k-NN for Local Probability Estimation in Generative Parsing Models

Deirdre Hogan
Department of Computer Science
Trinity College Dublin
Dublin 2, Ireland
dhogan@cs.tcd.ie

Abstract

We describe a history-based generative parsing model which uses a *k*-nearest neighbour (*k*-NN) technique to estimate the model's parameters. Taking the output of a base *n*-best parser we use our model to re-estimate the log probability of each parse tree in the *n*-best list for sentences from the Penn Wall Street Journal treebank. By further decomposing the local probability distributions of the base model, enriching the set of conditioning features used to estimate the model's parameters, and using *k*-NN as opposed to the Witten-Bell estimation of the base model, we achieve an *f*-score of 89.2%, representing a 4% relative decrease in *f*-score error over the 1-best output of the base parser.

1 Introduction

This paper describes a generative probabilistic model for parsing, based on Collins (1999), which re-estimates the probability of each parse generated by an initial base parser (Bikel, 2004) using memory-based techniques to estimate local probabilities.

We used Bikel's re-implementation of the Collins parser (Bikel, 2004) to produce the *n*-best parses of sentences from the Penn treebank. We then recalculated the probability of each parse tree using a probabilistic model very similar to Collins (1999) Model 1. In addition to the local estimation technique used, our model differs from Collins (1999) Model 1 in that we extend the feature sets used to predict parse structure to include more features from the parse history, and we further decompose some of the model's parameter classes.

2 Constraint Features for Training Set Restriction

We use the same k-NN estimation technique as Toutonava et al (2003) however we also found that restricting the number of examples in the training set used in a particular parameter estimation helped both in terms of accuracy and speed. We restricted the training sets by making use of constraint features whereby the training set is restricted to only those examples which have the same value for the constraint feature as the query instance.

We carried out experiments using different sets of constraint features, some more restrictive than others. The mechanism we used is as follows: if the number of examples in the training set, retrieved using a particular set of constraint features, exceeds a certain threshold value then use a higher level of restriction i.e. one which uses more constraint features. If, using the higher level of restriction, the number of samples in the training set falls below a minimum threshold value then "back-off" to the less restricted set of training samples.

3 Experiments

Our model is trained on sections 2 to 21 inclusive of the Penn WSJ treebank and tested on section 23. We used sections 0, 1, 22 and 24 for validation.

We re-estimated the probability of each parse using our own baseline model, which is a replication of Collins Model 1. We tested *k*-NN estimation on the head generation parameter class

Proceedings of the Ninth International Workshop on Parsing Technologies (IWPT), pages 202–203,
Vancouver, October 2005.

and the parameter classes for generating modifying nonterminals. We further decomposed the two modifying nonterminal parameter classes. Table 1 outlines the parameter classes estimated using *k*-NN in the final model settings and shows the feature sets used for each parameter class as well as the constraint feature settings.

Parameter Class	**History**	**Contraint Features**
$P(C_H \mid \ldots)$	C_p, C_H, w_p, t_p, t_{gp}	$\{C_p\}$
$P(t_i \mid \ldots)$	dir, C_p, C_H, w_p, t_p, dist, t_{i-1}, t_{i-2}, C_{gp}	$\{dir, C_p\}$, $\{dir, C_p, C_H\}$
$P(C_i \mid \ldots)$	dir, t_i, C_p C_H, w_p, t_p, dist, t_{i-1},t_{i-2}, C_{gp}	{dir,ti},$\{dir, t_i, C_p\}$
$P(coord,punc \mid \ldots)$	dir, C_i, t_i, C_p, C_H, w_p, ,t_p	$\{dir, C_i, t_i\}$
$P(C_i\ t_i \mid C_p = NPB \ldots)$	dir, C_H, w_p, C_{i-2}, w_{i-2}, C_{i-3}, w_{i-3}, C_{gp}, C_{ggp}, C_{gggp}	$\{dir, C_H\}$
$P(punc \mid C_p = NPB \ldots)$	dir, t_i, C_i, C_H, w_p,t_p, t_{i-2}, t_{i-3}	$\{dir, t_i\}$

Table 1: The parameter classes estimated using *k*-NN in the final model. C_H is the head child label, C_p the parent constituent label, w_p the head word, t_p the head part-of-speech (POS) tag. C_i, w_i and t_i are the modifier's label, head word and head POS tag. t_{gp} is the grand-parent POS tag, C_{gp}, C_{ggp}, C_{gggp} are the labels of the grand-parent, great-grandparent and great-great-grandparent nodes. *dir* is a flag which indicates whether the modifier being generated is to the left or the right of the head child. *dist* is the distance metric used in the Collins parser. *coord*, *punc* are the coordination and punctuation flags. NPB stands for base noun phrase.

We extend the original feature sets by increasing the order of both horizontal and vertical markovization. From each constituent node in the vertical or horizontal history we chose features from among the constituent's nonterminal label, its head word and the head word's part-of-speech tag. We found for all parameter classes $k = 10{,}000$ or $k = 20{,}000$ worked best. Distance weighting function that worked best were the inverse distance weighting functions either $(1/(d+1))^6$ or $(1/(d+1))^7$.

Model	**LR**	**LP**
WB Baseline	88.2%	88.5%
CO99 M1	87.9%	88.2%
CO99 M2	88.5%	88.7%
Bikel 1-best	88.7%	88.7%
***k*-NN**	89.1%	89.4%

Table 2: Results for sentences of less than or equal to 40 words, from section 23 of the Penn treebank. LP/LR =Labelled Precision/Recall. CO99 M1 and M2 are (Collins 1999) Models 1 and 2 respectively. Bikel 1-best is (Bikel, 2004). *k*-NN is our final *k*-NN model.

With our *k*-NN model we achieve LR/LR of 89.1%/89.4% on sentences ≤ 40 words. These results show an 8% relative reduction in *f*-score error over our Model 1 baseline and a 4% relative reduction in *f*-score error over the Bikel parser. We compared the results of our *k*-NN model against the Bikel 1-best parser results using the paired *T* test where the data points being compared were the scores of each parse in the two different sets of parses. The 95% confidence interval for the mean difference between the scores of the paired sets of parses is [0.029, 0.159] with $P < .005$. Following (Collins 2000) the score of a parse takes into account the number of constituents in the gold standard parse for this sentence. These results show that using the methods presented in this paper can produce significant improvements in parser accuracy over the baseline parser.

References

Daniel M. Bikel. 2004. On the Parameter Space of Generative Lexicalized Statistical Parsing Models. *PhD thesis, University of Pennsylvania.*

Michael Collins. 1999. Head-driven statistical models for natural language processing. *PhD thesis, University of Pennsylvania.*

Michael Collins. 2000. Discriminative reranking for natural language parsing. *In Proceedings of the 7th ICML.*

Kristina Toutanova, Mark Mitchell and Christopher Manning. 2003. Optimizing Local Probability Models for Statistical Parsing. In *Proceedings of 14th ECML.*