AN INTEGRATED MODEL OF ACOUSTICS AND LANGUAGE USING SEMANTIC CLASSIFICATION TREES

E. Nöth, R. De Mori, J. Fischer,
A. Gebhard, S. Harbeck, R. Kompe,
R. Kuhn, H. Niemann, M. Mast

F.–A.–Universität Erlangen–Nürnberg

Report 128
Mai 1996

Mai 1996

E. Nöth, R. De Mori, J. Fischer,
A. Gebhard, S. Harbeck, R. Kompe,
R. Kuhn, H. Niemann, M. Mast

Lehrstuhl für Mustererkennung (Inf. 5)
Friedrich–Alexander–Universität Erlangen–Nürnberg
Martensstr. 3
D–91058 Erlangen

Tel.: (09131) 85 - 7888
e-mail: `noeth@informatik.uni-erlangen.de`

Gehört zum Antragsabschnitt: 3.11, 3.12, 10.1

Das diesem Bericht zugrundeliegende Forschungsvorhaben wurde mit Mitteln des Bundesministers für Bildung, Wissenschaft, Forschung und Technologie unter dem Förderkennzeichen 01 IV 102 H/0 gefördert. Die Verantwortung für den Inhalt dieser Arbeit liegt bei den Autoren.

AN INTEGRATED MODEL OF ACOUSTICS AND LANGUAGE USING SEMANTIC CLASSIFICATION TREES

E. Nöth[1] R. De Mori[2] J. Fischer[1] A. Gebhard[1] S. Harbeck[1] R. Kompe[1] R. Kuhn[3]
H. Niemann[1] M. Mast[1]

[1]Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, Martensstr. 3, 91058 Erlangen, Germany
[2]McGill University, School of Computer Science, 3480 University Street, Montréal, Canada H3A 2A7
[3]Centre de Recherche Informatique de Montréal (CRIM), 1801 McGill College, Montréal, Canada H3A 2N4
e–mail: noeth@informatik.uni-erlangen.de

ABSTRACT

We propose Multi–level Semantic Classification Trees to combine different information sources for predicting speech events (e.g. word chains, phrases, etc.). Traditionally in speech recognition systems these information sources (acoustic evidence, language model) are calculated independently and combined via Bayes rule. The proposed approach allows one to combine sources of different types - it is no longer necessary for each source to yield a probability. Moreover the tree can look at several information sources simultaneously. The approach is demonstrated for the prediction of prosodically marked phrase boundaries, combining information about the spoken word chain, word category information, prosodic parameters, and the result of a neural network predicting the boundary on the basis of acoustic-prosodic features. The recognition rates of up to 90% for the two class problem *boundary* vs. *no boundary* are already comparable to results achieved with the above mentioned Bayes rule approach that combines the acoustic classifier with a 5–gram categorical language model. This is remarkable, since so far only a small set of questions combining information from different sources have been implemented.

1. INTRODUCTION

Semantic classification trees (SCTs) [5] are a modification of the well known classification tree approach [1] intended to model natural language. They can be trained automatically using a labeled text corpus. Classification is done by moving from the root to a leave of the tree while asking at each node binary questions about e.g. strings of words. So far, we have successfully applied SCTs to the mapping of word sequences onto semantic classes in the context of language understanding [5], the classification of dialog acts [7], development of context dependent phone models [4], and the prediction of phrase boundaries and accents on the basis of the word sequence [4]. All of this research has been carried out in the context of automatic speech understanding. In state–of–the–art word recognition, different information sources such as acoustic and word sequence information are modeled separately. If the results of the sources can be interpreted as probabilities, then Bayes Rule can be used to combine the information sources. In this paper, we will show that SCTs are well suited for the integration of different knowledge sources in one model. We will show that SCTs can process information as different as automatically chosen word classes, the word identity itself (*categorical variables*), acoustic feature values (*continuous variables*), and classifier outputs (*probabilities, activation levels*, or *certainty factors*). As an example of an application, we employ enhanced SCTs to classify each word in an utterance as either preceding a prosodically marked phrase boundary or not. The models stored in the leaves of the SCTs can be seen as modeling the *a posteriori* probabilities of these two events. The data are part of the German Verbmobil database (VM, [9]).

The paper is organized as follows: in section 2 we introduce the standard SCT approach and the training procedure that we use. Then we give a description of the extensions, necessary for handling several information sources. Following this, we will show in section 3 what kind of prosodic information we combine with the linguistic information. We will present the experiments and finally in section 4 we will discuss the results and indicate our future work plan.

[1]This work was funded by the German Federal Ministry of Education, Science, Research and Technology (*BMBF*) in the framework of the Verbmobil Project under the Grant 01 IV 102 H/0. The responsibility for the contents of this study lies with the authors.

2. MULTI–LEVEL SEMANTIC CLASSIFICATION TREES

2.1. Semantic Classification Trees

In [5] Kuhn and De Mori describe SCTs and use them to solve problems involving Speech Understanding. They classify complete utterances (e.g. is an utterance a question about air travel fares?) or parts of an utterance (substrings; e.g. is a mentioned airport a start, stop or destination airport?). For this classification to take place, the temporal structure of an utterance (the sequence of the words $\mathbf{w} = w_1 w_2 \dots w_m$) is analyzed by means of regular expressions.

The structure of a binary SCT is as follows: each non–terminal node consists of a YES/NO–question, a YES–subtree and a NO–subtree, and each node is labeled with a probability vector for the recognizable classes. The classification of a word sequence $\mathbf{w}$ begins with the question at the root of the SCT. Depending on the answer to the question, the YES or NO subtree respectively will be entered. This process is repeated until a leaf node has been reached. The classification vector of this leaf is then assigned to $\mathbf{w}$.

A possible question q_1 at the beginning of the analysis (at the root node) is:

Has $\mathbf{w}$ *the structure* $+w_i+$ *?*

where $+$ is a non–zero gap (unknown sequence of words of length ≥ 1) and w_i a word of a given vocabulary. If the word w_i is exactly once in $\mathbf{w}$ and neither at the beginning nor at the end, q_1 is answered by YES[2]. In this case, the known structure of $\mathbf{w}$ is $+w_i+$. If q_1 is answered by NO, the known structure of $\mathbf{w}$ is the previously known structure,

[2]To analyze a sequence of words in a unique way by regular expressions, six different types of questions are needed: the *join-, left-, right-, unique-, twin-,* and *non-adjacent*-questions, as shown in [5]; in this example, q_1 is a *unique*-question.

i.e. "+" in the example. It is important to remark that a question handles exactly one unknown part (i.e. one gap of the known structure) of a sequence of words. Hence, a possible question q_2 to word sequences which have the structure indicated by a YES answer to q_1 is

Has **w** *the structure* $w_j + w_i +$ *?*

q_2 tests whether the first of the two gaps in $+w_i+$ has the structure w_j+.

The question at each node of a SCT is established by an automatic training process, explained in the next section.

2.2. The Training of SCTs

The training of the SCTs [2] is carried out by alternating expansion and pruning steps for an initial tree, using two labeled and disjunct training sets S_1 and S_2. At first, the initial tree is expanded using set S_1. The result is a tree T_1. This tree is pruned by means of set S_2 which gives a tree T_2. By expanding T_2 with S_2, a tree T_3 is created. Pruning T_3 with S_1 gives T_4.

This process generates a sequence of trees $T_1, T_2, \ldots, T_n$. It stops if two subsequently pruned trees T_{2i} and $T_{2(i+1)}$ have the same structure (i.e. they have the same number of nodes). The resulting SCT is $T_{2(i+1)}$.

For the expansion, two basic elements are needed [1]:

- a set of possible YES/NO–questions that can be applied to the items of the task domain,
- a rule for selecting the best question at any node or deciding that it should be a leaf node.

The set of possible questions is build up by employing all the words in the given vocabulary with the regular expressions proposed in [5]. The rule to select the best question is the *Gini* impurity criterion [1].

The expansion of a node n is done by assigning the best question q_n^* to the node, creating child nodes n_{YES} and n_{NO}, splitting the set of training items S_n into $S_{n_{YES}}$ and $S_{n_{NO}}$ according to q_n^* and expanding the child nodes with the training set $S_{n_{YES}}$ and $S_{n_{NO}}$ respectively. If the quality of the best question of a node is zero, this node is declared a leaf node and will not be expanded anymore. Each node is assigned the class of the most present pattern class in the node's training set (i.e., among all the data items that have passed through that node).

To prune the expanded tree with a disjoint training set, the following steps are carried out. First, each data item in this set is fed into the root and shuttled to the appropriate leaf; meanwhile, a counter at each node calculates the error rate of items passing through the node (i.e., how often the node's class differs from the class of an item in it). Next, in a recursion that moves upward from the leaves to the root, all YES and NO subtrees whose leaves yield higher error rates than the parent are deleted, with the parent being converted into a new leaf.

2.3. Multi–level Information

The only information used for the classification done by standard SCTs is the word sequence of the utterance. However, additional information can be attached to an utterance. Figure 1 shows an example of appending knowledge at several levels to an utterance. The information that each level contains is represented in one of the following ways:

- discrete features represent textual information (i.e. the spoken words on level *word*) as in the standard SCTs and/or categorical information, e.g. syntactic/semantic tagging information at the levels *syn/sem*, the accent judgement, which is represented in the example as an integer between *0 (not accented at all)* and *10 (strongly accented)* at the level *acc*, and

word	it	is	Okay	on	Friday
syn	pron	auxv	adv	prep	noun
sem	-	-	conf	-	time
acc	1	0	7	0	6
reg	.1	.5	-1.2	1.0	-.7
phr	.0	.2	.85	.15	1.0

Figure 1. Examples of different information sources for a Multi–level semantic classification tree.

- continuous features represent acoustic information extracted from the speech signal, i.e. the slope of the F0 regression line as an indicator for the sentence mood *question* at the level *reg*. Continuous features can also be the output of classifiers — in the example, the activation of a neural net to classify phrase boundaries is given at level *phr*.

The additional information is handled by questions about the values of the parameters. The possible types of questions depend on the representation of the information. In the case of

- discrete parameters: questions can be used in order to find out whether the value of a parameter is equal to a specific value;
- continuous parameters: questions can be used in order to find out whether the value of a parameter
  - is less than or equal to a specific threshold[3],
  - is the maximum/minimum in a segment of the utterance, or
  - is the absolute maximum/minimum.

We extended the SCT approach to handle not only the word sequence of an utterance but also to take into account additional information as shown above. We will call the enhanced SCTs *Multi–Level Semantic Classification Trees* (MSCTs).

Note that the SCT approach allows the processing of variable length input vectors, i.e. the number of information units that are presented to the SCT is solely determined by the length of the utterance. In most other classification approaches, like for instance neural networks, the number of information units (i.e. input nodes) is fixed.

2.4. The types of the questions

We will now have a closer look at the design of the questions that analyze the multi-level information attached to an utterance. The following example contains only the levels *word* and *phr* referred to in Figure 1:

yes i will come to Erlangen tomorrow
.9 .1 .0 .6 .0 .4 1.0
oh yes i will come to Erlangen tomorrow
.1 .9 .1 .0 .6 .0 .4 1.0

The two sentences of the example are nearly the same. One of the differences is that the position of the words in the first utterance is shifted by one compared with the words in the second utterance. Although the (semantic, syntactic, etc.) parameters of the two utterances are likely to be very similar (also just shifted by one), this similarity cannot be recognized by asking questions like:

[3] The value of the threshold is determined automatically during training.

is the value of the parameter at position m of level n equal to a specific value?
is the value of the parameter at position m of level n the maximum in a segment of the utterance?

This problem can be solved by taking advantage of the fact that the elements at the entrance level, which match a regular expression in any parent node are marked, and thus the temporal order of the elements from all the levels is known. Questions about other information levels only concern marked elements and they can use the information about the order. We will call this level the *entrance level.* Notice that the entrance level does not have to be the word level but could also be another categorical level, like *syn*. In asking questions like

Is the value of the parameter of level **reg** *of the n^{th} marked element on the entrance level $\leq k_1$?*
or
Is the value of the parameter of level **phr** *of the k^{th} marked word the maximum in a limited neighborhood?*

it might be possible to show the similarity of the two sentences in the example above.

Therefore the different question types are used in a specific order: apply the regular expressions to the elements of the entrance level. If an utterance has the structure asked for in the regular expression, the keyword(s) appearing in the regular expression is/are marked at level 1 of the utterance. Questions dealing with parameters located at the appended levels are only allowed if the word of the utterance in the column of the parameter is marked.

3. PROSODIC PHRASE BOUNDARY AND ACCENT DETECTION

As the task for the first experiments with the MSCTs we chose the problem of classifying major prosodic boundaries (henceforth $B3$), which in most cases mark clause boundaries [3]. The speech data used are spontaneously spoken turns obtained from face–to-face dialogs in the domain of appointment scheduling in the context of the German VM project. We consider this task as an interesting problem, where it seems useful to integrate information about the wording and acoustic features in a single model. A drawback of the VM database is, that so far not much training data is available. We also have prosodically labeled data available for a large corpus of read speech, but the perplexity of this corpus is so small that experiments using language models are not significant.

3.1. Material

For VM there are 25 dialogs labeled prosodically. Out of these we chose 22 for training (592 turns, 32 different speakers, 71 minutes of speech, 9336 words), and 3 for testing (80 turns, 4 different speakers). In [3] we report on experiments on word chains and word graphs using n–grams, where we used a subset of these turns. To keep the results comparable we chose the same subset of 48 turns as testing data for the experiments of this paper. These consist of 237 seconds of speech, 520 words, 74 $B3$, and thus 398 $\neg B3$ boundaries not counting the end of turns. The prosodic reference labels are based on perceptive evaluation done by non–naive listeners [8]. To exclude word recognition errors for this paper all experiments are based on the spoken word chain.

3.2. The Acoustic–prosodic Features

The computation of the features is based on a time alignment of the spoken words on the phoneme level using our HMM word recognizer. For each word final syllable the following prosodic features were computed from the speech signal for the syllable under consideration and for the two syllables in the left and the right context:

	MSCT input		% recognized	
Exp. no.	entrance level	additional features	total	$B3$
1	words	—	86.6	22.9
2	cat.	—	87.1	29.4
3	cat.	words	87.5	32.4
4	—	$B3$	87.9	36.5
5	cat.	words, $B3$	87.7	44.6
6	cat.	words, $B3$, A M, $FEAT$	90.3	44.5
7	words	$B3$, A, M, $FEAT$	89.6	54.1
8	MLP & SCT(cat)		88.6	40.5

Table 1. Results of the different MSCTs for $B3$ detection.

- the normalized duration of the syllable nucleus with respect to phoneme intrinsic mean and standard deviation and to the speaking rate as suggested by [10],
- the position of the F0 maximum on the time axis relative to the position of the syllable under consideration,
- the mean energy, and the mean F0,
- flags indicating if the syllable carries the lexical word accent or if it is in a word final position.

Furthermore the following features were computed only for the syllable under consideration:

- the duration of syllable and syllable nucleus using different normalization methods as well as no normalization at all (all together 6 features),
- the length of the pause (if any) preceding or succeeding the word containing the syllable,
- the linear regression coefficients of the F0 contour and the energy contour computed over different 7 and 2 windows respectively to the left and to the right of the syllable.

This yields the same 46 features used in the VM $B3$ classification experiments described in [3].

We tried to classify these features directly with MSCTs, but we yielded poor results due to the limited amount of training data. Thus we first trained a multi–layer perceptron (MLP) as acoustic–prosodic model using Quickpropagation to classify the features described above for the purpose of $B3$ classification (row *phr* in Fig. 1). The MLP had 40/20 nodes in the first/second hidden layer and one output node per class, in this case one for $B3$ and the other for $\neg B3$. Since we trained the MLP with the unity vector as desired output, the output values can in theory be considered as a posteriori probabilities (our experience is that the sum of the output activations in general is very close to one). However, in order to balance for the a priori probabilities of the different classes, during training the MLP was presented with an equal number of feature vectors from each class. Furthermore, the sum of the MLP outputs was normalized to be equal to one in any case.

Another MLP was trained to classify accented syllables on a similar feature set as described above (row *acc* in Fig. 1). A third MLP classifies F0 contours at the end of words into one of the classes rising, continuation–rise, falling; this is based on features obtained from the F0 contour and its linear regression line.

3.3. MSCT experiments

The results of seven experiments with MSCTs (no. 1–7) are depicted in Table 1. The first column specifies the experiment number to which it is referred to in the following; the second and third column specify the input features used (except for experiment 8); the fourth column shows the total recognition rate, whereas the fifth column gives the recognition rate of $B3$ alone. Note, that the recognition rates do not take into account the turn final boundaries, which to classify is a trivial task.

We used the following categorical input levels:

- the words itself (about 1200), and
- 150 categories (*cat* in Table 1 and henceforth) as for the n–gram (polygram) experiments described in [3], which were determined automatically in order to minimize the perplexity of a bigram.

Experiments were conducted with the *word* or the *cat* level being the entrance level. Note, that in the first case the *cat* level is of no further use. So far no explicit syntactic or semantic information (which could e.g. be computed by a parts–of–speech tagger) was used.

Furthermore, the following continuous input features were used:

- $B3$: the probability for a $B3$ boundary computed by the MLP as described above.
- A: the probability for a word being accented computed by an MLP. (The position of accents can indicate the phrase structure of a turn in addition to the prosodic boundary markers.)
- M: the probability for the intonation contour being rising, falling or continuation–rise. (These three types of contours roughly indicate the sentence mood.)
- $FEAT$: the following five features: normalized duration of syllable and syllable nucleus, F0 regression over two different windows, the mean F0 over the syllable. Note, that these are a subset of the features being input to the $B3$–MLP.

From experiment 1 to 7 the recognition rate of $B3$ improves with increasing experiment number. In most cases also the total recognition rate increases. In the following we will mention the $B3$ recognition rates only: in the first experiment only the words were used, yielding a recognition rate of 22.9%. This could be improved to 28.4% by using the categories instead of the words (exp. 2). So far only the "traditional" SCT approach with regular expressions as questions is used. When combining the *cat* level and the *word* level with the *cat* level being the entrance level, the recognition rate further improves to 32.4% (exp. 3). So far only discrete features have been used. In exp. 4 the $B3$ probability was used as the only feature; since the entrance level is undefined only numerical questions over the $B3$ probability attached to the word to be classified is used. Thus, the SCT more or less learns the a priori probability of $B3$. The recognition rate of this classifier is better than the one of the purely categorical ones (36.5%). Combining these different information sources, i.e. words, *cat*, and $B3$, in a single MSCT the recognition rate increases again to 44.6% (exp. 5). Using the A and M features in addition does not change the $B3$ recognition rate, but it improves the total recognition rate (exp. 6). When keeping these continuous features but switching to the words as entrance level, the different knowledge sources obviously are integrated in a more effective manner by the MSCT, which improves the recognition rate further to 54.1%.

In experiment 8 we combined the probabilities for $B3$ and for $\neg B3$ computed by the $B3$–MLP and by the SCT of exp. 3, which is a pure language model, via Bayes rule (see below) yielding a recognition rate of 44.4%. Note, that this is better than the result of exp. 5 with which it directly compares, however, the MSCT allows for the integration of much more different knowledge sources, which finally yields better results than the pure multiplication of probably bad probability estimates.

4. FUTURE WORK AND CONCLUSION

The main conclusion we can draw from our experiments is, that the integration of different knowledge sources including categorical and continuous features improves the recognition rate. However, the recognition rates are still somewhat lower than the ones we achieved with the combination of the B3–MLP and n–grams, which were reported in [3]. We believe that the main reason is the small amount of training data, which especially does not allow the MSCT to make use of a broad context within the questions in the nodes. On the training data the MSCT shows about 1/3 as much errors as on the testing data; this also indicates that the amount of training data is not sufficient.

During the experiments, we observed that adding more input features sometimes can reduce the recognition rate of the MSCTs a lot. This is caused by the optimization of the trees which is done locally on the current leaf nodes. This can cause a globally suboptimal question to be asked early, since it might split the training data best at that time. In an extreme case, the subtrees of this node could be identical, which causes optimization problems having only sparse training data. Of course with "unlimited" training data this problem does not exist.

We have implemented the training described in [6]. In the future we will compare this training with the one that we used so far [2]. Also we will soon have a large VM training database of about 7 hours that has syntactically/prosodically labeled phrase boundaries. This will allow us to verify our results on a large corpus, test out new question types, and look at the importance of individual questions in detail.

REFERENCES

[1] L. Breiman. *Classification and Regression Trees.* Wadsworth, Belmont CA, 1984.

[2] S. Gelfand, C. Ravishankar, and E. Delp. An Iterative Growing and Pruning Algorithm for Classification Tree Design. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:302–320, 1991.

[3] R. Kompe, A. Kießling, H. Niemann, E. Nöth, E.G. Schukat-Talamazzini, A. Zottmann, and A. Batliner. Prosodic scoring of word hypotheses graphs. *Proc. EUROSPEECH*, Vol. 2, pp. 1333–1336, Madrid, 1995.

[4] R. Kuhn, A. Lazarides, Y. Normandin, J. Brousseau, and E. Nöth. Applications of Decision Tree Methodology in Speech Recognition and Understanding. In *Proc. of the CRIM/FORWISS Workshop (München, 1994)*, pp. 220–232, Sankt Augustin, 1994. infix.

[5] R. Kuhn and R. De Mori. The Application of Semantic Classification Trees to Natural Language Understanding . *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17:449–460, 1995.

[6] D.M. Magerman. *Natural Language Parsing as Statistical Pattern Recognition.* PhD thesis, Stanford University, 1994.

[7] M. Mast, E. Nöth, H. Niemann, and E.G. Schukat-Talamazzini. Automatic Classification of Speech Acts with Semantic Classification Trees and Polygrams. In *IJCAI-95 Workshop "New Approaches to Learning for Natural Language Processing"*, pp. 71–79, Montreal, 1995.

[8] M. Reyelt and A. Batliner. Ein Inventar prosodischer Etiketten für VERBMOBIL, VM–Memo–33–94, 1994.

[9] W. Wahlster. Verbmobil — Translation of Face–To–Face Dialogs. *Proc. EUROSPEECH*, Vol. "Opening and Plenary Sessions", pp. 29–38, Berlin, 1993.

[10] C.W. Wightman. *Automatic Detection of Prosodic Constituents.* PhD thesis, Boston University, 1992.