A Diverse Dirichlet Process Ensemble for Unsupervised Induction of Syntactic Categories

Roi Reichart[1] *Gal Elidan*[2] *Ari Rappoport*[3]
(1) The Computer Laboratory, University of Cambridge, UK
(2) Department of Statistics, The Hebrew University
(3) Institute of computer science, The Hebrew University
Roi.Reichart@cl.cam.ac.uk, galel@huji.ac.il, arir@cs.huji.ac.il

ABSTRACT
We address the problem of unsupervised tagging of phrase structure trees with phrase categories (parse tree nonterminals). Motivated by the inability of a range of direct clustering approaches to improve over the current leading algorithm, we propose a mixture of experts approach. In particular, we tackle the difficult challenge of producing a *diverse* collection of useful tagging experts, which can then be aggregated into a final high-quality tagging. To do so, we use the particular properties of the Dirichlet Process mixture model. We evaluate on English, German and Chinese corpora and demonstrate both a substantial and consistent improvement in overall performance over previous work, as well as empirical justification of our algorithmic choices.

KEYWORDS: Unsupervised parsing, Grammar induction, Non terminals, Dirichlet Process, Ensemble learning.

1 Introduction

Grammar induction is the task of learning grammatical structure from plain text without human supervision. The task is valuable for the understanding of human language acquisition and its output can potentially be used by NLP applications, avoiding the costly and error prone creation of manually annotated corpora. The task has been widely explored (Klein, 2005) and its importance has increased due to the recent availability of huge corpora.

The induced grammar can be represented in various ways. Most work (e.g., (Klein and Manning, 2004; Smith and Eisner, 2006; Seginer, 2007; Headden et al., 2009)) annotate text sentences using an unlabeled hierarchical phrase or a dependency structure, and thus represent the induced grammar through its behaviour in a parsing task.

An important task in theory and practice that we consider in this work is how to enrich phrase structures with syntactic categories. The two grammars that have been widely explored by the NLP community in the last two decades, phrase structure grammars and dependency grammars, allow the induced structure to be either labeled or not and use the labeling to describe substantially different syntactic functions. In this paper we focus on the former formalism and induce parse tree nonterminals (e.g., 'NP', 'VP', 'PP') in an unsupervised manner. We keep the discussion of dependency parsing for future research.

Many linguistic theories posit a hierarchical labeled constituent (or constructional) structure, arguing that it has a measurable psychological reality (e.g., (Goldberg, 2006)). Practically, most of the syntactic annotation of corpora used by the NLP community comes in the form of labeled structures. Indeed, modern *supervised* syntactic parsers aim at learning labeled structures. Moreover, phrase categories are often used in a variety of NLP tasks, such as SRL (Gildea and Jurafsky, 2002; punyakanok et al., 2008), alignment in syntax-based machine translation (Zhang and Gildea, 2004), information extraction (Miyao et al., 2008), etc.

Phrase categories can be induced either jointly with the phrase structure (Haghighi and Klein, 2006) or given a previously induced structure (Borensztajn and Zuidema, 2007; Reichart and Rappoport, 2008). Reichart and Rappoport (RR08), which has the leading results, uses a two stage approach where the second stage clusters the phrases of the parse trees induced by an unsupervised parser (Seginer, 2007). This is done by inducing an over-expressive large number of categories using the BMM model of Borensztajn and Zuidema (2007), and then clustering these categories into a final set.

In this work we focus on improving the critical last stage of narrowing down the large number of induced BMM labels into a smaller set of informative categories[1]. Naively, one might think that simply replacing the simple clustering algorithm used by RR08 with a more elaborate approach would result in improved final categories. However, as we show in Section 3, a variety of clustering approaches did not lead to a noticeable improvement.

To overcome this difficulty, we adopt a qualitatively different solution and tackle this task using a multiple experts approach (Dietterich, 2000). Intuitively, averaging over multiple predictions (experts) is useful when the output of the various experts obeys two requirements: (1) the output of *each* expert is of useful quality; (2) the experts are sufficiently different from each other.

[1] It is also possible to directly cluster phrases without first inducting BMM categories. This, however, leads to inferior overall results which we do not report for clarity.

The central challenge in building such ensembles is in ensuring that different experts capture different characteristics of the problem. To do so, we build on the Dirichlet Process Mixture Model (DPMM) (Ferguson, 1973; Antoniak, 1974) which relies on the Bayesian framework to induce an a posteriori clustering. Each DPMM expert is characterized by a concentration parameter, and we vary this parameter to induce different experts (clusterings). As our experiments show, it is the particular properties of the DPMM that lead to a diverse ensemble.

Finally, with a diverse ensemble at hand, we aggregate the experts into a single coherent phrase structure tagging. Qualitatively, the tendency of two phrases to share a category should increase with the number of DPMM experts that independently cluster them together. We formalize this idea as a global optimization problem and use the k-way normalized cut algorithm (Yu and Shi, 2003) to solve it.

We evaluate our algorithm on English, German and Chinese, using various tag set sizes and evaluation measures. Our results justify our reliance on DPMM and normalized-cut, and demonstrate consistent improvement over previous work.

2 Previous Work

Unsupervised parsing attracts researchers for many years (see reviews in (Clark, 2001; Klein, 2005)). In recent years efforts have been made to evaluate the algorithms on manually annotated corpora such as the WSJ PennTreebank (Klein and Manning, 2002, 2004; Dennis, 2005; Bod, 2006; Smith and Eisner, 2006; Seginer, 2007; Cohen and Smith, 2009; Headden et al., 2009; Berg-Kirkpatrick and Klein, 2010; Blunsom and Cohn, 2010; Gillenwater et al., 2010; Spitkovsky et al., 2010a,b, 2011b,a). All these works induce unlabeled phrase or dependency structures.

In this paper we focus on the induction of syntactic categories for unlabeled phrase structures (parse tree nonterminals) and its evaluation on corpora annotated with a similar representation. There are three previous papers we are aware of that address this problem. Haghighi and Klein (2006) presented two models: *PCFG* × *NONE* and *PCFG* × *CCM*. These models use the inside-outside and EM algorithms to induce bracketing and labeling simultaneously [2].

Borensztajn and Zuidema (2007) presented the Bayesian Model Merging (BMM), a framework for inducing a PCFG containing both a bracketing and a labeling. They use Stolcke's algorithm (Stolcke, 1994; Stolcke and Omohundro, 1994) with features from Petasis et al. (2004). The BMM model uses an iterative greedy search for an optimal PCFG according to the Bayesian criterion of maximum posterior probability. The data likelihood is proportional to the data description length according to the grammar and the prior distribution on the grammar is proportional to the grammar description length.

The number of categories induced by the BMM model is very large. For example, for the 7422 sentences of the WSJ10 corpus it induces 4944 categories. To enable generalization over the unlabeled bracketing, a much smaller number of final categories should be induced.

RR08 proposed a clustering algorithm for the BMM categories. They map each of the BMM categories to one of the R most frequent categories produced by the algorithm. Frequency was determined according to the number of phrases each BMM category tags. To perform the mapping, they construct a feature vector representation for each BMM category. The vector consists of $3|M| + |S|$ features, where M is the set of BMM categories and S is the set of POS

[2] Their other models, which were the core of their paper, are semi-supervised.

tags in the corpus. For each category l, they compute the cosine metric between its vector and that of every category among the R most frequent BMM categories. l is mapped to the category with which it obtains the highest cosine score.

RR08 applied their clustering scheme to the bracketing produced by the unsupervised parser of Seginer (2007). The labeled phrase structure trees induced by RR08 are better than those induced by (Haghighi and Klein, 2006) and (Borensztajn and Zuidema, 2007) which motivates separate learning of the phrase structures and their categories. In this work we provide an alternative algorithm for BMM categories clustering. We provide a detailed comparison to the work of RR08 and show superior results.

Sangati and Zuidema (2009) proposed head assignment algorithms. The concept of a head is related to syntactic categories. Their algorithms are trained on data without head annotations, but, unlike our unsupervised approach, requires manually created labeled phrase-structure trees as input.

The concept of head naturally connects to dependency parsing (Kubler et al., 2009) which has been extensively studied in the last decade. While the categories assigned to dependency structures are different in nature from the phrase categories explored in this paper, our algorithm may be applicable to this case as well. We keep this question for future research.

DP has been used for unsupervised syntactic acquisition tasks. Finkel and Manning (2007) Used DP for unsupervised POS induction from dependency structures. Liang et al. (2009) proposed a nonparametric Bayesian generalization of PCFG, based on the hierarchical Dirichlet Process, and applied it to supervised parsing. DP has been used for many other NLP tasks as well (e.g. (Goldwater et al., 2006; Johnson et al., 2007; Haghighi and Klein, 2007; Johnson and Goldwater, 2009)). However, we are not aware of works that explored DP as a model for creating a diverse ensemble of experts for clustering tasks.

3 Building a Diverse Clustering Ensemble

As discussed, to induce phrase categories we adopt a two stage approach where we first induce a large number of categories and then narrow this into a final set. The first stage, leading to a collection of BMM categories is similar to RR08. Our novelty is in taking a qualitatively different approach to the critical stage of narrowing down the large number of categories into a small informative set.

Motivation For the Ensemble Approach. To motivate our ensemble approach we must first consider more straightforward alternatives. Probably the simplest one is to use a one stage approach where we cluster phrases directly using several clustering algorithms (K-means, complete link, single link and average link) and distance metrics (Euclidean and cosine). Another alternative, is to keep running the BMM until the desired number of categories is obtained (that is, by selecting the least harmful update of its objective when no further improvement is possible) . In preliminary experiments, these approaches resulted in inferior results to RR08. For example, for WSJ10 with 26 clusters, the best of these algorithms (K-means with cosine distance), achieves F-score with many-to-one mapping of 38.7 compared to 58.9 of RR08 (see Table 2).

Building on the good performance of RR08, we next tried to simply replace its final clustering algorithm (of BMM labels). We tried the K-mean algorithm both with a random starting point and an informed starting point with the cluster centers initialized as the K most frequent BMM

	V	NVI	Many–to–1
DP	0.26	1.53	34.1
	0.23	1.59	37
	0.2	1.68	44.1
KM	0.65	0.74	69.4
	0.6	0.83	65.2
	0.65	0.75	66.8
RR08	0.67	0.71	73.1
	0.64	0.74	71.6
	0.61	0.84	68

Table 1: Average pairwise similarity between the clustering experts induced by the different clustering algorithms we use in this paper. KM is K–means. RR08 is the algorithm of Reichart and Rappoport (2008) run each time with a different number of induced clusters. For each clustering algorithm, the first line is for WSJ10 (English), the second line is for NEGRA10 (German) and the third line is for CTB10 (Chinese). Higher V and Many–to–1 and lower NVI scores imply that the clusterings are more similar. DP produces the least similar clusterings.

categories. We tried both variants with several different cluster set sizes. In both cases, the resulting final clustering was not superior to that of RR08.

Our next step was to adopt a Bayesian approach where multiple clusterings are considered. Concretely, the Dirichlet process mixture model (DPMM) defines a distribution over clusterings that is governed by a concentration parameter α. Given α, to get a specific clustering, the *maximum a-posteriori* clustering is a natural choice. One can also consider defining a prior over α to lessen the arbitrariness of the choice of parameters. We tried DPMM with a uniform prior over α which did not lead to an improvement in the results. Furthermore, individual clusterings for a range of α values, were all of similar quality. Thus, we cannot expect a more informed prior over α to lead to better performance[3].

Ensemble Construction: First Approach. With the above evidence as to the need of an ensemble approach, we are still left with the challenge of constructing diverse clustering experts. Intuitively, if we allow a different number of clusters for each experts we would often get qualitatively different solutions as, for example, the best three cluster solution is typically not a simple refinement of the best two cluster solution. The simplest approach to carry this out is to run different K-means, each with a different number of target clusters. Another possibility is to use the method of RR08, again with a different number of target clusters. Unfortunately, as Table 1 shows, in both cases the similarity of the different experts (clusterings) is quite high. Indeed, as we report in Section 7, this did only result in small improvement of the performance.

Part of the difficulty with this approach for constructing a clusterings ensemble is that both clustering methods used highly depend on the initial (informed) starting point. Unfortunately, as discussed above, starting with random initializations leads the K-means algorithm to low-quality clusterings.

Ensemble Construction: Improved Approach. To overcome the sensitivity to the initial clustering, we consider the Bayesian DPMM framework. Using standard MCMC techniques

[3] The effective range of α values in our experiments is $[10^{-4}, 10]$. We run the DP algorithm 1000 times with α values changed from 10^{-4} to 10 in steps of 10^{-3}.

(e.g., Gibbs sampling), the resulting clustering converges to the mode of the posterior solution, which is *independent* of the initialization of the procedure[4]. We start by describing the DPMM framework and then explain how it can be used to construct multiple diverse clusterings.

The DPMM defines a prior over the number of clusters that can be described via the so-called Chinese restaurant process (CRP). In this metaphor we have a Chinese restaurant with an infinite number of tables (clusters), each of which can seat an infinite number of customers (BMM categories). The first customer enters the restaurant and seats at the first table. When a new customer (BMM category) arrives at the restaurant, s/he either sits at an existing table with probability proportional to the number of customers already seated at the table (cluster size), or at a new table with probability proportional to α. This process defines a coherent probabilistic prior over the number of clusters (Ferguson, 1973).

Formally, let G be a collection of likelihood parameters, sampled for each cluster. Parameters $\theta_{1:M}$, one for each sample, are drawn from G and, finally, observations, $x_{1:M}$ (the BMM categories) are drawn from a distribution associated with the parameters. Each observation x_i is represented by a vector in N^d, and the values in its corresponding parameter vector, $\theta_i \in R^d$, sum to 1 (d is fixed for all observations). The probability of the i-th observation x_i is associated with the i-th parameter θ_i by a likelihood function $F(\theta_i)$. Now, since the number of clusters is unknown a-priori, the DP defines a distribution over G, governed by a base measure G_0 (parameter factory) and the concentration parameter α. The entire generative process is defined by the equations:

$$G|\alpha, G_0 \sim DP(\alpha, G_0)$$

$$\theta_m|G \sim G$$

$$x_m|\theta_m \sim F(\theta_m)$$

Our likelihood function $F(\theta_i)$ is a multinomial (see below) with parameters $\theta_{1:M}$. G_0 is chosen to be the Dirichlet distribution, the conjugate of the multinomial. The DP concentration parameter, α is kept fixed during the clustering process.

Given a DPMM, a concrete clustering is defined by the mode of the posterior distribution. Each cluster is assigned a different parameter from the collection defined by G, and observations (x_i, BMM labels) belonging to the same cluster (final phrase category), share this parameter.

Constructing multiple experts using the DPMM framework is simply done by varying the concentration parameter α. That is, each expert is associated with a different α value, resulting in a different clustering. While the quality of each such expert is not substantially different than RR08, as Table 1 shows, the resulting experts (clusterings) are substantially more varied than the alternatives considered above.

4 Model Averaging

We now face the task of aggregating the individual clusterings induced by the different DP experts. Intuitively, if several experts independently cluster together two BMM categories, our belief that these categories belong in the same cluster should increase.

[4] In practice, there are deterministic effective alternatives to the stochastic Gibbs procedure which are highly effective. In this work we use the one proposed in Daume (2007).

We now formalize this idea using the k-way normalized cut clustering algorithm (Yu and Shi, 2003). Its input is a undirected graph $G = (V, E, W)$ where V is the set of vertices, E is the set of edges and W is an edge weight matrix assumed to be non-negative and symmetric. For $A, B \subseteq V$ define:

$$links(A, B) = \sum_{i \in A, j \in B} W(i, j).$$

Using this definition, the normalized link ratio of A and B is defined to be:

$$NormLinkRatio(A, B) = \frac{links(A, B)}{links(A, V)}.$$

The k-way normalized cut problem is to minimize the links that leave a cluster relative to the total weight of the cluster. Denote the set of clusterings of V that consist of k clusters by $C = \{c_1, \ldots c_t\}$ and the j-th cluster of the i-th clustering by c_{ij}. Then

$$c^* = \operatorname*{argmin}_{c_i \in C} \sum_{j=1}^{k} NormLinkRatio(c_{ij}, V - c_{ij})$$

To apply an algorithm that solves this clustering problem to our task, we construct the input graph G from the clusterings contained in the ensemble. The graph vertices V correspond to the BMM categories and the (i, j)-th entry of the matrix W is the number of ensemble members that cluster the ith and jth BMM categories together.

A low edge weight implies that a small number of ensemble members cluster together the BMM categories represented by the vertices connected by the edge. To reduce noise, we exclude edges whose weight is less than 3 from the graph.

5 Feature Representation

To complete the picture, we now describe the feature representation of the BMM categories. We create for each BMM category a vector $x \in N^{6 \cdot |S|}$ where S is the set of POS tags in the corpus. The first $2S$ features correspond to the appearance of a POS tag in the leftmost/rightmost position of a constituent labeled by the represented BMM category. Specifically, the i-th feature ($i \in \{1, \ldots, |S|\}$) is the number of times the i-th POS tag appears in the leftmost position of a constituent labeled by the represented BMM category, and the $(|S| + i + 1)$-th feature is the number of times that tag appears in the rightmost position of a constituent labeled by the BMM category.

Similarly, the next $2|S|$ features correspond to the appearance of a POS tag in the leftmost/rightmost position of a leftmost sibling of a constituent annotated by the BMM category, and the last $2|S|$ features correspond to the appearance of a POS tag in the leftmost/rightmost position of a rightmost sibling. A constituent $C1$ is defined to be the leftmost sibling of a constituent $C2$ iff $C1$ is an immediate left neighbour of $C2$, and $C1$ and $C2$ have the same parent. A rightmost sibling is defined accordingly.

Note that DP does not force a specific parametric family for the likelihood. Our decision to use a multinomial likelihood function is due to the fact that our features are counts of events.

6 Experimental Setup

Overall Setup. We evaluated our algorithm on English, German and Chinese corpora: the WSJ Penn Treebank, the Negra corpus (Brants, 1997), and version 5.0 of the Chinese Penn Treebank (Nianwen et al., 2002). In each corpus, we used the sentences of length at most 10[5], numbering 7422 (WSJ10), 7542 (NEGRA10) and 4626 (CTB10). We used the gold standard POS tag annotation of these corpora.

To initialize the clustering algorithms, we sort the BMM categories according to the number of constituents they label, and use the most frequent ones. For K-means, this provides an informed starting point, which proved crucial to the performance of the algorithm. For DPMM, although in theory the algorithm does not depend on its starting point, such initialization is helpful in practice[6].

We induce $D = 10$ different experts. For the K-means and RR08 baselines we do so by changing the number of induced clusters from 5 to 50, in steps of 5. For DPMM we use different values of α sampled log-uniformly in the range $[10^{-4}, 10]$ (5 orders of magnitude). The DP search procedure we use is that of (Daume-III, 2007)[7], whose good convergence properties have been demonstrated. The k-way normalized cut code was written by Jianbo Shi[8]. The code of RR08 was provided to us by the authors[9].

Number of Final Categories. As discussed, the quality of the different experts (clusterings) induced by the DPMM was not substantially different than RR08. Thus, despite the fact that the number of clusters for each expert is inferred automatically, we are still faced with the problem of choosing the final number of categories that will be inferred using the expert ensemble. We face the same problem when considering the baseline ensembles.

Reichart and Rappoport (2008) induced for each corpus two sets of clusters. A first set consists of T clusters, where T is the number of gold categories in the experimental corpus. For the second set size they observed that in all three corpora about 95% of the constituents are covered by 23% – 37% of the categories, and the curve rises very sharply until that 95% value. Therefore, the number of clusters in the second set is the number of categories that cover at least 95% of the constituents in the corpus (denoted by P, for *prominent* categories). Following their work, we induce for each corpus T and P categories according to the values they defined.

The specification of syntax annotation schemes, including the number of categories, usually involves arbitrary decisions (see (Klein and Manning, 2003) for an example and its effects on parsing). We thus induce for each corpus 5 different sets of clusters. Two of these are the set consisting of T clusters and the set consisting of P clusters. The other set sizes are the 3 values in $\{5, 10, \dots 25\}$ that are not the two closest values to T and P (see Table 2).

Evaluation Measures. The induced labels have arbitrary names. To evaluate them against a manually annotated corpus, a proper correspondence with the gold standard labels should be established. We explore two types of evaluation measures, one is based on mapping between the induced and gold labels and one is based on information theory (IT) concepts. All measures

[5] Excluding punctuation and null elements, as in (Klein, 2005) and other previous work.

[6] Note that this is true even when stochastic algorithms are used to infer the clusterings since convergence time *strongly* depends on the starting point.

[7] http://www.cs.utah.edu/∼hal/DPsearch

[8] http://www.cis.upenn.edu/∼jshi/

[9] We thank the authors for letting us use their code.

are based on the co-occurrence matrix between the induced and gold labels defined as follows: given a corpus tagged once with the n_1 gold standard labels and once with the n_2 induced labels, the co-occurrence matrix has $n_1 \times n_2$ and the number in the (i, j)-th entry is the number of times the i-th gold cluster and the j-th induced cluster annotate the same constituent.

We evaluate with two mapping schemes: greedy *many–to–1* and greedy *1–to–1* mappings. In both cases we find the mapping between the induced and gold clusters which maximizes the co–occurrence between the clusterings. In the first mapping two induced clusters can be mapped to the same gold standard cluster, while in the latter each and every induced cluster is assigned a unique gold cluster. Under both mapping schemes, if the number of induced clusters is lower than the number of gold clusters, there will be gold clusters to which no induced cluster is mapped. Computing the greedy 1–to–1 mapping is equivalent to finding the maximal weighted matching in a bipartite graph, whose weights are given by the co-occurrence matrix. We use the Kuhn–Munkres (Kuhn, 1955; Munkres, 1957) algorithm to solve this problem.

For these measures, we follow the previous works and apply labeled parse trees evaluation by first mapping the induced labels to the gold labels and then computing the standard labeled parsing F–score [10]. While the labeling accuracy after mapping is not explicitly given, it can computed by dividing the unlabeled F–score with the labeled F–score.

The IT based measures provides a way to evaluate the induced clustering without performing a direct mapping to the gold standard. They are based on the observation that a good clustering reduces the uncertainty of the gold standard cluster given the induced cluster and vice-versa. Several such measures exist, we use two widely–accepted ones, the **V** (Rosenberg and Hirschberg, 2007) measure and the **VI** (Meila, 2007) measure.

The V measure is defined as follows:

$$V = \frac{2hc}{h+c}$$

$$h = 1 - \frac{H(G|T)}{H(G)}, c = 1 - \frac{H(T|G)}{H(T)}$$

For the VI measure, we report its normalized version, **NVI**. NVI and VI induce the same order over clusterings but NVI values for good clusterings ranges in $[0, 1]$ (Reichart and Rappoport, 2009). The NVI measure is defined to be:

$$NVI = \frac{H(G|T) + H(T|G)}{H(G)}$$

Note that V scores are in [0, 1] and the higher the score, the better the clustering. For NVI, the scores are non-negative and lower scores imply improved clustering quality. We use e as the base of the logarithm. Many other clustering evaluation measures exist. The ones we use here are well accepted in the literature. For a recent review see (Reichart and Rappoport, 2009).

7 Results

In this section we demonstrate the effectiveness of our DPMM ensemble for the task of unsupervised induction of syntactic categories (parse tree non-terminals) for three different languages. We start by demonstrating an overall and consistent improvement over RR08 and then provide evidence that justify the specific algorithmic choices.

[10] $f = \frac{2*LP*LR}{LP+LR}$, LP and LR are labelled precision and recall.

	English, WSJ10, IT measures									
	$\|C\| = 5$		$\|C\| = 8(P)$		$\|C\| = 15$		$\|C\| = 20$		$\|C\| = 26(T)$	
	NVI	V	NVI	V	NVI	V	NVI	V	NVI	V
DP+NC	**0.98**	**0.5**	**1.02**	**0.5**	1.2	0.47	**1.22**	**0.47**	**1.3**	**0.47**
RR08	1.2	0.38	1.15	0.4	**0.89**	**0.51**	1.33	0.44	1.44	0.44

	German, NEGRA10, IT measures									
	$\|C\| = 6(P)$		$\|C\| = 10$		$\|C\| = 15$		$\|C\| = 22(T)$		$\|C\| = 25$	
	NVI	V	NVI	V	NVI	V	NVI	V	NVI	V
DP+NC	**0.85**	**0.53**	**0.87**	**0.54**	**0.94**	**0.53**	**0.92**	**0.55**	**0.98**	**0.52**
RR08	1.05	0.44	1.09	0.46	1.06	0.5	1.14	0.48	1.18	0.48

	Chinese, CTB10, IT measures									
	$\|C\| = 5$		$\|C\| = 9(P)$		$\|C\| = 15$		$\|C\| = 20$		$\|C\| = 24(T)$	
	NVI	V	NVI	V	NVI	V	NVI	V	NVI	V
DP+NC	**0.9**	**0.47**	**0.92**	**0.47**	**0.95**	**0.47**	**1**	**0.46**	**1**	**0.46**
RR08	0.96	0.44	1	0.44	1.18	0.41	1.21	0.42	1.26	0.42

	English, WSJ10, Mapping measures, (UF = 74.6)									
	$\|C\| = 5$		$\|C\| = 8(P)$		$\|C\| = 15$		$\|C\| = 20$		$\|C\| = 26(T)$	
	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)
DP+NC	**59.7**	**50.8**	**61**	**48.9**	**60.6**	42.6	**61.8**	**42.3**	**61.4**	**38**
RR08	50	42.7	49.6	42.5	55.9	**42.8**	60.5	38.9	58.9	33.2

	German, NEGRA10, Mapping measures, (UF = 58.1)									
	$\|C\| = 6(P)$		$\|C\| = 10$		$\|C\| = 15$		$\|C\| = 22(T)$		$\|C\| = 25$	
	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)
DP+NC	**44.9**	**44.6**	**44.7**	**43**	45	**41.1**	46.6	**41.1**	45.4	**40.6**
RR08	42.6	37.7	43.6	37	**48.1**	36.7	**48.1**	35.2	**48.2**	34.9

	Chinese, CTB10, Mapping measures, (UF = 51.8)									
	$\|C\| = 5$		$\|C\| = 9(P)$		$\|C\| = 15$		$\|C\| = 20$		$\|C\| = 24(T)$	
	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)	F(m:1)	F(1:1)
DP+NC	**35**	**29.8**	**35.5**	**29.8**	**35.8**	**29.6**	35.9	**28**	36	**28**
RR08	33.6	28.5	34.7	27.1	35.2	23.1	**36.2**	21.5	**36.4**	22

Table 2: Comparison between our main clustering ensemble model and the model of Reichart and Rappoport (2008) (RR08). DP: Dirichlet process. NC: normalized cut. The top three tables are for information theoretic based measures. The bottom three tables are for mapping-based measures. The columns of each table represent the specified number of induced clusters. F(m:1) and F(1:1) are labeled F-score values computed after the induced categories were mapped to the gold categories with many-to-one and 1-to-1 mappings respectively. Higher V, F(m:1) and F(1:1) and lower NVI scores imply the induced clustering to be more similar to the gold standard. The clustering ensemble model (DP+NC) provides considerable improvement over RR08.

Our Expert Ensemble Approach. We start by comparing our DPMM with normalized-cute approach (DP+NC) to RR08. Table 2 shows that DP+NC clearly outperforms the RR08 model. For IT measures it is better in 28 out of 30 experimental conditions and for F(1:1) it is better in 14 of 15 conditions. For F(m:1), DP+NC is better for English while for German and Chinese it is better when the number of induced clusters is small.

RR08 showed that their algorithm is superior to an algorithm that performs random labeling or replaces their final BMM mapping with a random mapping. Since our algorithm is shown to be superior over theirs, it is also better than these random baselines.

Note that RR08 proposed a representation vector of $3|M| + |S|$ features where M is the number of BMM categories. Our DPMM algorithm with this feature representation is very slow due to the high values of $|M|$ (between 2299 and 5559 for our experimental corpora). Consequently, we used a different representation (Section 5). To make sure that our results are not due to this change of features, we also ran their algorithm using our feature set. The superiority of our approach relative to RR08 using this setting was essentially similar to that reported above.

DPMM Clustering. To justify our selection of a DPMM for clustering we compare our results to a variant where DPMM is replaced by K-means (KM) with a cosine similarity measure. KM is known to be very sensitive to its initialization. We ran it once where the cluster centers are randomly initialized and once where, like our DPMM models, cluster centers are initialized to be the k most frequent BMM categories. Since the former model has a substantially inferior performance, we only compare to the latter. We also compare to an ensemble of experts, where each expert is a run of the BMM mapping scheme of RR08 with a different number of target clusters. For both K-means and RR08, we averaged the resulting experts using the same normalized cut algorithm used by our method.

Due to the large number of experimental conditions (3 models and 60 setups: 3 corpora, 4 measures and 5 label set sizes), for clarity of exposition, we only provide a summary of the results. For V, NVI and F(1:1), DP+NC achieves the best score in 39 out of 45 cases. For F(m:1), DP+NC is the best performing model for WSJ10. For NEGRA10 and CTB10, RR08+NC and KM+NC often provide the best performance, but DP+NC is superior to RR08 when the number of induced clusters is small.

Normalized Cut Model Averaging. To justify our selection of the normalized cut algorithm for model averaging, we experimented with various variants of our algorithm (DP+NC) and of the baseline ensemble of experts algorithm where the experts are induced by the informed KM algorithm (KM+NC). In these variants only the model averaging component (NC) is changed. We experimented with several linkage clustering algorithms (complete, single and average) and distance functions (cosine, sample correlation between observations, sample Spearman's correlation between observations and Euclidean). In these experiments each BMM category j is represented by a vector in which the value of the ith coordinate is the number of ensemble members that cluster the ith and jth categories together. We report only the results of the complete link (CL) with cosine distance that provides the best results[11].

In 52 out of the 56 cases where a clustering ensemble model outperforms RR08 NC is the algorithm that is used by the best performing model. Over all cases, DP+NC produces better clustering than DP+CL in 53 of the 60 cases.

[11] In summary, we now discuss five ensemble models. The three that were defined above: DP+NC, KM+NC, and RR08+NC, and the DP+CL and KM+CL defined here.

	No. of Constituents	1	2	3	4	5	6
NP	7898	40.4	59.76	70.4	80.3	88	90.1
VP	6758	60.18	74.77	85.35	90.2	94.24	95.96
PP	3234	44.77	45.92	63.73	71.8	79.22	85.93
S	1076	25.84	43.22	58.74	70.07	81.13	89.22
SBAR	492	34.15	47.56	60.57	72.97	80.78	87.2
ADJP	331	47.13	67.98	74.02	79.76	85.2	90.7
ADVP	226	43.8	66.8	77.88	84.96	89.82	92.48
SQ	119	45.38	85.7	96.6	98.3	100	100

Table 3: Performance of the DP+NC model on the 8 most frequent WSJ10 syntactic categories. For each gold category (lines) we show the fraction of the constituents it annotates that are labeled in the induced annotation by the k induced categories that label the most of these constituents (column headed with 'k').

Measure	V	NVI	F(m:1)	F(1:1)
English	0.034	0.12	0.013	0.117
German	0.048	0.049	0.017	0.040
Chiense	0.031	0.119	0.032	0.123

Table 4: The standard deviation to mean performance ratio of the DP+NC model when the number of induced clusters is varied. The effect of the change in the number of clusters on the F(1:1) and NVI measures is an order of magnitude larger, for English and Chinese.

The Mixture of Experts Approach. Finally, we want to demonstrate the importance of adopting an ensemble approach regardless of the specifics of the experts or the aggregation algorithm. Looking at a total of 60 experiments for corpora (3), evaluation measures (4) and number of induced clusters (5), we see a clear advantage of the ensemble approach: in 56 of the 60 cases the ensemble models substantially outperform the RR08 model, and are competitive in the other 4 experiments.

For F(m:1) the difference is up to 11.4% (WSJ10), up to 4.3% (NEGRA10) and up to 1.9% (CTB10). For V it is up to 12%, up to 9% and up to 6% for these corpora respectively. Results are even more impressive for F(1:1) and NVI. For F(1:1), improvement is up to 8.1% (WSJ10), 6.9% (NEGRA10) and 6.5% (CTB10). For NVI, the error reduction[12] is up to 18.3%, 20.2% and 20.6% respectively.

8 Qualitative Analysis

To get a better understanding of the quality of the syntactic categories induced by our model, we provide in this section a qualitative analysis of the performance of our model. We first provide a detailed error analysis of the performance of one of our models, the DP+NC model on the WSJ10 corpus when 15 categories are induced [13]. We then analyze the cross-lingual effect of an important aspect of our model – the number of induced clusters.

English Error Analysis The WSJ10 gold annotation obeys the Zipf law according to which most of the constituents (phrases) of the corpus are annotated with a small number of categories

[12] NVI values are not limited to [0,1], we thus report error reduction, computed as: $\frac{NVI_{model}-NVI_{baseline}}{NVI_{baseline}}$

[13] In order to better analyze the ability of our algorithm to detect the 'S' category, in the analysis of this section we do not count the sentence level constituent which is annotated with this category in 84.7% of the cases.

and the rest of the constituents are annotated by a larger number of much smaller categories. Concretely, while this gold annotation consists of 26 categories, 97.6% (91.8%) of the corpus constituents are labeled with the 8 (4) categories that annotate the highest number of constituents (referred to as 'the most frequent 8(4) categories').

A similar pattern is observed in the categories induced by the DP+NC model: the 8(4) most frequent categories annotate 88.5% (62.2%) of the constituents. The stronger magnitude of the Zipfian effect in the gold annotation suggests that biasing our model towards a stronger Zipfian pattern (e.g. by adding a normalization term to the NC optimization problem) may improve its performance.

Table 3 presents the distribution of each of the 8 most frequent gold categories between the 6 most frequent induced categories that annotate most of its constituents. The table shows that for 6 of these categories (all categories except from 'S' and 'SBAR') at least 40% of the constituents are annotated by the same induced category and 63.7%-96.6% of the constituents are annotated by 3 induced categories. The algorithm is shown to performs especially well in detecting the 'VP' and 'SQ' categories (60.18%-85.35% and 45.38%-96.6% of the constituents in 1-3 induced categories respectively). Performance on the 'SBAR' and 'S' categories are somewhat lower (at least in terms of overlapping with their 3 most overlapping categories).

Cross-Lingual Analysis Here we provide cross-lingual error analyse for one of the choices made by our model, the number of induced clusters. Table 4 shows the standard deviation to mean performance ratio for the DP+NC model in all three languages. While for German, all measures are relatively indifferent to the number of clusters, for Chinese and English the ratio for the F(1:1) and NVI measures is an order of magnitude larger than for F(m:1) and V.

This pattern leads us to two interesting observations that may guide future research in the field. First, a large number of clustering evaluation measures have been proposed in the literature (Reichart and Rappoport, 2009). Our experiments suggest that F(1:1) and NVI are more sensitive to a change in the number of induced clusters. Second, the performance of our model on German is mostly indifferent to the number of clusters, according to all measures. This calls for a deeper investigation of the properties of our algorithm especially with respect to languages that are typologically similar to German.

Conclusion and perspectives

We presented a novel clustering ensemble model for unsupervised induction of syntactic categories. Our model uses the Dirichlet process mixture model for expert induction and normalized-cut model averaging, providing a substantial improvement over previous works in English, German and Chinese.

Our contribution is two-fold. First, we bring the idea of ensemble learning into the task of unsupervised induction of syntactic categories, leading to substantial performance improvement. Second, and more importantly, we show how to construct a diverse ensemble of experts using the Dirichlet Process mixture model.

In future work we intend to experiment with more languages. The hierarchical generalization of the Dirichlet Process offers an opportunity for future joint learning of the syntactic structures and its annotation. On an orthogonal axis, the output of our algorithm can be used to train supervised parsers.

Acknowledgments

The work in this paper was funded by the EU grant 7FP-ITC-248064.

References

Antoniak, C. (1974). Mixture of dirichlet processes with applications to bayesian non–parametric problems. *The Annals of Statistics*, 2(6):1152–1174.

Berg-Kirkpatrick, T. and Klein, D. (2010). Phylogenetic grammar induction. In *Proc. of ACL*.

Blunsom, P. and Cohn, T. (2010). Unsupervised induction of tree substitution grammars for dependency parsing. In *Proc. of EMNLP*.

Bod, R. (2006). Unsupervised parsing with u-dop. In *Proc. of CoNLL-X*.

Borensztajn, G. and Zuidema, W. (2007). Bayesian model merging for unsupervised constituent labeling and grammar induction. In *Technical Report, ILLC*.

Brants, T. (1997). The negra export format. In *CLAUS Report, Saarland University*.

Clark, A. (2001). *Unsupervised Language Acquisition: Theory and Practice*. PhD thesis, University of Sussex.

Cohen, S. and Smith, N. (2009). Shared logistic normal distributions for soft parameter tying in unsupervised grammar induction. In *Proc. of NAACL*.

Daume-III, H. (2007). Fast search for dirichlet process mixture models. In *Proc. of AISTAT*.

Dennis, S. (2005). An exemplar-based approach to unsupervised parsing. In *Proc. of CogSci*.

Dietterich, T. (2000). Ensemble methods in machine learning. *Lecture Notes in Computer Science*, 1857:1–15.

Ferguson, T. (1973). A bayesian analysis of some non-parametric problems. *The Annals of Statistics*, 1(2):209–230.

Gildea, D. and Jurafsky, D. (2002). Automatic labeling of semantic roles. *Computational Linguistics*, 28:245–288.

Gillenwater, J., Ganchev, K., Jo∼ao V. Grac¸a, n. T., and Preira, F. (2010). Sparsity in dependency grammar induction. In *Proc. of ACL*.

Goldberg, A. (2006). *Constructions at Work*. Oxford University Press.

Goldwater, S., Griffiths, T., and Johnson, M. (2006). Interpolating between types and tokens by estimating power-law generators. In *Proc. of NIPS*.

Haghighi, A. and Klein, D. (2006). Prototype-driven grammar induction. In *Proc. of ACL*.

Haghighi, A. and Klein, D. (2007). Unsupervised coreference resolution in a nonparametric bayesian model. In *Proc. of ACL*.

Headden, W., Johnson, M., and McClosky, D. (2009). Improving unsupervised dependency parsing with richer contexts and smoothing. In *Proc. of NAACL*.

Johnson, M. and Goldwater, S. (2009). Improving nonparametric bayesian inference: Experiments on unsupervised word segmentation with adaptor grammars. In *Proc. of NIPS*.

Johnson, M., Griffiths, T., and Goldwater, S. (2007). Adaptor grammars: a framework for specifying compositional nonparametric bayesian models. In *Proc. of NIPS*.

Klein, D. (2005). *The unsupervised learning of natural language structure*. PhD thesis, Stanford University.

Klein, D. and Manning, C. (2002). A generative constituent-context model for improved grammar induction. In *Proc. of ACL*.

Klein, D. and Manning, C. (2003). Accurate unlexicalized parsing. In *Proc. of ACL*.

Klein, D. and Manning, C. (2004). Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proc. of ACL*.

Kubler, S., McDonald, R., and Nivre, J. (2009). *Dependency Parsing – Synthesis Lectures on Human Language Technologies*. Morgan and Claypool.

Kuhn, H. (1955). The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 28:245–288.

Meila, M. (2007). Comparing clustering – an information based distance. *Journal of Multivariate Analysis*, 98:873–895.

Miyao, Y., Satre, R., Sagae, K., Matsuzaki, T., and Tsujii, J. (2008). Task-oriented evaluation of syntactic parsers and their representations. In *Proc. of ACL*.

Munkres, J. (1957). Algorithms for the assignment and transportation problems. *Journal of the SIAM*, 5(1):32–38.

Nianwen, X., Chiou, F.-D., and Palmer, M. (2002). Building a large–scale annotated chinese corpus. In *Proc, of ACL*.

punyakanok, V., Roth, D., and tau Yih, W. (2008). The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 24(1):257–287.

Reichart, R. and Rappoport, A. (2008). Unsupervised induction of labeled parse trees by clustering with syntactic features. In *Proc. of COLING*.

Reichart, R. and Rappoport, A. (2009). The nvi clustering evaluation measure. In *Proc. of CoNLL*.

Rosenberg, A. and Hirschberg, J. (2007). Entropy-based external cluster evaluation measure. In *Proc. of EMNLP-CoNLL*.

Seginer, Y. (2007). Fast unsupervised incremental parsing. In *Proc. of ACL*.

Smith, N. and Eisner, J. (2006). Annealing structural bias in multilingual weighted grammar induction. In *Proc. of ACL*.

Spitkovsky, V., Alshawi, H., and Jurafsky, D. (2011a). Lateen em: Unsupervised training with multiple objectives, applied to dependency grammar induction. In *Proc. of EMNLP*.

Spitkovsky, V., Alshawi, H., Jurafsky, D., and Manning, C. D. (2010a). From baby steps to leapfrog: How less is more in unsupervised dependency parsing. In *Proc. of NAACL-HLT*.

Spitkovsky, V., Alshawi, H., Jurafsky, D., and Manning, C. D. (2010b). Viterbi training improves unsupervised dependency parsing. In *Proc. of CoNLL*.

Spitkovsky, V., Chang, A., and Jurafsky, D. (2011b). Unsupervised dependency parsing without gold part-of-speech tags. In *Proc. of EMNLP*.

Stolcke, A. (1994). *Bayesian Learning of Probabilistic Language Models*. PhD thesis, University of of California at Berkeley.

Stolcke, A. and Omohundro, S. M. (1994). Inducing probabilistic grammars by bayesian model merging. In *Grammatical Inference and Applications, Second International Colloquium*.

Yu, S. and Shi, J. (2003). Multiclass spectral clustering. In *Proc, of ICCV*.

Zhang, H. and Gildea, D. (2004). Syntax-based alignment: Supervised or unsupervised? In *Proc, of COLING*.