

Combination of a Hidden Tag Model and a Traditional N-gram Model: A Case Study in Czech Speech Recognition

Pavel Krbec, Petr Podveský, Jan Hajič

Charles University, Prague, Czech republic

{krbec, podvesky, hajic}@ufal.mff.cuni.cz

Abstract

A speech recognition system targeting high inflective languages is described that combines the traditional trigram language model and an HMM tagger, obtaining results superior to the trigram language model itself. An experiment in speech recognition of Czech has been performed with promising results.

1. Speech Recognition of Inflective Languages

Inflective languages pose a hard problem in speech recognition due to two phenomena: highly inflective nature (causing data sparseness problem and excessive vocabulary growth), and free word order (causing the traditional speech recognition systems, such as n-gram Hidden Markov Models (HMMs) on word forms to be less accurate than for English). Specific methods targeting speech recognition of inflective languages have been already introduced in [1], [2] and [3]. The authors mainly focus on improving the language model by decomposing words from the vocabulary into stems and endings. This approach has mainly helped in reducing the size of the vocabulary of the speech recognizer reducing the WER slightly.

2. Combining Taggers with Language Models

Tagger has been to our best knowledge first introduced as a speech recognition language model component in [4] without improving results over the baseline bigram model. The idea has been further explored in [5] where the author proposes the interpolation with a trigram model.

$$P(W) = \lambda P(w_i | w_{i-2}, w_{i-1}) + (1 - \lambda) Q(w_i | g(w_{i-2}), g(w_{i-1})), \quad (1)$$

where $g(w_i)$ is the tagging function. The importance of formula (1) for languages with the data sparseness problem is that the new component Q can have enough evidence to give us reliable statistics about the word sequence W as the size of tag set tends to be much smaller than the size of the vocabulary itself. The problem with approach (1) is that the tagging function $g(w_i)$ depends on all words of the utterance (supposing that the tagging component is performed by an HMM tagger). The standard solution is to replace the probability Q by a new probability Q^* :

$$Q^*(w_i | w_1, \dots, w_{i-1}) = \sum Q(w_i | g_1, g_2) T(g(w_{i-2}) = g_2, g(w_{i-1}) = g_1) \quad (2)$$

The new probability T is the corresponding forward probability of the HMM tagger. The calculation of the forward probabilities

is an expensive task in this model. We can thus maximize the sequence of tag-word pairs instead of maximizing sequence of words only. One of the consequences of this approach is that we do not use the forward probabilities at all. We compute the standard Viterbi decoding instead. The new $Q(T, W)$ function becomes:

$$Q^{**}(w_i, g_i | w_1, \dots, w_{i-1}) = p_t(g_i | g_{i-1}, g_{i-2}) p_o(w_i | g_i, g_{i-1}) \quad (3)$$

where p_t and p_o are the corresponding transition and output probabilities distributions of the HMM-tagger and g, g_i, g_{i-1} are the tags corresponding to words w, w_i, w_{i-1} . We can hope that even without ever seeing the word w_i in the train data the information hidden in the corresponding preceding tags will tell us something about the word itself. In our setup we are using the morphological analyzer [6] so that for every input word only the list of possible tags is considered.

2.1. Taggers for Inflective Languages

The average tagset usually contains around 1000 - 2000 distinct tags; the size of the set of possible tags can reach several thousands. Apart from agglutinative languages such as Turkish, Finnish and Hungarian (see e.g. (Hakkani-Tur et al., 2000 [7])), there have been attempts at solving the problem of tagging for some of the highly inflectional European languages, such as (Daelemans et al., 1996) [8], (Erjavec et al., 1999) (Slovenian) [9], (Hajic and Hladka, 1998) (Czech) [10] and (Hajic, 2000) (five Central and Eastern European languages) [11]. Some new techniques have been explored for tagging inflective languages by combining rules and statistical methods [12]. By using this method the authors report results on Czech slightly above 95%. We should note that the use of rules for tagging is not practical for the task of HMM based speech recognition as the rule-based component of the tagger has the form of a restarting automaton with deletion (Platek 1999) [13]. This means that the whole rule based tagging system has more than a context free power and can't be integrated with the FSA approach we are using. But even this system has never reached - in the absolute terms - a performance comparable to English tagging (such as [14]), which stands above 97%.

2.2. HMM Tagger Component

We decided to use a strictly probabilistic trigram based HMM tagger. The tagger operates according to the well known formula

$$T = \operatorname{argmax}_T P(W|T)P(T) \quad (4)$$

For the purpose of our LM we need reliable distributions p_t and p_o :

$$\begin{aligned}
p_t(t_i|t_{i-2}, t_{i-1}) &= \lambda_3^t P(t_i|t_{i-2}, t_{i-1}) + \\
&\lambda_2^t P(t_i|t_{i-1}) + \lambda_1^t P(t_i) + \lambda_0^t \frac{1}{|T|} \\
p_o(w_i|t_{i-1}, t_i) &= \lambda_3^o P(w_i|t_{i-1}, t_i) + \\
&\lambda_2^o P(w_i|t_i) + \lambda_0^o \frac{1}{|V|}
\end{aligned} \tag{5}$$

where $P(\dots)$ is the raw maximum likelihood estimate of probability distributions. $|V|$ is the size of the word forms dictionary. $|T|$ is the size of the tagset. The interpolation coefficients are grouped to buckets according to histories. We use a standard bucketing scheme based on the formula

$$v(h) = \frac{C(h)}{|\{w : C(h, s) > 0\}|}, \tag{6}$$

where s is either word or tag. With this approach we get for each history h (in our case of length two) an index $v(h)$ into a certain bucket. The buckets (as shown in Table 1) are chosen during the training phase so that each contains approximately the same amount of data.

Table 1: Example of buckets for the smoothing coefficients of the p_t probability.

λ_3^t	λ_2^t	λ_1^t	λ_0^t	Bucket interval
0.032	0.788	0.178	0.002	(0.00; 1.00)
0.124	0.706	0.167	0.003	(1.00; 1.48)
0.215	0.643	0.138	0.003	(1.48; 1.83)
0.749	0.222	0.029	0.001	(22.80; 26.10)
0.816	0.177	0.004	0.003	(92.48; ∞)

The maximum likelihood estimate has been obtained by computing relative frequencies from the hand-annotated text. We have used the PDT [15] development set which has been divided into training and held-out data. The training of smoothing coefficients has been performed by maximizing the probability of observing the held-out data by the bucketing model described above. This maximization has been done using the standard Baum-Welch algorithm [16]. The error rate of the statistical tagger we have trained is slightly above 95% on the evaluation part of the PDT corpora.

3. Speech Corpora

3.1. Acoustic Data

Our acoustic corpus consists of 26 hours of clean speech of broadcast radio and TV news. Weather forecast, traffic announcements and sport news were excluded from the corpus. The channel has been sampled at 22.05 kHz with 16-bit resolution. 22 hours were used for acoustic modeling the remaining four hours were used as the test set. The corpus was collected at the University of West Bohemia [17].

3.2. Acoustic Features

The acoustic features are Mel-Frequency Cepstral coefficients. Each acoustic feature vector consists of twelve cepstral coefficients plus energy and their delta and delta-delta coefficients. Cepstral mean subtraction was applied to all feature vectors on a per utterance basis [18].

3.3. Baseline system

In order to see how much improvement the integration of the tagger component will bring us we decided to implement the best baseline we can achieve using traditional LM techniques. It is still impossible to run a full trigram decoder on word forms for Czech due to its vocabulary size. Thus we took a bigram decoder (using the AT&T tools [19]) and created lattices with it. The lattices have been transformed to trigram lattices and rescored with a trigram language model which has been trained on a collection of Lidove Noviny (Czech daily newspaper) containing approximately 33 million words.

Our bigram back-off language model used in the decoder and the trigram model used for lattices rescoring has been build with a vocabulary of 62k most frequent tokens. The out-of-vocabulary rate of the transcriptions of the test data is 8.17%. We utilized [20] to estimate the corresponding back-off parameters of the language model. The oracle accuracy of the held-out data lattices is 87.76%.

From our preceding experiments we learned that the correct setting of scaling factors makes a significant difference on the WER. Based on the preceding experiments we have also found that introducing the insertion penalty does not get us any gain in the accuracy as long as we use the optimum scaling factors for the language model. On a set of held-out data (400 lattices) we found the optimal scaling factors for the baseline trigram LM and the acoustic model. The scaling factor f_{LM} is optimized for achieving the best accuracy on the held-out data with the following formula:

$$-10^3 f_{LM} \log P(W) - \log P(A|W) \tag{7}$$

Table 2: Finding the optimum scaling factor on the set of held-out data

f_{LM}	Accuracy
11	70.35%
12	70.88%
13	71.33%
14	71.51%
15	72.04%
16	71.59%

The baseline trigram system uses the scaling factor 15 as shown in Table 2. The Accuracy of the baseline system on the test data is 71.90%.

4. Speech recognition experiment

The formula 1 gives us a hint how to combine the tagger component with the trigram language model. For practical reasons we decided to use a slightly different approach similar to the way we tuned our baseline. Our goal is to find the optimum scaling factors f_{LM} and f_{tag} on the same set of the held-out data as used for the baseline tuning. The formula now becomes:

$$-10^3 (f_{LM} \log P(W) + f_{tag} \log Q^{**}) - \log P(A|W) \tag{8}$$

The effect of tuning the parameters f_{LM} and f_{tag} can be seen in figure 1.

We can see from the figure 1 that the introduction of the tagger component leads to the accuracy improvement in every case. The maximum accuracy gain point occurs with the factors

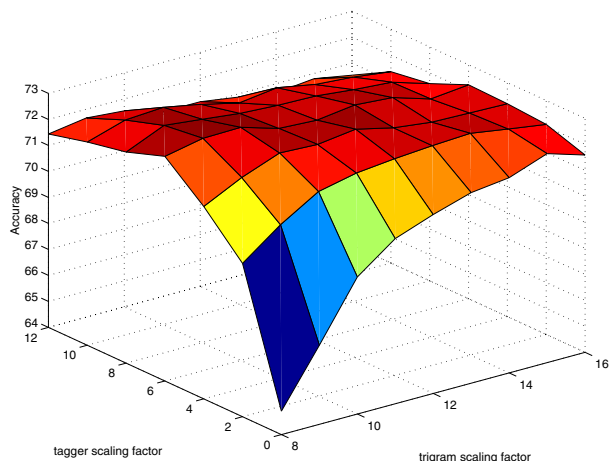


Figure 1: Accuracy as a function of f_{LM} and f_{tag}

at $f_{LM} = 10$ and $f_{tag} = 5$. The WER of this best setup is 27.03% on the held-out data.

The test set contains 2500 lattices with the oracle accuracy of 87.69%. By combining the trigram language model with the tagger component we succeeded (see Table 3) to improve the baseline by 1.21% absolute.

Table 3: Test data experiments

Accuracy	Language model used
70.24%	bigram model
69.31%	trigram model $f_{LM} = 10$
71.90%	trigram model $f_{LM} = 15$ (baseline)
73.11%	combination $f_{LM} = 10, f_{tag} = 5$
87.69%	Oracle Accuracy

5. Conclusion

In this paper we introduced a new language model which is a combination of a traditional trigram language model and an HMM tagger. We achieved a promising improvement in accuracy over the baseline trigram model. The method described in this paper can be combined with the morpheme based approach as introduced in [1].

6. Acknowledgements

This work has been supported by the Grant Agency of the ČR No. 405/03/0913 and Ministry of Education of the ČR No. LN00A063

7. References

[1] W. Byrne, J. Hajic, P. Ircing, F. Jelinek, S. Khudanpur, P. Krbec, J. Psutka, "On Large Vocabulary Continuous Speech Recognition of Highly Inflectional Language - Czech", Eurospeech, Aalborg, 2001.

[2] W. Byrne, J. Hajic, P. Ircing, P. Krbec, J. Psutka, "Morpheme Based Language Models for Speech Recognition of Czech", TSD, Brno, Czech republic, 2000.

[3] P. Ircing, J. Psutka, "Lattice Rescoring in Czech LVCSR System Using Linguistic Knowledge", Specom, 2002.

[4] E. Brill, D. Harris, S. Lowe, X. Luo, P.S. Rao, E. Ristad, S. Roukos, "A Hidden Tag Model for Language", LM workshop report at Johns Hopkins University, Baltimore, 1995.

[5] F. Jelinek, "Statistical Methods for Speech Recognition", The MIT Press, Cambridge, 70-73, 1997.

[6] J. Hajic, "Disambiguation of Rich Inflection (Computational Morphology of Czech)", MFF UK, 342pp., Prague 2002.

[7] D. Hakkani-Tur, K. Oflazer, and G. Tur, "Statistical morphological disambiguation for agglutinative languages", In Proceedings of the 18th Coling 2000, Saarbruecken, Germany.

[8] Walter Daelemans, Jakub Zavrel, Peter Berck, and Steven Gillis "MBT: A memory-based part of speech tagger generator.", In Proceedings of WVLC 4, pages 14-27, ACL, 1996.

[9] Tomaz Erjavec, Saso Dzeroski, and Jakub Zavrel, "Morphosyntactic Tagging of Slovene: Evaluating PoS Taggers and Tagsets", Technical Report IJS-DP 8018, Dept. for Intelligent Systems, Jozef Stefan Institute, Ljubljana, Slovenia, 1999.

[10] Jan Hajic and Barbora Hladka, "Tagging inflective languages: Prediction of morphological categories for a rich, structured tagset.", In Proceedings of ACL/COLING98, Montreal, Canada, pages 483-490, ACL/ICCL, 1998.

[11] Jan Hajic, "Morphological tagging: Data vs. dictionaries.", In Proceedings of the NAACL00, Seattle, WA, pages 94-101, ACL, 2000.

[12] Hajic, Krbec, Kveton, Oliva, Petkevica, "Serial Combination of Rules and Statistics: A Case Study in Czech Tagging", ACL, Toulouse, France, 2001.

[13] M. Platek, P. Jancar, F. Mraz, and J. Vogel, "On restarting automata with rewriting.", Technical Report 96/5, Charles University, Prague, 1995.

[14] Adwait Ratnaparkhi, "A maximum entropy model for part-of-speech tagging." In Proceedings of EMNLP 1, pages 133-142, ACL, 1996.

[15] Jan Hajic et al., "PDT v. 1.0 CD-ROM", LDC 2001.

[16] Bahl, Jelinek, Mercer, "A maximum likelihood approach to continuous speech recognition", In IEEE Transactions on PAMI, 5(2), 179-190, 1983.

[17] J. Psutka and V. Radova and L. Müller and J. Matoušek and P. Ircing and D. Graff, "Large Broadcast News and Read Speech Corpora of Spoken Czech", Proceedings of Eurospeech 2001, Aalborg, 2001.

[18] Young et al., S., "The HTK Book", Entropic Inc., 1999.

[19] M. Mohri, F. Pereira and M. Riley, "Weighted Finite-State Transducers in Speech Recognition", Proceedings of ASR2000, International Workshop on Automatic Speech Recognition: Challenges for the Next Millennium, Paris, 2000.

[20] Stolcke, A., "SRILM - An Extensible Language Modeling Toolkit", Proc. Intl. Conf. on Spoken Language Processing, 901-904, Denver, 2002.

