In: *Proceedings of CoNLL-2000 and LLL-2000*, pages 49–54, Lisbon, Portugal, 2000.

Overfitting Avoidance for Stochastic Modeling of Attribute-Value Grammars

Tony Mullen
Alfa-Informatica
University of Groningen
mullen@let.rug.nl

Miles Osborne
Division of Informatics
University of Edinburgh
osborne@cogsci.ed.ac.uk

Abstract

We present a novel approach to the problem of overfitting in the training of stochastic models for selecting parses generated by attribute-valued grammars. In this approach, statistical features are *merged* according to the frequency of linguistic elements within the features. The resulting models are more general than the original models, and contain fewer parameters. Empirical results from the task of parse selection suggest that the improvement in performance over repeated iterations of iterative scaling is more reliable with such generalized models than with ungeneralized models.

1 Introduction

The maximum entropy technique of statistical modeling using random fields has proved to be an effective way of dealing with a variety of linguistic phenomena, in particular where modeling of attribute-valued grammars (AVG's) is concerned (Abney, 1997). This is largely because its capacity for considering overlapping information sources allows the most to be made of situations where data is sparse. Nevertheless, it is important that the statistical features employed be appropriate to the job. If the information contributed by the features is too specific to the training data, *overfitting* becomes a problem (Chen and Rosenfeld, 1999; Osborne, 2000). In this event, a peak in model performance will be reached early on, and continued training yields progressive deterioration in performance. From a theoretical standpoint, overfitting indicates that the model distribution is unrepresentative of the actual probabilities. In practice, it makes the performance of the model dependent upon early stopping of training. The point at which this must be done is not always reliably predictable.

This paper describes an approach to feature selection for maximum entropy models which reduces the effects of overfitting. Candidate features are built up from basic grammatical elements found in the corpus. This "compositional" quality of the features is exploited for the purpose of overfitting reduction by means of *feature merging*. In this process, features which are similar to each other, save for certain elements, are merged; i.e, their disjunction is considered as a feature in itself, thus reducing the number of features in the model. The motivation behind this methodology is similar to that behind that of Kohavi and John (1997), but rather than seeking a proper subset of the candidate feature set, the merging procedure attempts to compress the feature set, diminishing both noise and redundancy. The method differs from a simple feature cutoff, such as that described in Ratnaparkhi (1998), in that the feature cutoff eliminates statistical features directly, whereas the merging procedure attempts to generalize them. The method employed here also derives inspiration from the notion of *Bayesian model merging* introduced by Stolcke and Omohundro (1994).

Section 2 describes parse selection and discusses the "compositional" statistical features employed in a maximum entropy approach to the task. Section 3 introduces the notion of *feature merging* and discusses its relationship with overfitting reduction. Sections 4 and 5 describe the experimental models built and the results of merging on their performance. Finally, section 6 sums up briefly and indicates some further directions for inquiry on the subject.

2 Maximum entropy-based parse selection

The task of *parse selection* involves selecting the best possible parse for a sentence from a set of possible parses produced by an AVG. In the present approach, parses are ranked according to their goodness by a statistical model built using the *maximum entropy technique*, which involves building a distribution over events which is the most uniform possible, given constraints derived from training data. Events are composed of *features*, the fundamental statistical units whose distribution is modeled. The model is characterized by constraints upon the distributions of features, derived from the features' empirical frequencies. An untrained (thus unconstrained) max ent model is by definition characterized by the uniform distribution. The constraints which characterize the model are expressed as weights on individual features. Training the model involves deriving the best weights from the training data by means of an algorithm such as *Improved Iterative Scaling* (IIS) (Della Pietra et al., 1995).

IIS assigns weights to features which reflect their distribution and significance. With each iteration, these weights reflect the empirical distribution of the features in the training data with increasing accuracy. In ideal circumstances, where the distribution of features in the training data accurately represents the true probability of the features, the performance of the model should increase asymptotically with each iteration of training until it eventually converges. If the training data is corrupt, or noisy, or if it contains features which are too sparsely distributed to accurately represent their probability, then overfitting arises.

2.1 The structure of the features

The statistical features used for parse selection should contain information pertinent to sentence structure, as it is the information encoded in these features which will be brought to bear in prefering one parse over another. Information regarding constituent heads, POS tags, and lexical information is pertinent, as is information on constituent ordering and other grammatical information present in the data. Most or all of these factors are considered in some form or another by current state-of-the-art statistical parsers such as those of Charniak (1997), Magerman (1995) and Collins (1996).

In the present approach, each feature in the feature set corresponds to a depth-one tree structure in the data, i.e. a mother node and all of its daughters. Within this general structure various *schemata* may be used to derive actual features, where the information about each node employed in the feature is determined by which schema is used. For example, one schema might call for POS information from all nodes and lexical information only from head nodes. Another might call for lexical information only from nodes which also contain the POS tag for prepositions. The term *compositional* is used in this context to describe features built up according to some such schema from basic linguistic elements such as these. Thus each compositional feature is an ordered sequence of elements, where the order reflects the position in the tree of the elements. Instantiations of these schemata in the data are used as the statistical features. The first step is to run a given schema over the data, collecting a set of features. The next step is to characterize all events in the data in terms of those features.

This general structure for features allows considerable versatility; models of widely varying quality may be constructed. This structure for statistical features might be compared with the Data-Oriented Parsing (DOP) of Bod (1998) in that it considers subtrees of parses as the structural units from which statistical information is taken. The present approach differs sharply from DOP in that its trees are limited to a depth of one node below the mother and, more importantly, in the fact that the maximum entropy framework allows modeling without the independence assumptions made in DOP.

Since maximum entropy allows for overlapping information sources, features derived using different schemata (that is, collecting different pieces of node-specific information) may be collected from the same subtrees, and used simultaneously in a single model.

3 Feature merging and overfitting reduction

The idea behind feature merging is to reduce overfitting through changes made directly to the model. This is done by combining highly

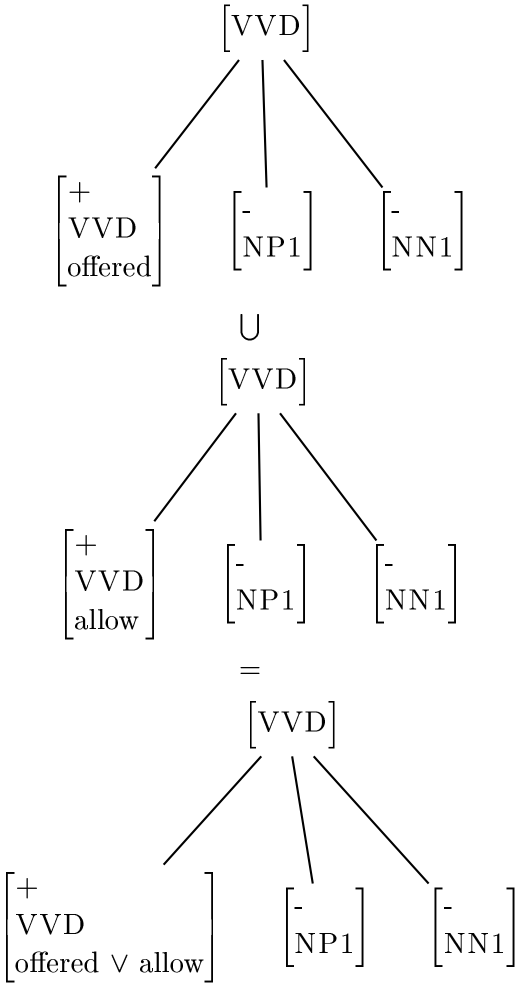

Figure 1: An example of feature merging. The top two features are merged in the form of the bottom feature, where the lexical elements have been replaced by their disjunction. The merged feature represents the union of the sets of tokens described by the unmerged feature types. All instances of the original two features would now be replaced in the data by the merged feature.

specific features which occur rarely to produce more general features which occur more often, resulting in fewer total features used. Even if the events are not noisy or inaccurate in actual fact, they may still contribute to overfitting if their features occur too infrequently in the data to give accurate frequencies. The merging procedure seeks to address overfitting at the level of the features themselves and remain true to the spirit of the maximum entropy approach, which seeks to represent what is unknown about the data with uniformity of the distribution, rather than by making adjustments on the model distribution itself, such as the Gaussian prior of Osborne (2000).

Each feature, as described above, is made up of discrete elements, which may include such objects as lexical items, POS tags, and grammatical attribute information, depending on the schema being used. The rarity of the feature in the data is largely—although not entirely—determined by the rarity of elements within it. In the present merging scheme, a set of elements is collected whose empirical frequencies are below some predetermined cutoff point. Note that the use of the term "cutoff" here refers to the empirical frequency of *elements* of features rather than of features themselves, as in Ratnaparkhi (1998). All features containing elements in this set will be altered such that the cutoff element is replaced by a uniform disjunctive element, effectively merging all similarly structured features into one, with the disparate elements replaced by the disjunctive element. An example may be seen in figure 1, where the union of the two features at top of the figure is represented as the feature below them. The merged elements in this case are the lexical items *offered* and *allow*. Such a merge would take place on the condition that the empirical frequencies of both elements are below a certain cutoff point. If so, the elements are replaced by a new element representing the disjunction of the original elements, creating a single feature. This feature then replaces *all instances* of both of the original features. If both of the original features appear once each together in an event, then two instances of the merged feature will appear in that event in the new model.

4 Experiments

The experiments described here were conducted using the Wall Street Journal Penn Treebank corpus (Marcus et al., 1993). The grammar used was a manually written broad coverage DCG style grammar (Briscoe and Carroll, 1997). Parses of WSJ sentences produced by the grammar were ranked empirically using the treebank parse as a gold standard according to a weighted linear combination of crossing brackets, precision, and recall. If more than fifty parses were produced for a sentence, the best fifty were used and the rest discarded. For the training data, the empirical rankings of all parses for each sentence were normalized so the total parse scores for each sentence added to a constant. The events of the training data consisted of parses and their corresponding normalized score. These scores were furthermore treated as frequencies. Thus, high ranked parses would be treated as events occurring more frequently in the training data, and low ranked parses would be treated as occurring rarely.

The features of the unmerged model consisted of depth-one trees carrying node information according to the following schema: the POS tag of the mother, POS tags of all daughters ordered left to right, HEAD+ information for the head daughter, and lexical information for all daughters carrying a verbal or prepositional POS tag. The features themselves were culled using this schema on 2290 sentences from the training data. The feature set consisted of 38,056 features in total, of which 6561 were active in the model (assigned non-zero weights) after the final iteration of IIS. Two models using this feature set were trained, one on only 498 training sentences, a subset of the 2290 sentences used to collect the features, and the other on nearly ten times that number, 4600 training sentences, a superset of the same set of sentences.

Several merged models were made based on each of these unmerged models, using various cutoff numbers. Cutoffs were set at empirical frequencies of 100, 500, 1000, 1250, and 1500 elements. For each model merge, all elements which occurred in the training data fewer times than the cutoff number were replaced in each feature they appeared in by the uniform disjunctive element, and the merged features then took the place of the unmerged features.

Iterative scaling was performed for 150 iterations on each model. This number was chosen arbitrarily as a generous but not gratuitous number of iterations, allowing general trends to be observed.

The models were tested on approximately 5,000 unseen sentences from other parts of the corpus. The performance of each model was measured at each iteration by binary best match. The model chose a single top parse and if this parse's empirical rank was the highest (or equal to the highest) of all the parses for the sentence, the model was awarded a point for the match, otherwise the model was awarded zero. The performance rating reflects the percentage of times that the model chose the best parse of all possible parses, averaged over all test sentences.

5 Results

5.1 Performance of unmerged models

Of the unmerged models, as expected, the one trained on the smaller set shows the worst performance and most drastic overfitting. Its peak at approximately 42.5% performance comes early, at around 20 iterations of IIS, and subsequently drops to 40.5% at around 50 iterations. At around 80 iterations, it plunges to about 39%, where it remains. This model's performance may be seen in figure 2 represented by the solid black line.

In figure 3, the solid black line represents the original model trained on 4600 sentences. The feature set is the same, although in this case all of the 38,057 features are active. The advantage of having so much more training data is evident. The performance peaks at a much higher level and overfitting, although present, is much less drastic at the end of 150 iterations. Nevertheless, the curve still reaches a maximum point fairly early on, at about 40 iterations, and the performance diminishes from there.

5.2 Performance of merged models

Different cutoffs yielded varying degrees of improvement. A cutoff of 100 elements seemed to make no meaningful difference either way with either model. Increasing the cutoff for the 498 sentence-trained model both lowered the peak before 40 iterations and raised the dip after 80 in a fairly regular fashion. The best balance seemed to be struck with a cutoff of 1250. In this case the number of active features in the model was reduced to 4801.

As can be seen from figure 2, the merged model, represented by the dotted line, shows a much more predictable improvement, its curve much closer to the optimal asymptotic improvement. In terms of actual performance, the early peak of the unmerged model is not present at all in the merged model, which catches up between around 40 and 80 iterations. After 80 iterations, the merged model begins to outperform the unmerged model, which has begun to suffer from

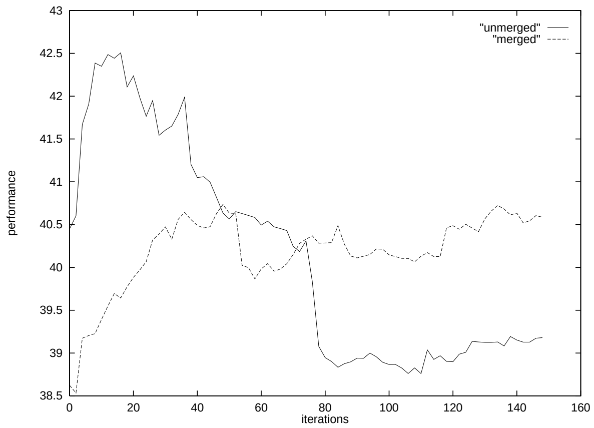

Figure 2: For the model trained on 498 sentences, features containing elements appearing fewer than 1250 times are merged. The early peak of the unmerged model gives way to drastic overfitting. The merged model, on the other hand, does not reach this peak, but overfitting is not present.

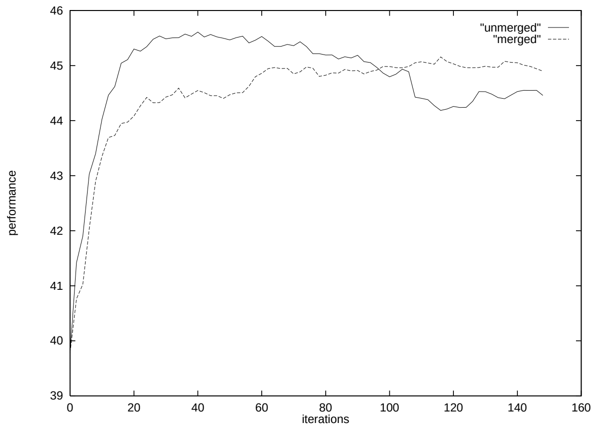

Figure 3: For the model trained on 4600 sentences, features containing elements appearing fewer than 500 times are merged. The overfitting in the unmerged model, represented by the solid line, is less drastic due to more extensive training material, but an improvement can still be seen in the curve of the merged model.

severe overfitting. The merged model, on the other hand, shows no evidence of overfitting.

Likewise, the merged model represented by the dotted line in figure 3 shows no overfitting either, an improvement in that regard over its unmerged counterpart. For this model, the best cutoff of those tried appeared to be 500, and the number of active features was reduced to 77,286. Higher cutoffs led to slower rates of improvement and lower levels of performance.

Both merging operations may be viewed as yielding improvements over the unmerged models, as the accuracy of the model should ideally increase with each iteration of the IIS algorithm until it converges. It is likely that further iterations would yield even more clear improvement, although it is also possible that the merged models themselves would begin to exhibit overfitting after some point. The rate of increase in performance and the point of onset of overfitting varies from model to model. In general, predictable improvement, even if gradual, is preferable to sporadic peaking and drastic overfitting. This may not always be the case in practice.

6 Conclusion

The feature merging strategy described in this paper may be employed to reduce overfitting in situations where statistical features are built up compositionally from basic elements. As mentioned, the merging strategy bears certain similarities with other methods of overfitting reduction, such as standard feature cutoffs where entire features appearing less than some number of times are ignored (Ratnaparkhi, 1998). Intuitively, it seems that in a sparse data situation, it would be beneficial to retain the general information in features, rather than ignoring rare features entirely. It would be worthwhile to verify this suspicion by comparing the present approach directly with a simple feature cutoff, and furthermore comparing a simple cutoff to one where the low-frequency features were merged according to the present scheme, rather than simply discarded. It is to be expected that a combination of both approaches would be likely to outperform either individual approach. How much improvement may be gained remains to be seen.

References

Steven P. Abney. 1997. Stochastic attribute-value grammars. *Computational Linguistics*, 23(4):597–618, December.

Rens Bod and Ronald M. Kaplan. 1998. A probabilistic corpus-driven model for lexical-functional analysis. In *Proceedings of ACL/COLING '98*, Montreal.

Ted Briscoe and John Carroll. 1997. Automatic extraction of subcategorization from corpora. In *Proceedings of the* 5^{th} *Conference on Applied NLP*, pages 356–363, Washington, DC.

Eugene Charniak. 1997. Statistical parsing with a context-free grammar and word statistics. Technical Report CS-95-28, Department of Computer Science, Brown University.

Stanley F. Chen and Ronald Rosenfeld. 1999. A gaussian prior for smoothing maximum entropy models. Technical Report CMU-CS-99-108, Carnegie Mellon University, School of Computer Science.

Michael John Collins. 1996. A new statistical parser based on bigram lexical dependencies. In Arivind Joshi and Martha Palmer, editors, *Proceedings of the 34th Annual Meeting of the ACL*, pages 184–191, San Francisco. Association for Computational Linguistics, Morgan Kaufmann Publishers.

Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. 1995. Inducing features of random fields. Technical Report CS-95-144, Carnegie Mellon University, School of Computer Science.

Ron Kohavi and George H. John. 1997. Wrappers for feature subset selection. *Artificial Intelligence: special issue on relevance*, 97:273–324.

David M. Magerman. 1995. Statistical decision-tree models for parsing. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 276–283.

M.P. Marcus, B. Santorini, and M.A. Marcinkievicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19:313–330.

Miles Osborne. 2000. Estimation of stochastic attribute-value grammars using an informative sample. In *Proceedings of Coling 2000*, Saarbrücken.

Adwait Ratnaparkhi. 1998. *Maximum Entropy Models for Natural Language Ambiguity Resolution*. Ph.D. thesis, University of Pennsylvania.

Andreas Stolcke and Stephen Omohundro. 1994. Inducing probabilistic grammars by bayesian model merging. In R.C. Carrasco and J. Oncina, editors, *Grammatical Inference and Applications*, pages 106–118. Springer.