UTILIZING PROSODY FOR UNCONSTRAINED MORPHEME RECOGNITION

Volker Strom (1), Henrik Heine (2)
e-mail: vst@ikp.uni-bonn.de heine@informatik.uni-hamburg.de

(1) Institute of Communications Research and Phonetics (IKP), University of Bonn,
(2) University of Hamburg, Computer Science Institute, NatS - Natural Language Systems

ABSTRACT

Speech recognition systems for languages with a rich inflectional morphology (like German) suffer from the limitations of a word–based full–form lexicon. Although the morphological and acoustical knowledge about words is coded implicitly within the lexicon entries (which are usually closely related to the orthography of the language at hand) this knowledge is usually not explicitly available for other tasks (e.g. detecting OOV words). This paper presents an HMM–based 'word' recognizer that uses morphemes on the string level for recognizing spontaneous German conversational speech (VERBMOBIL corpus). The system has no explicit word knowledge but uses a morpheme–bigram to capture the German word and sentence structure to some extent. The morpheme recognizer is tightly coupled with a prosodic classifier in order to compensate for some of the additional ambiguity introduced by using morphemes instead of words. Although the recognizer's morpheme accuracy of 85.3% is comparable to that of our word–based decoder (word accuracy 86%) until now the benefit of introducing the prosodic classifier is not yet clear.

1. INTRODUCTION

Using lexica with fully inflected word forms in speech recognition for languages with a rich morphology introduces problems into the development and successfull application of such systems. This paper presents an HMM–based speech recognizer that uses morphemes instead of words.

The proposed architecture easily copes with problems that can usually not be addressed in an elegant manner, e.g. a speech recognizer might know of the (compound number) word 'einhundertsiebenundzwanzig' (one hundred twenty seven) but not of the word 'einhundertachtungzwanzig' (128) because numbers are not exhaustivly enumerated in lexica. But even if numbers were modeled by means other than the (orthographically correct) full–form (e.g. by a regular grammar using their corresponding sub parts) the inherent problems of composition ('Montagstermin' (a meeting on monday), 'Dienstagstermin' (on tuesday)), derivation ('absagen'/'Absage' (to cancel/cancelation), 'ansagen'/'Ansage' (to announce/announcement)), inflection ('erster', 'erstem', 'ersten', 'erste' (different cases/gender of 'first')) and ad-hoc word creation ('Diaabend-Weintrink-Revisionstreffen') remain.

Although our recognizer can potentially produce the correct solution (i.e. morpheme sequence) as long as the input consists of morphemes known to the system (the potential 'coverage of the language' is huge compared to recognition systems using full–forms) the task has become more difficult due to the additional ambiguity that has been introduced into the model by using smaller lexical units (on the acoustical as well as on the string level, i.e., language model). In order to compensate for some of the loss we incorporated a prosodic classifier and tightly coupled the two systems. We expected to benefit from the additional acoustic information and the predictive power of a 'prosodiy language model'.

The following sections will first describe the standard speech decoder and the prosodic classifier before we explain in section 5 how the two were combined into one morpheme–based speech recognizer. Section 6 will then focus on the evaluation of the overall system performance compared to a word–based recognizer using the same training and test data. Finally we will present some results that show how difficult the remaining problem of the reconstruction of meaningful words from morpheme sequences is.

2. MORPHEME RECOGNIZER

We used an adapted version of the HTK/HVite V2.1[1] speech recognizer for all experiments. The recognizer uses a (pruned) Viterbi search to find the globally 1–best path through the search net given the utterance.

In addition the recognizer can produce a lattice by keeping the (locally) `k` best Viterbi hypotheses.

[1]Hidden Markov Model Toolkit of Entropic Research Laboratory, Inc.

When run in this mode the result can either be a lattice or an N–best list (derived by a backward A^*–search) of sentence level solutions.

We used a backoff bigram language model and (clustered) cross–word triphone HMMs which were compiled into the (static) word level search net before decoding.

3. PROSODIC CLASSIFIER

The prosodic classifier is an improved version of the one described in [5]. It distinguishes whether a syllable is accented or not (**A/0**) and followed by a phrase boundary or nor (**B/0**). This makes up four classes **AB, A0, 0B, 00** (prosodic labels).

The prosodic label of a morpheme is composed of the accent label of the syllable carrying the standard lexical accent, and the boundary label of its last syllable. If a morpheme has no syllable nucleus — like some inflection morphemes — it gets the default label **00**.

The classifier's input is the speech signal, its fundamental frequency[2], and a phoneme segmentation within the analysis window, which defines the time alignment and the phoneme symbol of the syllable nuclei. The F_0 is interpolated and decomposed[5]. Three short time energy features are derived from the speech signal. Together with the F_0 features this makes 11 basic prosodic features for each frame describing the energy and F_0 contour.

During automaticaly labelling of the training corpus, a context of up to two preceeding and two succeeding morphemes is used to get an analysis window of up to five syllables (two left and two right from the syllable being classified). During morpheme recognition no right context is used. The final feature vector for a syllable consists of the basic features at the center of each syllable nucleus, the syllable nucleus duration, normalized duration (with respect to the intrinsic phoneme duration), and the frame distances between the syllable nuclei. If no right context is available (as is the case for decoding), an automatic detection of one right–context syllable nucleus is tried using the energy features. Even though the normalized duration cannot be determined in this case, phrase boundary detection is still clearly improved.

The classifier was initially trained with 668 manually prosodically labelled [2] turns of the VERBMOBIL corpus. The phoneme segmentation for the trainig data was obtained by forced alignment with respect to the most frequent pronunciation variants.

For each window type (12 in total) a gaussian distribution classifier was trained: In a speaker–wise leave–one–out procedure first the best of all features were automatically selected (at most 41 of maximal 66). Then a cluster analysis was performed. For the initial training set, the optimal number of subclasses turned out to be 2 (we did not compute this for the whole training set yet). Thus e.g. the class **A0** is split into classes $\mathbf{A0_1}$ and $\mathbf{A0_2}$ which may represent high and low accents. The alternative would have been to explicitly distinguish high and low accents beforehand and train the classifier with these finer labels. But the more classes we use, the less well they are predictable by the prosody model (see next section).

4. PROSODY MODEL

The so called 'prosody model' is the link between the HMM recognizer and the prosodic classifier. It models the relation between accentuation and phrasing on the one hand, and morphotactics on the other hand.

Due to the very small corpus we did not use morphemes themselves for that purpose, but rather morpheme categories. Like words that behave in a syntactically similar fashion can be united to one word class, it might well be the case that morphemes that behave morphologically similar (and thus fall into the same morpheme categorie) can be treated the same in terms of prosody (i.e. prosodic labels).

We decided to use the morpheme category system developed at the University of Bielefeld [4]. For affix classification it uses three features of which one is accentability. In total 19 categories are distinguished.

For 5728 of the 44650 VERBMOBIL I lexicon entries (see section 6.1) the categories of the morpheme segments could be obtained from a word–related morpheme data base. 505 further entries were semi–automatically added. This extended lexicon covers more than 70% of the selected data. 75% of the missing 981 words occur only once or twice.

The word–related morpheme data base had no entries for words consisting of only one morpheme. This concerns mainly function words. For those words the word category was taken instead, which was obtained by an automatic POS tagging of the whole corpus. Morpheme and word categories together make 53 categories plus one for non–speech.

Five of these categories contain many morphemes with no syllable nucleus which might be followed by a phrase boundary (e.g. -t, -s). Since these boundaries cannot be acoustically classified (see previous section), they were shifted to the preceeding morpheme in the initial training data and the category containing the non–syllabic morphemes was split — one category for 'has a syllable nucleus' and one for 'has not'. This way these cases could be handled by the prosody model.

The acoustic prosodic classifier trained with the initial training set was used to classify the whole corpus. Each morpheme is tagged with its category and prosodic label. There are $53 \cdot 4 + 6 = 218$ possible combinations, of which 178 actually occur in the training set. With these sequences of morpheme categories combined with prosodic labels a backoff trigram model is trained, which has a perplexity of 9.8 on the training and 10.9 on the test set.

[2]The fundamental frequency F_0 was determined with the `get_f0` program of the Entropic Signal Processing System.

The prosody model does not only yield the prosody factor (see next section) in combination with the a posteriori probability given by the prosodic classifier. It was also used to predict new prosodic labels for the training set. Again a classifier trained with these labels classified the corpus and with the resulting labels yet another prosody model was trained, and so on. This way the prosody model and the prosodic classifier are iteratively adapted to each other. The benefit of the prosody factor during morpheme recognition is maximal, when the agreement of prosody recognition and prediction is maximal.

During classification and during prediction biases both for the accent super class ($\mathbf{A0} + \mathbf{AB}$) and the break super class ($\mathbf{0B} + \mathbf{AB}$) were introduced to guarantee that they are in the same proportion to their inverse class. Otherwise the less frequent labels would vanish during the iterative adaptation of prosody recognition and prediction.

During 10 iterations the overall recognition rate for all four classes unfortunately stays nearly constant at 72%. While the recognition rate for the accent super class rises from 61% to 65%, the recognition rate of the less frequent break super class falls from 67% to 43%.

5. INTEGRATION

We decided to apply the prosodic analysis whenever the Viterbi search hypothesized a morpheme end. At this point in the search process all phone boundaries are available and thus the segmentation information of the current and the preceeding morpheme can be passed to the prosodic classifier. Until now we have integrated the prosodic classifier into the 1–best search only.

Let us denote morphemes with lower case characters x, y, z and the corresponding categories with upper case characters X, Y, Z. An upper right index indicates that a morpheme z may belong to different categories $Z^1 \dots Z^n$. The probability R_c of z belonging to a certain Z^c is

$$R_c = P(Z^c|z) \cdot P(Z^c|X^a, Y^b)$$

assuming that the preceeding morphemes x and y belong to the categories X^a and Y^b. The probability $P(Z^c|z)$ is obtained by counting out the pairs (morpheme, category) within the training set. The probability $P(Z^c|X^a, Y^b)$ is estimated by a backoff trigram model.

The prosodic classification of a morpheme z gives the a posteriori probability P_{ij} for each prosodic class $ij \in \{00, 0B, A0, AB\}$.

The prosody model yields for a certain prosodic label ij and a certain category index c the probability

$$Q_{ij,c} = P(Z^c_{ij}|X^a_{ef}, Y^b_{gh})$$

assuming that the preceeding morphemes x, y belonged to categories X^a, Y^b and have been prosodically labelled with ef and gh respectively.

The prosody factor is obtained by maximizing the product of these three probabilities over all prosodic labels ij and all category indices:

$$\mathrm{PFac} = max_{i,j,c}\,(P_{ij} \cdot Q_{ij,c} \cdot R_c)$$

The prosody factor is then used like an additional language model. A bias is applied in order to balance between the prosody factor and the morpheme bigram language model.

It should be noted that our current implementation violates the Viterbi assumption: the static structure of the search net entails a dependency on the preceeding morpheme only (as is sufficient for a bigram) — since we are using long term dependencies (segmentation of the two preceeding morphemes, morpheme category trigram) and we are applying the prosodic factor in a "post mortem" fashion at the end of a morpheme, the true globally best Viterbi path is only approximated.[3]

6. EXPERIMENTS

6.1. The Data

We used the German part of the VERBMOBIL I corpus [3]. It consists of 13939 turns on 8 CD ROMs. 3638 turns containing spelling sequences, non–words or aborted words were discarded, as well as 2874 turns that were not covered by the lexicon extended with morpheme categories (see section 4). Of the remaining 7427 turns 911 were used for testing, 6516 for training.

For the VERBMOBIL I corpus there existed already a lexicon containing orthography, morpheme segmented orthography, and morpheme segmented pronunciation [1]. This lexicon was used to segment the corpus into morphemes and to obtain the morpheme based pronunciation lexicon.

Although it is clear that the acoustical ambiguity of our model is increased by using morphemes instead of words we hoped that we would benefit on the string level. From table 1 and 2 it seems that the predictive power of the language model (perplexity 29.15 instead of 70.20) and the restriction imposed by the word (i.e., morpheme) inventory should make the classification problem easier in the case of morpheme recognition. Unfortunatly it turns out that the overall entropy of the morpheme–coded test set given the corresponding bigram is still greater compared to words.

6.2. Results

The recognition results are listed in table 3. Since we did not use a development set all weights, biases

[3] This is similar to the way others have implemented 'cheap trigrams'.

		token	types	bigrams
wrd	train	99.1K	2.5K	27.9K
	test	15.2K	1.1K	7.1K
	OOV	179 (1.17%)	153	
mor	train	154.5K	1.3K	18.5K
	test	22.2K	0.7K	6.2K
	OOV	76 (0.34%)	59	

Table 1: Word–coded (wrd) and morpheme–coded (mor) data: 86.8% of the morpheme pairs were seen in the training set while it were only 74.0% for the word pairs (bigram hits on token).

		test		train	
		perpl.	entr.	perpl.	entr.
wrd	test	17.66	4.14	30.23	4.92
	train	**70.20**	**6.13**		
	both	31.37	4.97		
mor	test	16.77	4.07	20.03	4.32
	train	**29.15**	**4.87**		
	both	21.90	4.45		

Table 2: Perplexity/entropy measures (for test set and training set) of language models trained on varying sets. Types not seen in the training data were added and backed–off by the language model (unigram count was 1.0)

and insertion penalties were optimized on the test set. OOV types were added to the recognizer inventory and backed–off by the language model.

Even though the accuracy of the morpheme recognizer is close to the word accuracy[4] it turns out that the greater number of morphemes results in a lower word accuracy when trying to reconstruct words from morphemes.

For this reconstruction we used the same word list as for the word recognizer. For any sequence of morphemes in the 1–best morpheme solution that made up a known word this word was added to the morpheme sequence (using the respective start and end points of the morpheme sequence — thus building a graph). Finaly the morphemes were removed (while preserving the connectivity of the graph) and the graph was compared to the word–based reference (see 'rcb' in Table 3).

Although we used a tighter beam for the prosody experiments due to the great number of experiments that had to be run in order to optimize the various parameters, the results show that the integration of the two classifiers did not improve the recognition accuracy.

[4]It is difficult to judge the results for graphs since the densities differ and usually can not be set to a specific value in advance.

		corr.	acc.
wrd	1–best	88.0	86.0
	graph (k=2)	93.7	92.9 (4.8 hyps/ref)
mor	**1–best**	**87.8**	**85.3**
	graph (k=2)	94.1	93.4 (6.6 hyps/ref)
	incl. pros.	**87.2**	**84.2**
rcb	mor 1–b	82.6	81.4 (2.1 hyps/ref)
	incl. pros.	82.2	80.9 (2.0 hyps/ref)

Table 3: Word (wrd) and morpheme (mor, including prosodic classifier) recognition results for 1–best and graph evaluation (graph density). Row 'rcb' shows the word recognition for the recombined morpheme sequences.

7. CONCLUSION

We presented a morpheme recognizer that was tightly coupled with a prosodic classifier. We outlined the training and the integration of the prosodic component and presented preliminary results. So far recognition accuracy has not significantly changed and future investigation will focus on improving the prosodic component and including right context for the prosodic prediction.

8. ACKNOWLEDGEMENT

This work was partly funded by the Deutsche Forschungsgemeinschaft DFG under Grant PO 441/3-1. The responsibility for the contents of this study lies with the authors.

9. REFERENCES

1. D. Gibbon and U. Ehrlich. Spezifikation für ein VERBMOBIL Lexikondatenbankkonzept. Verbmobil Memo Nr. 69, Universität Bielefeld, Daimler Benz AG, 1995.
2. M. Grice, M. Reyelt, R. Benzmüller, J. Mayer, and A. Batliner. Consistency in transcription and labelling of German intonation with GToBI. In *Proc. Int. Conf. on Spoken Language Processing*, Philadelphia, 1997.
3. K. Kohler, G. Lex, M. Pätzold, M. Scheffers, A. Simpson, and W. Thon. Handbuch zur Datenerhebung und Transliteration in TP14 von Verbmobil - 3-0. Verbmobil Technisches Dokument Nr. 11, Universität Kiel, 1994.
4. D. Steinbrecher. MSEG: Morphologische Segmentierung graphemischer und phonemischer Wortformen. Verbmobil Memo 99, Univ. of Bielefeld, 1995.
5. V. Strom. Detection of accents, phrase boundaries and sentence modality in German with prosodic features. In *Proc. European Conf. on Speech Communication and Technology*, volume 3, pages 2039–2041, Madrid, 1995.