language, the difference between intra-phrasal and inter-phrasal transitions substantially increased. In an experiment with variants of this language type in which the two groups were exposed to a comparatively small set of canonical sentence patterns (5 percent of the training set), the experimental subjects achieved far greater success than the control subjects in learning both sentence and phrasal patterns.

These results indicate that transitional probabilities can provide an important cue for identifying constituent structure from word sequences. While the experiments provide data only on syntax learning by adults, when taken together with Saffran et al.'s (1996) research on infant identification of word boundaries, they strongly suggest that Bayesian inference of the kind employed by ML methods in NLP plays a significant role in human language acquisition at a variety of levels of morphological and syntactic structure.

This work also gives credence to a bootstrapping view of language learning on which information theoretic methods yield an initial classification of linguistic entities that can then be used to construct successive levels of representation. Each previous cycle of learning provides a set of structural constraints on the entities out of which the next stage is developed by the same kinds of Bayesian inference. If this view is sustained by further research, then the weak bias model of language learning proposed in Lappin (2005), Lappin and Shieber (2007), and Clark (2004) will achieve psychological as well as computational credibility. Clearly much additional work remains to be done in clarifying these issues before any such model can be endorsed with any confidence as an account of human language acquisition. It does, however, provide a serious alternative to the strong nativist approach that has dominated linguistics and cognitive science for the past five decades, generally without a learning theory to motivate it.

5 Conclusion

Unsupervised learning is a rich and varied area of research. It includes different motivations, techniques, and methods of evaluation. In this chapter we have surveyed the field and provided an overview of what we regard as the most significant theoretical and engineering developments.

It is important to recognize that while the application of these techniques to practical problems in NLP is still at an early stage, unsupervised learning is almost certain to expand as an area of interest and activity.

It is also plausible to hope that, as we make progress in understanding the capacities and limits of unsupervised methods, we will achieve deeper insight into how much and what kinds of linguistic knowledge can be acquired by domain-general learning algorithms operating on raw linguistic data. Such insight is of direct significance to work in theoretical linguistics and the study of human cognition.

NOTES

[1] See, Chapter 18 of this book, INFORMATION EXTRACTION, Section 3.4, and Jurafsky and Martin (2009: 455) for discussions of recall, precision, and weighted F-measures.

[2] See Manning and Schütze (1999) for a discussion of Bayesian inference and the role of Bayesian reasoning about probability in statistical NLP.

[3] See Chapter 13 of this book, STATISTICAL PARSING, Manning and Schütze (1999), and Jurafsky and Martin (2009) for accounts of probabilistic context-free grammars and lexicalized probabilistic context-free grammars as language models for supervised grammar induction.

[4] See Lappin and Shieber (2007) and Clark (2001a, chapter 4) for discussions of some of the problematic assumptions in Gold's *identification in the limit* learning paradigm.

[5] A supra-finite class includes all finite languages and at least one infinite language.

[6] See, for example Crain and Thornton (1998) for arguments to the effect that, because the class of natural languages is unlearnable from positive evidence only, a rich innate UG must be posited to explain human language acquisition.

[7] See, for example, Saxton (1997) and Chouinard and Clark (2003) for psycholinguistic research supporting the widespread availability and effectiveness of negative evidence in child grammar induction. See also Clark and Lappin (2009) for a proposal on how indirect negative evidence can be stochastically modeled within a PAC framework.

[8] See Chapter 2, COMPUTATIONAL COMPLEXITY, for the relevant notions of complexity and efficiency of computation.

[9] Questions have been raised about the extent to which this helps the child (Gleitman et al., 2001).

[10] The Baum–Welch algorithm (also known as the forward–backward algorithm) used to estimate the parameter values for such models only finds a local optimum. See Manning and Schütze (1999) for discussion of this procedure.

11 See Chapter 11, EVALUATION OF NLP SYSTEMS, and Chapter 10, LINGUISTIC ANNOTATION, for discussions of this and related issues in connection with a variety of NLP tasks.

12 The texts were tagged automatically, which might introduce some variability.

13 See Chelba, STATISTICAL LANGUAGE MODELING, and Manning and Schütze (1999), Section 2.2 for discussions of perplexity and entropy.

14 See Chapter 3, STATISTICAL LANGUAGE MODELING, and Pereira (2000) on the application of smoothing techniques for statistical modeling, originally introduced by Good (1953), in NLP.

15 Actually, an unsupervised POS tagger will also rely on morphological analysis of the words in a corpus. This can be provided by an unsupervised morphological analyzer. See Goldsmith (2001), Chapter 14 of this book, SEGMENTATION AND MORPHOLOGY, and Schone and Jurafsky (2001) for alternative systems of unsupervised morphological analysis.

16 See Clark and Eyraud (2006) for a simple unsupervised distributional algorithm that learns a PCFG which correctly handles subject–auxiliary inversion.

17 See, *inter alia*, Chomsky (1965; 1971; 1981; 1986; 1995; 2000; 2005), and Pinker (1989; 1996).

18 See Chomsky (1971); Crain and Nakayama (1987); Crain (1991); Berwick and Chomsky (2009) for versions of this argument, and Clark and Lappin (2010) for critical discussion of it.

19 Pullum and Scholz (2002) and their critics conduct a lively debate on the APS in Volume 19 (2002) of *The Linguistic Review*. Scholz and Pullum (2006) offer an updated version of some of their criticisms of the APS.

20 For detailed discussion of the relevance of work in machine learning and computational learning theory to APS-based claims for linguistic nativism see Lappin (2005); Lappin and Shieber (2007); Clark (2004); and Clark and Lappin (2010).

21 This is not the case for the algorithm proposed in Clark and Eyraud (2006), which uses a simple criterion of distributional congruence to identify equivalence classes of words and phrases.

22 Yang (2004) disputes this conclusion. He reports a word identification experiment on a subset of the CHILDES corpus using transitional syllable probabilities. The results of the experiment indicate poor recall and precision for this procedure. As he observes, this is due to the fact that 85 percent of the words in his test set are monosyllabic. Therefore there is no significant distinction between intra-word and inter-word transitional probabilities for most of the terms in this corpus. Yang claims that his experiment shows that transitional probability is not an adequate cue for word boundary identification in realistic data of the sort that children receive. In fact, this claim is seriously undermotivated. Child directed speech of the kind that appears in CHILDES does not exhaust the linguistic samples to which children are exposed in their normal environments. They generally have access to the full range of multi-syllabic utterances of normal adult speech, even when it is not directed to them. There is no reason to exclude this additional data from the range of evidence that children can make use of when computing transitional probabilities for syllable pairs. It is not unreasonable to hypothesize that, when one takes account of the full range of evidence available to child language learners, a significant correlation between transitional probability patterns and word boundaries in real language data will prove robust.