Refining Syntactic Categories Using Local Contexts
– Experiments in Unlexicalized PCFG Parsing –

John Pate and Detmar Meurers
The Ohio State University
Department of Linguistics

Abstract

As an extension of decades of syntactic theorizing, treebanks have inherited a small set of phrasal categories, which abstract over the environments that the categories can occur in. Extending ideas from Johnson (1998), we explore encoding information from the local tree context in each category. We then discuss two clustering techniques which preserve the distributionally relevant category distinction, forming linguistically relevant generalizations and improving PCFG parsing performance.

1 Introduction

The syntactic annotation schemes used in treebanks typically build on long traditions of linguistic investigation and analysis. For constituency-based annotation schemes such as the Penn Treebank (Taylor et al. 2003), the syntactic structure and the constituency labels are rooted in the distributional analysis of structural linguistics and generative theory, which established a range of constituency tests for determining and labelling syntactic structure. While the exact nature and reliability of constituency tests are a recurring subject of debate, the essential idea is readily apparent in the so-called substitution test: a string that is labelled with a syntactic category can be substituted with other strings bearing the same label. The result is another grammatical sentence. In the context of this paper, the relevant aspect is that substitutability does not make reference to the context in which a category occurs. A string of a given syntactic category can be replaced with another string of that category, independent of where it occurs. For example, an NP occurring as sister of a VP (i.e., a subject), can be realized in all the same ways as an NP occurring within the VP (i.e., an object). Where this turns out to be empirically incorrect, distinct categories need to be postulated for the different environments to capture the difference.

In PCFG parsing, this independence assumption is, of course, taken one step further by assuming that the different ways of realizing a given category have the same likelihood independent of the context in which that category occurs. There is a single probability distribution over the different ways to rewrite a given category,

De Smedt, K., Hajič, J. and Kübler, S. (Eds.)
Proceedings of the Sixth International Workshop
on Treebanks and Linguistic Theories (2007)

independent of the context in which it occurs. As before, when this assumption turns out to be wrong for a specific category occurring in different contexts, the conclusion is that one needs to assume distinct categories, one for each context in which its realization differs.

Sidestepping the question for which categories in which contexts additional category distinctions are warranted, Johnson (1998) explored enriching *all* nonterminal categories in a treebank with information about the context in which they occur. He enriched each category with the category label of the mother of the local tree that it occurs in and found that this significantly improves the performance of parsing with a PCFG extracted from such a treebank. Klein and Manning (2003) subsequently pursued manually distinguishing linguistically motivated distinctions, and automatic methods for dividing the traditional linguistic categories into a richer category set have resulted in some of the best PCFG parsing results for the Penn Treebank (cf. Petrov et al. 2006, and references therein). However, these methods have grown progressively more complex, typically requiring estimations over entire parse trees and specialized parsing algorithms.

The current study seeks to investigate and measure the impact of the information immediately available in the local tree context, using a plain PCFG parser. In so doing, we seek to isolate the impact of new distinctions in syntactic categories per se. We map out the maximal category space that can result from including attested local context distinctions. Then we return to the original linguistic intuition of only keeping those new distinctions which differ in their distribution, i.e., we investigate which of the categories resulting from local contextualization are distinctive enough to warrant distinguishing them categorially.

We begin by measuring the gain in parse performance obtained by contextualizing non-preterminals according not only to mother context, as in Johnson (1998), but also to local left sister and local right sister contexts, individually and in combination. Such contextualization results in a large number of syntactic categories and rules, which, as motivated above, are not necessarily useful and for which data sparsity and overfitting to the training data become an issue. We thus proceed by exploring two methods of clustering the newly created contextualized categories. Both are based on the distributional similarity of the contextualized categories as measured by the probability distribution over the local trees dominated by the contextualized categories created for a given category. The first method identifies clusters solely on the basis of the similarity of the probability distributions, while the second method identifies clusters on the basis of expected information gain.

Both clustering methods result in grammars with a dramatically reduced number of categories and rules compared to raw contextualization and some improvement in parsing results. Additionally, we show that these methods identify theoretically appealing clusters, suggesting that these methods exploit linguistic generalizations rather than statistical happenstance.

ISSN 1736-6305 Vol. 1
http://dspace.utlib.ee/dspace
/handle/10062/4476

2 Approach

2.1 General Setup

Using the standard setup for English PCFG research, we ran our experiments on the Wall Street Journal portion of the third edition of the Penn Treebank (Taylor et al. 2003). We trained on sections 2–21, used section 22 as a development set, and obtained the final parse performance numbers for section 23. We removed all grammatical function and coindexation tags, so that every nonterminal node contained only the core syntactic category information. All nodes dominating only the empty string were removed from the trees. No other transformations were performed on the corpus except for the use of contextualized categories explored in this paper and described below.

2.2 Contextualizing the categories in the training corpus

As our starting point, we contextualized all non-terminal and non-preterminal nodes according to local mother context, the local left sister context, and the local right sister context, and in all the combinations thereof. This was done by transforming the training corpus to enrich each such node with the information from the local tree it occurs in. Figure 1 exemplifies the transformation that is performed to obtain the contextualized categories encoding the mother context.

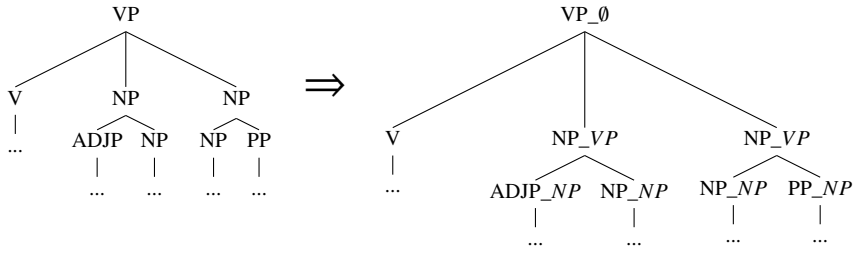

Figure 1: Encoding the local mother context in the category (à la Johnson)

The special symbol ∅ is used to encode that a particular context does not exist, such as the mother context of the VP root node in this example. Note also that the preterminal V is not contextualized, i.e., we do not enrich the set of preterminal categories since otherwise we would also need to obtain the contextualized categories for the input that is to be parsed.[1]

When contextualizing according to both mother and local right sister context, we obtain atomic categories of the form *Cat_Mother_RightSister*. Transforming the input of Figure 1 using mother and local right sister information results in the tree shown in Figure 2.

[1] Such richer lexical categories could possibly be obtained by supertagging, in the spirit of Clark and Curran (2004). Preliminary experiments including gold-standard contextualized preterminals with the input obtained F-scores in the low 90's, suggesting that a supertagging approach would be worthwhile exploring in future work.

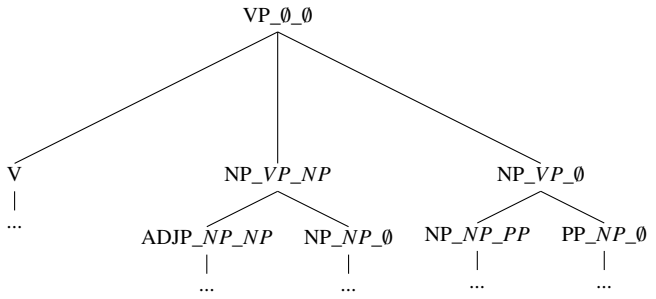

Figure 2: Result of encoding mother and right sister context in the category

Finally, when contextualizing a category with both the local mother, local left sister, and local right sister information, we make use of categories of the form *Cat_LeftSister_Mother_RightSister*. Throughout the paper, we will refer to the categories as found in the original treebank as the "original categories" and to the categories after the above-described transformation as "contextualized categories".

2.3 Grammar extraction and parsing setup

We extracted the grammars from the transformed training corpus in the usual way by counting the number of times each local tree appears.[2]

We ran our parsing experiments using BitPar (Schmid 2004), a freely available, efficient PCFG parser implementation, without lexicalization or additional components. As input, we followed the common practice of using the lexical tags from the original corpus as input, which prevents confounding the comparative contribution of local context with the accuracy of a separate part-of-speech analysis.[3]

For evaluation, we mapped the output of the parser, bearing the contextualized categories, back to the original categories simply by stripping off the contextualization suffix. We evaluated parse performance on sentences with 40 or fewer words from the original corpus in terms of labelled bracketing precision, recall, and F-score using the standard EVALB program (Sekine and Collins 1997). Failed parses were handled by assigning each word in the failed sentence the tag 'FAILED', so that the precision, recall, and F-score numbers produced by EVALB correctly reflect the performance on the complete corpus section.

2.4 Experiment 1: The space of locally contextualized categories

We first explored the contribution of the local left daughter, the local right daughter, and the local mother context, as well as combinations thereof. The parsing performance resulting for the locally-contextualized grammars is shown in Table 1.

[2]We performed no smoothing whatsoever. In particular, we did not assign non-zero counts to rules which could be constructed by inserting all potential contextualized categories for each category in each local tree. Only local trees actually occurring in the transformed training corpus were counted.

[3]For BitPar we thus used a dummy lexicon that pairs each part-of-speech tag with itself.

ISSN 1736-6305 Vol. 1
http://dspace.utlib.ee/dspace
/handle/10062/4476

	Prec.	Recall	F	Failed	Categ.	Nonce	Rules
Baseline	74.79	69.94	72.28	0%	28	1	14,974
Mother (M)	81.10	79.64	80.36	0%	300	62	22,696
Left Sister (L)	79.75	78.05	78.89	0%	648	182	32,304
Right Sister (R)	80.10	77.02	78.53	0%	523	170	26,327
L & R	80.49	80.79	80.64	0.13%	3,004	1,263	47,677
M & L	81.44	80.86	81.15	0.09%	1,905	723	38,350
M & R	82.37	82.07	82.22	0%	1,592	640	34,390
M & L & R	80.99	81.34	81.16	0.8%	5,177	2,627	52,756

Table 1: Parsing results for categories contextualized with local tree information

The table separately lists the percentage of failed parses, but these are taken into account in the precision and recall figures. The table also includes the number of categories in the training set, the number of categories occurring only once in the training set, and the number of rules, i.e., distinct local trees in the training set.

The parsing performance figures in Table 1 show that every contextualization scheme outperforms the baseline grammar, which is the grammar extracted from the original corpus, i.e., without contextualization. Moreover the two-context grammars outperform the single-context grammars. The mother-right-sister grammar, the best case, outperforms the baseline grammar in terms of F-score by 10%.

Interestingly, the three-context mother-both-sisters grammar underperforms the two-context mother-right-sister grammar. As left-sister context is beneficial in every other case, it seems that the decrease in performance is due to the data sparsity concomitant with the exploding number of both rules and categories. The mother-both-sisters grammar contains more than three times as many categories as the mother-right-sister grammar. Fully half of the categories in the mother-both-sisters grammar are observed only once in the training set, and that grammar also exhibits the most failed parses. These nonce categories are particularly alarming because they are essentially descriptions of single data instances without evidence for generalization; one thus has to expect that an equal number of specialized categories is needed for unseen data, which are missing from the grammar derived from the training corpus.

2.5 Determining which of the contextualized categories are motivated

While it is possible to introduce all category distinctions derivable from the local context, as in the results reported in the previous section, there is a clear price for introducing spurious distinctions. There only is a limited amount of training data, so the more category distinctions are introduced, the less empirical evidence is available to characterize the distributional properties of a given distinction. Rather than handle this data sparsity problem with clever smoothing techniques, we are interested in whether unmotivated distinctions can be automatically collapsed to recover a smaller, more general category set.

Returning to the original motivation for postulating contextualized categories

from the introduction, we should only use new category distinctions when a category in a particular context is not realized in the same way as in other contexts. As a measure of how different two contextualized variants of the same category are, we use the relative frequencies over the possible right-hand sides for those categories, i.e., we count in the training data how often each contextualized category immediately dominates which daughters, divided by the number of total occurrences of the contextualized category. We represent these relative frequencies as a vector with one dimension for every distinct expansion the original category can take. Note that the vectors of the contextualized categories have the same dimensionality as the vector of the original category they are derived from.

To keep only the relevant contextualized categories, we collapse distinctions between contextualized categories which have similar expansion vectors, which we assess using hierarchical clustering. Agglomerative hierarchical clustering produces a dendrogram, such as the one we will see in Figure 3 below, which expresses the similarity not only between individual items but also between groups of items.

Lee (1999) demonstrates that distributional similarity measures differ from each other in terms of their attention to the support[4] of each distribution being compared, and finds that those which focus on the intersection of the supports perform best. We require our measure to be symmetric not only due to the constraints of hierarchical clustering but also because we aim to collapse distinctions between contextualized categories which may be freely substituted for each other. Indeed, this very intersubstitutability forms the basic notion of a syntactic category. We tried both symmetric similarity measures examined by Lee, the Jensen-Shannon Divergence and the Manhattan distance, and obtained very similar results from each. This is not surprising, given that they consider the same information and performed similarly on Lee's own assessment of the measures. For felicity of exposition we present only results for the somewhat simpler Manhattan distance:

$$\text{Manhattan}\,(p, q) = \sum_{i=0}^{n} |p_i - q_i|$$

To obtain the hierarchically clustered dendrograms, we used the hclust routine from the R statistical package (R Development Core Team 2007). It proceeds using a recursive bottom-up algorithm, each step of which calculates pairwise distances between the clusters of probability vectors under consideration, and assigns the two least distant clusters to a new cluster. We use 'complete link' clustering, where the distance between two clusters is the distance between the two members most distant from each other.[5] The algorithm proceeds until only one cluster remains.

Hierarchical clustering can be an effective method for capturing linguistic generalizations, as exemplified by the dendrogram for particle ('PRT') contextualized according to mother and right sister displayed in Figure 3. There are two clear groupings that can be identified in the dendrogram. The smaller grouping, on the

[4]The support of a distribution is the set of dimensions with non-zero values.

[5]This marginally outperformed calculating cluster distance in terms of average vectors and vectors calculated directly from observed cluster member counts.

ISSN 1736-6305 Vol. 1
http://dspace.utlib.ee/dspace
/handle/10062/4476

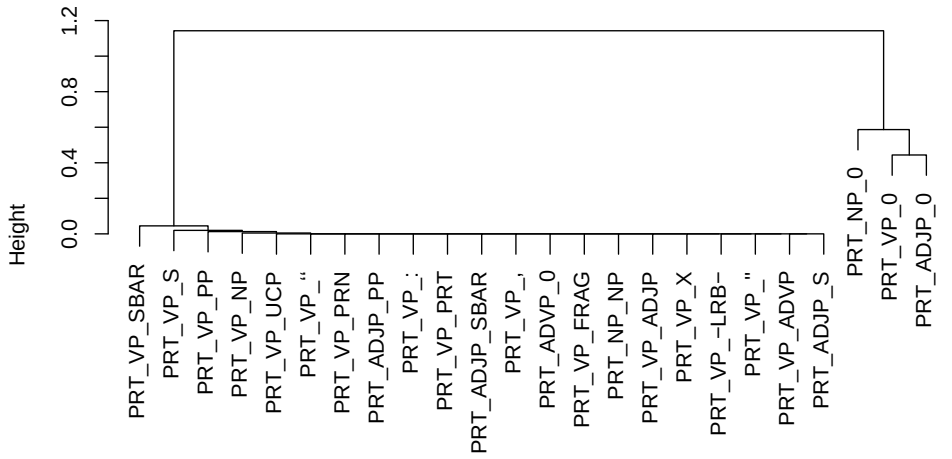

Figure 3: Dendrogram for particle (PRT) contextualized with mother-right-sister

right, consists of particles with a variety of mothers but with no local right sister, meaning that the smaller grouping corresponds to phrase-final particles, i.e., particles displaced to the end of the verb phrase or particles to verbs with no locally realized complement. The larger grouping consists of particles with a right sister (in all but one of twenty-one items) and it corresponds roughly to particles between a verb and its complements. In fact, when the left sister context is added, all three distinctions (particle with no complement, non-displaced particle, displaced particle) become evident in the larger dendrogram.

Dendrograms portray how contextualized categories and groups of contextualized categories relate to each other generally. As we want to collapse distinctions between similar contextualized categories while maintaining distinctions between different contextualized categories, we will essentially prune the dendrogram. Not contextualizing at all is equivalent to cutting the dendrogram at the root node, whereas contextualizing fully is equivalent to cutting the dendrogram at the leaf nodes. As we have seen, the former loses useful distributional information, whereas the latter gives rise to data sparsity problems compromising the reliability of the information. The intention is to prune somewhere in the middle – where exactly, and based on which criterion, is discussed further below.

2.6 Setup

We ran our clustering methods on two contextualized treebank versions. The one with mother-right-sister context categories was included because it achieved the best unclustered performance. The mother-both-sisters context version was used because it included the largest set of categories, which in principle can provide the most information (in addition to the irrelevant distinctions we want to prune away).

For the two experiments in this section, the training corpus undergoes a second transformation after the contextualization step, in which the contextualization

annotation is replaced with the label of the appropriate cluster. Contextualized categories assigned to singleton clusters naturally can keep their names as these names are straightforward atomic symbols. For example, assume that the clustering procedure assigns NP_*NP*_∅ and NP_*NP_PP* to a cluster NP_3, and that the other contextualized categories in Figure 2 are assigned to singleton clusters, the tree of Figure 2 would be transformed to the one in Figure 4.

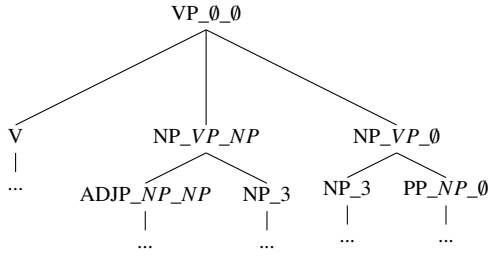

Figure 4: Tree with mother and right sister context categories after clustering

2.7 Experiment 2: Clustering according to Dendrogram Height

Remember that each iteration of the hierarchical clustering algorithm calculates pairwise distances between all the vectors to be clustered and merges the two least distant vectors. The dendrogram records the distance at which each pair of vectors was merged in the height of the node representing the merge, displayed in the y-axis of the example dendrogram in Figure 3. The smaller the height, the more similar the probability distribution over the expansions for the two (sets of) contextualized categories – and the less we want to keep that category distinction.

Our first method thus simply defines clusters to be the largest sub-dendrograms whose merge height is less than some cut-off value. Using the development set, we found that a cut-off value of 0.7 worked best for mother-right-sister context and that a cut-off value of 1 worked best for mother-both-sisters context. The performance of these height-clustered grammars on the test set is shown in Table 2.

	Precision	Recall	F	Failed	Categ.	Nonce	Rules
Baseline	74.79	69.94	72.28	0%	28	1	14,974
Mother and Right Sister							
Unclustered	82.37	82.07	82.22	0%	1,592	640	34,390
Height-Clust.	82.13	81.83	81.98	0%	849	224	29,259
KLD-Clust.	82.35	81.41	81.88	0%	215	21	24,392
Mother and Both Sisters							
Unclustered	80.99	81.34	81.16	0.8%	5,177	2,627	52,756
Height-Clust.	82.17	82.57	82.37	0%	1,672	556	33,628
KLD-Clust.	82.32	82.24	82.28	0%	495	87	28,781

Table 2: Parsing results for unclustered and clustered contextualized categories

We see that in both Mother and Right Sister as well as Mother and Both Sisters

ISSN 1736-6305 Vol. 1
http://dspace.utlib.ee/dspace
/handle/10062/4476

contextualization schemes, clustering according to dendrogram height effectively decreases the number of syntactic categories, nonce categories, and distinct rules. However, we see a benefit in parse performance only in the case of mother-both-sisters context. The improvement confirms the conjecture that the earlier underperformance of mother-both-sisters' context was due to data sparsity.

2.8 Experiment 3: Clustering according to expected information gain

The dendrogram expresses distance relationships between the contextualized expansion vectors without reference to the expansion vector of the original category. The above clustering method, then, identifies clusters consisting of contextualized categories which are similar to each other – but not necessarily different from the original category. We thus tried a second clustering method, which instead identifies sub-dendrograms which diverge from the original category. We evaluate the divergence in terms of the Kullback-Leibler Divergence (KLD)[6] of the expansion distribution from that of the original category:

$$\mathrm{KLD}(p,q) = \sum_{i=0}^{n} p_i \cdot log_2\left(\frac{p_i}{q_i}\right)$$

The Kullback-Leibler divergence of one probability distribution p from another q expresses the amount of information lost, in bits, by using q to encode the behavior of p. In our terms, then, this sum represents the information lost by ignoring that a particular syntactic node appears with a particular class of contexts.[7]

The KLD-based method described here involves pruning the same dendrogram as in experiment 2. But this time the cut-off values are in terms of the Kullback-Leibler Divergence of the expansion vector at the dendrogram node from that of the overall category. Any sub-dendrogram which exceeds the KLD cut-off value is assigned to its own cluster.

The underlying dendrogram is derived using the same pairwise Manhattan-distance computation as before since the Kullback Leibler Divergence is not a replacement for the pairwise distance measure. It is not symmetric and it does not satisfy the triangle inequality[8]. Note that this method evaluates only the subset of the powerset of the contextualized categories provided by the dendrogram. There may be a member of that powerset which would obtain a large KLD but is not itself made available by the dendrogram because it does not consist of contextualized categories deemed maximally intersubstitutable by the Manhattan distance. Our particular method creates the largest possible clusters given that dendrogram by beginning at the root node of the dendrogram and descending until an excess of the cut-off value is encountered. The KLD of successive sub-dendrograms does not necessarily increase monotonically, and beginning from the leaves and moving up would likely result in more clusters with fewer members. The representative vector

[6] An alternative choice would be Lee (1999)'s α-skew divergence, which is essentially a smoothed version of the Kullback-Leibler divergence, and thus in our context only adds orthogonal complexity.

[7] Since many dimensions of the category vectors are 0, one frequently relies on $0 \cdot log(0) = 0$.

[8] In other words, $\mathrm{KLD}(a,b) + \mathrm{KLD}(b,c) \geq \mathrm{KLD}(a,c)$ does not necessarily hold.

used to compute the KLD for a sub-dendrogram is recalculated directly from the counts observed for each member of the sub-dendrogram.

Proceeding top-down in the dendrogram, it is possible to descend all the way to the leaves without obtaining a large Kullback-Leibler divergence from the original category's expansion vector. As these leaves have been determined to be similar to each other by the dendrogram and non-divergent from the original category, we assign each hitherto unclustered sub-dendrogram to be its own cluster.[9] In sum, the method used in this experiment collapses distinctions between expansion vectors which are both similar to each other and indistinct from the norm.

Using the development set, we found that a cut-off of 1 worked best for mother-right-sister context and that a cut-off of 2 worked best for mother-both-sisters context. The parse performance results for the test set are included in Table 2, displayed in the previous section. We see that the performance of the KLD-clustering grammars is almost identical to that of the height-clustering grammars. However, the KLD-clustering grammars exhibit dramatically fewer categories, nonce categories, and unique rules than do the height-clustering grammars.

2.9 Comparing clustering methods

One way to assess and compare these clustering methods is to look at the number of clusters produced for each original category. Table 3 displays such data for the grammar contextualized with mother and both sisters.

For each original category, it lists the number of subcategories created by raw contextualization, after height-clustering, and after KLD-clustering along with the ratio of clustered subcategories to contextualized subcategories within each clustering method as a percent. We observe great variability in the number of distinct contexts that original categories appear in, ranging from just eight contexts in the case of WHADJP to 1,043 in the case of NP. *Wh*-phrases appear in a substantially smaller number of contexts than do their non-*wh*-counterparts. Although the number of contextualized subcategories obviously limits the number of clustered subcategories, the number of contextualized categories does not necessarily predict the number of clustered categories. For example, PRT (Particle) appears in 93 distinct contexts whereas UCP (Unlike Coordinate Phrase) appears in only 72, but PRT is consolidated into 5 clusters (height) or 2 clusters (KLD), whereas UCP appears significantly more diverse with 63 (height) and 27 (KLD) clusters.

Both clustering methods rely on pulling out sub-dendrograms on the basis of some cut-off value, which we crudely optimized by trying values to maximize the F-score on the development set. However, the significant variability apparent from

[9] We could alternatively assign all non-divergent contextualized categories to the same cluster, essentially using the dendrogram only to identify divergent clusters. This method performed poorly on the development set, suggesting that sub-dendrograms whose expansion vectors do not diverge from the original category's expansion vector do diverge from each other significantly. This is reasonable, as distinct sub-dendrograms are distinct precisely because the hierarchical clustering algorithm found them to be different from each other.

ISSN 1736-6305 Vol. 1
http://dspace.utlib.ee/dspace
/handle/10062/4476

Original	Ctxt'd	Clustered Height	Clustered KLD	Original	Ctxt'd	Clustered Height	Clustered KLD
ADJP	465	178 (38%)	52 (11%)	S	457	52 (11%)	66 (14%)
ADVP	601	107 (18%)	21 (3%)	SBAR	360	48 (13%)	28 (8%)
CONJP	46	9 (20%)	2 (4%)	SBARQ	37	18 (49%)	9 (24%)
FRAG	78	59 (76%)	23 (29%)	SINV	46	32 (70%)	2 (4%)
INTJ	44	14 (32%)	4 (9%)	SQ	52	39 (75%)	7 (13%)
LST	13	6 (46%)	1 (8%)	UCP	72	63 (88%)	27 (38%)
NAC	51	19 (37%)	7 (14%)	VP	319	202 (63%)	24 (8%)
NP	1043	474 (45%)	76 (7%)	WHADJP	8	3 (38%)	1 (13%)
NX	59	37 (63%)	18 (31%)	WHADVP	43	3 (7%)	3 (7%)
PP	734	51 (7%)	23 (3%)	WHNP	54	16 (30%)	8 (15%)
PRN	325	148 (46%)	66 (20%)	WHPP	13	3 (23%)	1 (8%)
PRT	93	5 (5%)	2 (2%)	X	57	32 (56%)	16 (28%)
QP	94	45 (48%)	4 (4%)	ROOT	1	1 (100%)	1 (100%)
RRC	11	7 (64%)	2 (18%)	PRT|ADVP	1	1 (100%)	1 (100%)
				TOTALS	5,177	1,672	495

Table 3: Breakdown by original category for the mother-both-sisters grammar

the subcluster counts in Table 3 suggests that a single cut-off value, while optimized for the contextualization scheme as a whole, is not necessarily optimal for each original category. Indeed, in the discussion of PRT dendrograms in section 2.5 we mentioned that three theoretically-appealing subclusters are readily apparent from the mother-both-sisters dendrogram. But neither method successfully identifies exactly three. Height-clustering creates 5 clusters, forming one cluster each for two of the groups, but splitting the third into three singleton subcategories. KLD-clustering obtains two clusters, conflating into one subcategory the first two groups distinguished by height-clustering, but appealingly clustering into one subcategory the remaining three 'missed' by height-clustering. An approach which optimizes the cut-off values for each original category might improve clustering further.

3 Summary and Outlook

Extending ideas from Johnson (1998), we explored enriching the syntactic category distinctions in the Penn Treebank with the contextual information available within the local tree. PCFG parsing experiments using a grammar extracted from the enriched corpus confirm that significant information about the expansion properties of syntactic categories is immediately available in the local context.

At the same time, blindly introducing all contextually possible category distinctions results in well-known data sparsity issues, with many of the possible categories only rarely or never occurring in the training data. We therefore explored introducing only those contextualized categories which are distributionally distinct, as measured by the probability distribution over the local expansions. We showed that clustering can identify theoretically appealing clusters, automatically exploiting linguistic generalizations. Clustering based on the distance between contextu-

alized categories was equally effective as clustering based on the divergence of a contextualized category from the original category, even though the latter resulted in fewer clusters.

The relevance of the local context for defining or enriching the category set can be seen as an interesting reflection of the role of such context frames in the acquisition of categories during language acquisition (cf., e.g., Mintz 2003).

In future work, we intend to explore whether the approach can help identify distributionally relevant category information from outside the domain of syntax per se. For example, we are investigating whether the method can identify distributionally valuable category distinctions based on spoken language input that includes intonational phrasing and other prosodic information.

Acknowledgements We would like to thank Adriane Boyd for her kind support and Markus Dickinson and the TLT reviewers for their useful comments.

References

Clark, S. and J. R. Curran (2004). The Importance of Supertagging for Wide-Coverage CCG Parsing. In *Proceedings of COLING'04*. Geneva, Switzerland.

Johnson, M. (1998). PCFG Models of Linguistic Tree Representations. *Computational Linguistics* 24(4), 613–632.

Klein, D. and C. Manning (2003). Accurate Unlexicalized Parsing. In *Proceedings of the 41st ACL '03*.

Lee, L. (1999). Measures of distributional similarity. In *Proceedings of the 37th ACL'99*. pp. 25–32.

Mintz, T. H. (2003). Frequent frames as a cue for grammatical categories in child directed speech. *Cognition* 90, 91–117.

Petrov, S., L. Barrett, R. Thibaux and D. Klein (2006). Learning Accurate, Compact, and Interpretable Tree Annotation. In *Proceedings of COLING/ACL'06*. Sydney, Australia, pp. 433–440.

R Development Core Team (2007). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Schmid, H. (2004). Efficient Parsing of Highly Ambiguous Context-Free Grammars with Bit Vectors. In *Proceedings of COLING'04*. Geneva, Switzerland.

Sekine, S. and M. J. Collins (1997). EVALB bracket scoring program. web page, `http://nlp.cs.nyu.edu/evalb/`.

Taylor, A., M. Marcus and B. Santorini (2003). The Penn Treebank: An Overview. In A. Abeillé (ed.), *Treebanks: Building and using syntactically annotated corpora*, Dordrecht: Kluwer, chap. 1, pp. 5–22.

ISSN 1736-6305 Vol. 1
http://dspace.utlib.ee/dspace
/handle/10062/4476