A Hidden Markov Model for Text Chunking

Rachel Basse

November 18, 2010

Contents

1 Description

A first-order Hidden Markov Model (HMM) was developed to identify noun-phrases (NP) based on part-of-speech (POS) tags. The system was trained on sections 15-18 and tested on section 20 of the Wall Street Journal corpus [1]. This section describes the components of the HMM and their tested options.

1.1 Base HMM

The HMM's states, $S = \{B, I, O\}$, represent a word's inclusion in a NP. O denotes a word that is not part of a NP. B denotes the first word of a NP when that NP follows or is nested inside of another NP. I denotes any other word that is part of an NP.

This choice of states makes an occurrence of B relatively rare, creating special problems when training. Table 1 displays the distribution of states in the training and testing data, where B makes up only 2.0% of the states. Methods to compensate for this rarity are considered below.

The observation alphabet is the following set of 44 POS tags.

> {uh, sym, wp<dollar>, fw, <colon>, rp, pdt, rbs, wrb, wdt, jjr, rbr, wp, jjs, prp, nnps, <rparen>, <dollar>, <lparen>, <hash>, md, ex, vbd, <apost>, <quote>, cd, prp<dollar>, vbg, vbp, <period>, pos, cc, <comma>, nnp, nns, jj, vb, to, vbn, rb, vbz, dt, in, nn}

The states and alphabet remained constant throughout training and testing.

Several emission, transition, and initial probability distributions were generated. For the base HMM, simple proportions were used, and emission probabilities were conditioned on the current state. These base

Table 1: Distribution of HMM States

	Training		Testing		Total	
State	count	%	count	%	count	%
B	4,466	2.0	967	2.0	5,433	2.0
I	114,149	52.8	25,824	54.5	139,973	53.1
O	97,578	45.1	20,586	43.5	118,164	44.8

distributions were variously smoothed, sharpened, and combined, and the emission distribution was extended to condition on first-order transitions (t-HMM).

1.2 Laplace Smoothing

Laplace smoothing increases all counts by some constant λ and increases the normalization factor proportionally to effectively start all counts at λ. If c_i is the count for some outcome i and $|\Omega|$ is the size of the sample space, the Laplace-smoothed probability for c_i at strength λ, $\mathcal{L}_\lambda(c_i)$, is given by

$$\mathcal{L}_\lambda(c_i) = \frac{\lambda + c_i}{\lambda|\Omega| + \sum_i c_i}. \tag{1}$$

Tested λ-values ranged from .5 to 15.

1.3 Uniform Smoothing

Uniform smoothing takes a probability distribution closer to the uniform distribution. If $P(x)$ is the probability of x under distribution P based on sample space Ω, then the uniform-smoothed probability of x at strength λ, $\mathcal{U}_\lambda(P(x))$, is given by

$$\mathcal{U}_\lambda(P(x)) = P(x) - \lambda(P(x) - |\Omega|^{-1}). \tag{2}$$

In order to target the extremes (i.e., the x where $P(x) >> |\Omega|^{-1}$ or $P(x) << |\Omega|^{-1}$), this method of smoothing was varied from the linear case by sign-squaring[1]or cubing the distance of the probability from the uniform probability:

$$\mathcal{U}^2_\lambda(P(x)) = P(x) - \lambda(P(x) - |\Omega|^{-1})|(P(x) - |\Omega|^{-1})| \tag{3}$$

$$\mathcal{U}^3_\lambda(P(x)) = P(x) - \lambda(P(x) - |\Omega|^{-1})^3 \tag{4}$$

Tested λ-values ranged from .001 to 1.

1.4 Threshold Sharpening

The threshold sharpening function $\mathcal{S}_\lambda$ is applied to the emission counts in order to compensate for relatively rare observations. First, a pool of counts is collected to be later redistributed. The redistribution pool p for a set of counts c_i is given by

$$p = \sum_i c_i - \sum_i \text{saved}_\lambda(c_i), \tag{5}$$

where saved_λ is a λ-parameterized function from the nonnegative integers $\mathbb{N}$ to $\mathbb{N}$ defined by

$$\text{saved}_\lambda(n) = \begin{cases} n & \text{iff } n > \lambda \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

[1]Sign-squaring is simply squaring that preserves the sign of the input, e.g., `sign-square(-x) = -(x`^{2}`)`.

Next, a new set of counts c'_i is created by redistributing p among the saved counts according to their distribution:

$$c'_i = \left(\frac{\text{saved}_\lambda(c)}{\sum_i \text{saved}_\lambda(c_i)} \right) p + \text{saved}_\lambda(c). \tag{7}$$

Finally, the c'_i are converted to integers by a fix function so that

$$\sum_i \text{fix}(c'_i) = \sum_i c_i. \tag{8}$$

Then

$$\mathcal{S}_\lambda(c) = \text{fix}(c'). \tag{9}$$

The fix function was not automated.

The reasoning behind threshold sharpening is that any counts below the threshold are so rare as to be inconsequential with regards to classification. Their presence only decreases the emission probabilities for the other outcomes. So these low counts are redistributed as counts for the other outcomes respecting the rest of the distribution.

The emission counts for state B were sharpened with λ-values ranging from .01% to 1% of $\sum_i c_i$.

1.5 Double Counting

A common assumption in natural-language grammars is that the internal structure of a phrase is independent of its position within other phrases. That is, for example, the rules for NP-structure are the same whether the NP is a daughter of a PP, a sister of a PP, a sister of a VP, etc.

With this position-independence in mind, the distinction between the B and I states poses a problem. The distinction between a B state and an I state that occurs at the beginning of a NP is made to encode information about the position of the NP with respect to its siblings and ancestors. Assuming this information is irrelevant to the internal structure of a NP, it makes sense to count any transition or emission from B as a transition or emission from I.

This double-counting (DC) was done on counts that involved transitioning or emitting from a B state since the distinction becomes irrelevant once inside of the NP. To keep the distribution of states constant when the frequency of I is increased by that of B, the counts for B and O states were increased proportionally according to their distributions.

1.6 Distribution Subtraction & Repulsion

Distribution subtraction (DS_λ) subtracts a fraction λ of each count in one count-distribution from its respective count in another distribution, setting any negative counts to zero. The goal is to make the distributions more distinct by removing or decreasing the counts that they have in common. This simple approach, however, does not always have the intended effect. In a segment of distributions such as $a = (500, 100)$ and $b = (10, 200)$, if a is subtracted from b, the new segments become $a = (500, 100)$ and $b = (0, 100)$. The first counts are now more distinct but the second counts are more similar, identical even. The second count has also been decreased by a large fraction of itself, which can have unintended negative consequences. Distribution repulsion (DR_λ) avoids these problems.

In distribution repulsion, each count from two distributions is compared. The larger count is increased by some fraction λ of itself, and the smaller count is decreased by the same fraction of itself. For the above example, if $\lambda = .1$, $a = (500, 100)$ and $b = (10, 200)$ are converted to $a = (550, 90)$ and $b = (9, 220)$, making them easier to distinguish.

One last negative consequence to avoid occurs if the two distributions are not on the same scale, i.e., if the difference between their total counts is large. In this case, the counts of the smaller distribution

will be reduced more often, even those counts that are high relative to its own scale, which will reduce its overall scale and tend to flatten the distribution rather than enhance its characteristic shape. The larger distribution will similarly be disproportionately increased and sharpened. To avoid this, the distributions are first normalized and the comparison is made on these normalized counts.

Tested λ-values ranged from .05 to .6.

2 Results

Results for the base and variations are displayed in Table 2. Only optimal results for each option are shown. Several options that improved the scores for the base HMM had a negative or no effect on the t-HMM. These negative or null results are not reported; they include all smoothing, double-counting, sharpening, and subtraction of B from I.

Underflow did not occur in the base case, and converting to log-probabilities had no effect on the base scores. Log-probabilities were used throughout as a precaution and for computational simplification.

Table 2: Test Results

Option	Accuracy	Precision				Recall				F-measure			
		I	B	O	Avg	I	B	O	Avg	I	B	O	Avg
base	94.66	95.3	96.2	93.9	95.1	95.0	52.3	96.2	81.2	95.2	67.8	95.0	86.0
change from base													
$\mathcal{L}_3$	–	–	.2	–	.1	–	–	–	–	–	–	–	–
$\mathcal{U}^2_{.02}$	.08	.1	.3	–	.1	–	4.1	–	1.3	–	3.3	–	1.1
$\mathcal{S}^B_\lambda$	.08	.1	.2	–	.2	–	3.9	–	1.3	–	3.2	–	1.1
DC	-.18	-.3	.3	–	–	.1	-9.0	–	-3	-.2	-8.0	–	-2.7
$\mathrm{DS}^{I-B}_{.2}$	.08	.1	.3	–	.1	–	4.1	–	1.3	–	3.3	–	1.1
$\mathrm{DR}^{BI,IO}_{.3,.5}$	.08	.1	.6	–	.3	–	4.1	–	1.3	–	3.4	–	1.2
t-HMM	.45	.7	-1.2	.2	-.8	.1	26.6	.4	8.8	.4	17.5	–	6.0
t-HMM	95.11	96.0	92.7	94.1	94.3	95.1	78.9	95.9	90.0	95.6	85.3	95.0	91.9
change from t-HMM													
$\mathrm{DS}^{I-O}_{.1}$	.01	-.3	-.7	.4	-.1	.4	–	-.5	–	–	-.1	–	–
$\mathrm{DR}^{IO}_{.4}$	.19	.2	.3	.2	.2	.2	-.2	.2	–	.1	–	.2	.2

3 Discussion

The best scores for options operating on the base HMM were 94.7% accuracy, 95.4% precision, and 82.5% recall, for an 87.2% F-measure. All scores for the I and O states were in the mid-90's. The average recall was brought down by a recall of 56% for B. The best scores for options operating on the t-HMM were 95.3% accuracy, 94.5% precision, and 90.0% recall, for a 92.1% F-measure. As Table 3 shows, these scores fall in the middle of the scores of the systems from the CoNLL shared task[2], which used the same training and testing data with a slightly different set of states and alphabet.

Both sets of best scores were obtained using emission probabilities transformed by distribution repulsion. Most options behaved differently on the base HMM versus the t-HMM, so they are discussed separately.

3.1 Base HMM

Laplace smoothing had almost no effect at any of the tested strengths. At the higher strengths, it begins to cause misclassifications of B and O as I. Uniform-linear smoothing increased every type of misclassification

Table 3: Comparison to CoNLL-2000 Systems

System	Precision	Recall	F-measure
ZDJ01	94.29	94.01	94.13
KM01	93.89	93.92	93.91
CM03	94.19	93.29	93.74
KM00	93.45	93.51	93.48
Hal00	93.13	93.51	93.32
TKS00	94.04	91.00	92.50
ZST00	91.99	92.25	92.12
Dej00	91.87	92.31	92.09
t-HMM w/ DR	**94.50**	**90.02**	**92.06**
Koe00	92.08	91.86	91.97
Osb00	91.65	92.23	91.94
VB00	91.05	92.03	91.54
PMP00	90.63	89.65	90.14
Joh00	86.24	88.25	87.23
VD00	88.82	82.91	85.76
baseline	72.58	82.14	77.07

at all tested strengths. Uniform-square smoothing decreases misclassifications of B as I up a point at which it begins to cause I-as-B misclassifications. This makes sense if linear-square smoothing is targeting the low B probabilities and simply making the B states more likely. It seems plausible that the training data was representative enough that no room is left for improvement by smoothing.

Threshold sharpening of B reduced misclassifications of B as I and affected nothing else. This is the expected result of increasing the highest emission counts for B. That the recall of B was not reduced implies that the counts below the threshold were indeed inconsequential.

Double-counting B as I improved the recall of I as expected. However, since the added B counts were such a small percentage of the total I counts, this improvement was slight. Additionally, since double-counting made the B state more closely resemble the I state, the misclassifications of B as I increased, decreasing the recall of B and precision of I. This negative impact far outweighed the former gain. Despite this problematic implementation, the conceptual basis of this option is thought to be sound and potentially beneficial in a more restricted application.

The distribution subtraction of B from I had almost the same effect as threshold sharpening on B. The misclassification of B as I was reduced, and nothing else was affected. Due to the particular relationship between the B and I count distributions, the subtraction appears to have, in effect, relatively increased the counts for B. Subtracting a smaller distribution from a larger one may sometimes be preferable to the potential loss of information from sharpening, but its successful use is more complicated and precarious.

A moderate distribution repulsion of .3 between B and I produced results similar to their subtraction. Even a strong repulsion of .5 between I and O had an insignificant effect on their classification.

3.2 T-HMM

The t-HMM produced a dramatic improvement in misclassifications of B as I and a slight improvement in misclassifications of I as O. Misclassifications of O as I and of I as B both increased moderately. The relationship between O and B was unaffected. The fact that there are no transitions from O to B and very

few transitions from B to B goes a long way toward explaining how considering transitions changes the relationship between B and the other states. The change in the relationship between O and I is not well understood.

Distribution repulsion between I and O was the only option that produced significant improvements to the t-HMM. Near a λ-strength of .4, both I-as-O and O-as-I misclassifications were reduced from the baseline, leaving the other relationships largely unaffected. The behavior of the I and O classifications under this option is also not well understood. Table 4 shows their misclassification counts at several strengths of repulsion.

Table 4: I and O Misclassifications under Repulsion
$O.I$ denotes an actual O that was classified as an I.

λ	$O.I$	$I.O$
0	841	1213
.2	882	1167
.3	921	1110
.35	1067	1110
.4	790	1175
.45	776	2038
.5	776	2038
.6	776	2035

4 Conclusion

The best performance was obtained from the HMM (t-HMM) that conditions emission probabilities on transitions rather than on states. The most effective option in both the base and t-HMM was distribution repulsion (DR). The t-HMM-with-DR system achieved higher precision than any system in the CoNLL-2000 shared task, though a mediocre recall placed it in the middle of the pack overall. Refinements to the DR function and a restricted form of double-counting could make this system more competitive.

References

[1] E. Charniak, D. Blaheta, N. Ge, K. Hall, J. Hale, and M. Johnson, *BLLIP 1987-89 WSJ Corpus Release 1*. Philadelphia: Linguistic Data Consortium, 2000.

[2] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: language-independent named entity recognition," in *Proceedings of the Seventh Conference on Natural language learning at HLT-NAACL 2003 - Volume 4*, (Morristown, NJ, USA), pp. 142–147, Association for Computational Linguistics, 2003.