Telephone Speech Recognition via the Combination of Knowledge Sources in a Segmental Speech Model

László Tóth [*], András Kocsor [†], and Gábor Gosztolya [‡]

Abstract

The currently dominant speech recognition methodology, Hidden Markov Modeling, treats speech as a stochastic random process with very simple mathematical properties. The simplistic assumptions of the model, and especially that of the independence of the observation vectors have been criticized by many in the literature, and alternative solutions have been proposed. One such alternative is segmental modeling, and the OASIS recognizer we have been working on in the recent years belongs to this category. In this paper we go one step further and suggest that we should consider speech recognition as a knowledge source combination problem. We offer a generalized algorithmic framework for this approach and show that both hidden Markov and segmental modeling are a special case of this decoding scheme. In the second part of the paper we describe the current components of the OASIS system and evaluate its performance on a very difficult recognition task, the phonetically balanced sentences of the MTBA Hungarian Telephone Speech Database. Our results show that OASIS outperforms a traditional HMM system in phoneme classification and achieves practically the same recognition scores at the sentence level.

1 Introduction

Although speech recognition requires the fusion of several information sources, it is rarely viewed as an expert combination problem. Such approaches were abandoned in favor of the Hidden Markov Modeling technique (HMM) [13], which treats speech as a stochastic process. The source of the success of HMM is that it offers a sound mathematical framework along with efficient training and evaluation. The price paid for this, however, is that the simplistic mathematical assumptions of the model do not accord with the real behavior of speech. One of these assumptions is the conditional independence of the spectral vectors. Several alternative models have been proposed, and one of these is the so-called segmental modeling

[*] Research Group on Artificial Intelligence of the Hungarian Academy of Sciences and University of Szeged, H-6720 Szeged, Aradi vértanúk tere 1., Hungary, e-mail: `tothl@inf.u-szeged.hu`

[†] Research Group on Artificial Intelligence of the Hungarian Academy of Sciences and University of Szeged, H-6720 Szeged, Aradi vértanúk tere 1., Hungary, e-mail: `kocsor@inf.u-szeged.hu`

[‡] Research Group on Artificial Intelligence of the Hungarian Academy of Sciences and University of Szeged, H-6720 Szeged, Aradi vértanúk tere 1., Hungary, e-mail: `ggabor@inf.u-szeged.hu`

approach [21]. Segmental models treat speech phonemes as one unit – instead of building them up from frames – and thus alleviate the flaw caused by the independence assumption. In recent years our team has been developing a segmental recognition system called OASIS. Our main observation so far is that, although the segmental representation indeed results in a somewhat better phoneme classification performance, these kind of recognizers also have similar robustness problems that traditional HMM systems have. A promising way for obtaining robust recognizers might be to treat speech recognition as a knowledge source combination problem. In this paper we give a generalized speech decoding algorithm created in this spirit. Moreover, we show that both HMM systems and segmental models can be viewed as a special case of this framework. In the second part of the paper we present the current components of the OASIS system, and then perform recognition experiments on the MTBA Hungarian Telephone Speech Database. Knowing that this database contains phonetically balanced sentences recorded from telephone calls from all parts of the country and from people of varying gender and age, this will be quite a difficult recognition task. The performance of the OASIS system will be compared to HTK, known as a sort of standard HMM recognizer in the speech community.

2 Speech Recognition as a Knowledge Source Combination Problem

Although speech recognition is obviously a pattern classification task, the most successful solution, Hidden Markov Modeling, is not a classification algorithm in the strict sense, but a generative model for stochastic random processes. This is because speech recognition does not fit the usual pattern classification framework. That is, most classification algorithms assume that the items to be classified are always represented by the same number of features. In addition, both the dimension of the feature space and the number of classes must be reasonably small. In contrast, speech is a continuous stream of acoustic information. Even if we assume that the talker must stop sometimes, the possible utterances vary in length and their number is practically unlimited. A possible solution is to trace the problem back to the recognition of some properly chosen building blocks. During recognition these building blocks have to be found, identified, and the information they provide needs to be combined. This approach turns speech recognition into a task of classifier combination integrated in a search process.

In the following we present a general speech decoding scheme in the spirit of classifier combination. Firstly, it makes it possible to experiment with alternative combination schemes which could not easily be done within the a traditional HMM framework. Secondly, it provides a more intuitive picture of how the whole recognition process actually works.

Algorithm 1 shows the pseudocode of our generalized speech decoder. Expressed simply, the algorithm works in the following way. Let us assume that our building blocks are denoted by the elements of the symbol set $\mathcal{F}$. Let the speech signal

Algorithm 1 A Generalized Speech Decoding Algorithm

```
solutions := $\emptyset$
hypothesis cue := $h_0(t_0, "", 0)$
// a hypothesis consists of a time index, a phoneme string, and a score
while there is an extendible hypothesis do
  select an extendible hypothesis $H(t, F, w)$ according to some strategy
  if $t = T$ then
    if only the first solution is required then
      return $H$
    else
      put $H$ on the list of solutions
    end if
  end if
  for $t' = t + 1, t + 2, \cdots$ do
    for all $f \in \mathcal{F}$ do
      $w_f := g_1(f, < t, t' >)$ // where $g_1$ estimates the cost of fitting $f$ to $< t, t' >$
                                  // based on the relevant $a_i$ measurements
      $w' := g_2(w, w_f)$ // where $g_2$ is a proper aggregation function
      if pruning-criterion($w_f, w'$) then
        construct a new hypothesis $H'(t', Ff, w')$ and put it in the hyp. cue
      end if
    end for
    if stopping-criterion($< t, t' >$) then
      break
    end if
  end for
end while
```

be given by the series of measurements $A = a_1, ..., a_T$. The goal of recognition is to map the speech signal A into a series of symbols $F = f_1...f_n$, where $f_j \in \mathcal{F}$. The algorithm works from left to right, and stores its partial results in a priority cue. Having processed the signal up to a certain point t, the algorithm looks ahead in time and, from the corresponding measurements, it collects evidence that the next symbol belongs to the time interval being inspected. As neither the exact length nor the identity of the next segment is known, we examine every time index $t' = t + 1, t + 2, ...$ that might be the end point of the segment. Each element f of the symbol set is matched to the interval $< t, t' >$, and from each (t', f) pair a new hypothesis is formed and put in the hypothesis cue. As every hypothesis has several extensions, this means creating a search tree. By adjusting the hypothesis selection strategy, the pruning and the stopping criteria one can control how the search space is traversed and pruned.

When the whole signal has been processed, the best scoring leaf is returned as the output result. The score of a hypothesis is calculated in two steps. First, there is a function (g_1) to combine the evidence for each symbol that was collected from the local information sources. Second, this local evidence is combined (via g_2) with the prefix of the hypothesis to obtain a global score. Thus, in effect, classifier combination occurs at two levels.

Obviously, we can have quite different decoders, depending on how the measurements a_i, the symbol set $\mathcal{F}$ and the functions g_1 and g_2 are chosen. Researchers agree only in that g_1 and g_2 should work on probabilistic grounds. In this case Bayes' decision theorem guarantees optimal performance, and statistical pattern recognition provides methods for approximating the probabilities from training corpora.

As regards the selection of the building units, the most reasonable choice is the phoneme since phonemes are the smallest information carrying units of speech (in the sense that the insertion/deletion/substitution of a phoneme can turn a word into another one). Furthermore, in many languages there is an almost one-to-one correspondence between phonemes and letters, so working with phonemes is an obvious choice when converting sound to written text. Nevertheless, smaller or larger units could be used as well. For example, there are arguments that syllables give a more suitable representation of (the English) language. Going the other way, current recognizers mostly decompose phonemes into three articulation phases [13].

The acoustic information sources a_i display the greatest variation from system to system. Traditionally the acoustic signal A is processed in small uniform-sized (20-50 ms) chunks called "frames", and the spectral representation of these serves as direct input for the model. It has been observed, however, that better results are obtained if this representation is augmented with features of longer time-spans so the feature vectors in current systems are a combination of the local and the neighboring 5-50 frames [13].

3 A Special Case: Hidden Markov Models

In spite of its unusual appearance, *Algorithm 1* is not so different from the standard technologies. In particular, its components can be chosen so that it becomes mathematically equivalent to the left-to-right Hidden Markov Models preferred in large-vocabulary speech recognition. In this setup the set of states of the Markov model will play the role of the symbol set in our algorithm. Although the states might simply represent the phonemes of the language, better results are normally obtained if the phonemes are decomposed into three states – one corresponding to the middle steady-state part, and two others describing the transitional phase before and after.

Instead of modeling the class posteriors $P(F|A)$ directly, in speech recognition the product $P(A|F)P(F)$ is normally modelled instead, which leads to the same result but allows one to separate the priors $P(F)$. Building words from states and assessing their prior probability is the problem of language modeling. Here we assume that $P(F)$ is readily given, and deal only with the acoustic component $P(A|F)$. This factor will be estimated by HMM in the way described below[1].

[1] Note that we slightly deviate from the standard decomposition into language and acoustic

During processing the HMM goes through a sequence of state transitions. This determines a segmentation based on how long the model stayed in a given state. The probability associated with a given segment sequence is calculated as follows. The probability corresponding to a given segment $S_i =< t, t' >$ and state f is calculated as

$$P(< t, t' > |f) = l_f^{(t'-t)} \cdot \prod_{i=t}^{t'} P(a_i|f), \tag{1}$$

where l_f is a constant between 0 and 1.

The probability corresponding to the whole segment sequence is obtained by multiplying the segmental probabilities:

$$P(A, S|F) = \prod_{i=1}^{n} P(S_i|f_i). \tag{2}$$

In terms of our model, Eq. (1) corresponds to g_1 while Eq. (2) corresponds to g_2. This means that g_2 is simply a multiplication, while g_1 consists of two factors. The term $l_f^{(t'-t)}$ is an exponentially decaying duration model. The product $\prod_{i=t}^{t'} P(a_i|f)$ is a spectral factor that renders a state-conditional likelihood for each measurement of the segment, and then combines these by multiplication – that is, by applying the naive Bayes assumption.

4 An Alternative Technology: Segmental Models

The contradiction between the model that assumes independence and the feature extraction method that makes it patently false has been understood and criticized by many authors [12][21]. Several cures were suggested, some of them only patching the original HMM algorithm, while some totally abandoning it. The family of segmental models [21] recommends modeling phonemes 'in one', instead of estimating their probabilities by multiplying the frame-based scores. In our framework this means that g_1 (see Eq. (1)) is replaced by some more sophisticated approximation[2]. There are several possibilities to parametrize phonetic segments as one unit. The most popular approach is to create special models that fit parametric curves on the feature trajectories [6][8][11][21]. However, it is also an option to convert the variable-length segmental data into a fixed number of segmental features. What makes this latter method tempting is that this way all the standard classification algorithms become applicable to the phoneme classification task. Thus, while the segmental trajectory models are usually built on Gaussian curves, representation by segmental features allows the use of almost any machine learning algorithm.

models as, in our notation, the state transitions between the states of a multi-state acoustic model are also included in the language factor, while only the self-transitions of a state are included in the acoustic model.

[2] In contrast to g_1, combination by multiplication at the g_2 level seems quite reasonable because the presence of all phonemes is required for the identity of a word. This makes an AND-like combination logical.

This is why we prefer this approach. In our studies we have reported experiments with a broad range of classifier methods, some of them being very new and not really known by the speech community [17]. Moreover, these classifiers allow the application of such linear and non-linear feature space transformation methods that are currently in the focus of machine learning research. We have published several papers that apply these groundbreaking techniques to phoneme classification [18]. The basic acoustic feature set we invented to represent phonetic segments is very similar to those used in the MIT SUMMIT system [7], but we have added several further features [17]. We have seen similar solutions from other authors, too [3].

A drawback of segmental systems is that the models trained to classify phonetic segments are not necessarily able to handle non-phonetic intervals which is required to find the proper segmentation of a signal. This problem was realized relatively lately [27]. One possible solution is to combine the segmental scores with a frame-based one that assesses the probability belonging to the given segmentation [27]. Another approach is to create artificial "anti-phone" examples and train a classifier to recognize these. We apply discriminative models (neural nets) for this goal, and presented the mathematical formulation of their application in [22]. Later we realized that our solution is similar to that employed in the SUMMIT system [7], but it is build on generative models (Gaussian mixtures) that requires a different formulation. Recently we have proposed another possible solution for the modeling of anti-phone segments by means of replicator neural nets [23].

5 Components of the OASIS Recognition System

The general decoding algorithm of Section 2 forms the core of the OASIS recognition system developed at our institute. That is, the decoding scheme of Algorithm 1 is performed by the 'Matching Engine' component of the system. In the following subsections we will describe in detail what the specific knowledge source components of the system are, how they work, and how they get integrated. As we shall see, the recognition methodology fits the general framework described above, and more actually belongs to the class of segmental models.

5.1 The Phoneme Classifier

The task of the phoneme classifier component is to map a probability to a given $(<t, t'>, f)$ segment-phoneme pair, that is to implement function g_1 of the general decoding scheme. For this we represent each (variable-length) segment by a fixed number of segmental features (for a description of the features see Section 6.2). These segmental features can theoretically be classified by any standard classification method that is able to produce a probabilistic output. Currently the system uses Artificial Neural Nets, but in earlier papers we described our investigations with many other classifiers as well [17]. Moreover, this component has the option of applying feature transformation algorithms prior to classification. We also wrote several reports on this [18]. In general we find that this modeling scheme results in a 10-30% reduction in the phoneme classification error compared to HMM. This is in accordance with the findings of other authors (see [3] and [11], for example).

5.2 The Anti-Phone Component

During recognition the algorithm will encounter such $< t, t' >$ segments that do not correspond to real phonemes. This may cause two big problems. First, the phoneme classifier might not automatically be able to report these segments. This is the case with our neural network classifier that returns phoneme posteriors and has no output for 'outlier' segments. The second problem is that a manually segmented training corpus contains only examples of real phonemic segments, so these 'anti-phone' segments cannot automatically be trained. One possible solution is to extend the phoneme classifier with an anti-phone class and artificially generate training examples for it [7]. Another option is to assess the probability of a segmentation from frame-based scores [27]. Our system applies the first approach, and the anti-phone probability of a segment is calculated by a complex method, as reported in [22].

5.3 The Language Model

The previous two knowledge sources were acoustic in type. But, of course, in most recognition tasks we have a very serious linguistic restriction on the possible phoneme sequences. The role of the language model is to provide the decoder with the possible phoneme sequences, along with their corresponding probabilities.

When designing the language model component of the OASIS system we initially followed the language description techniques of other recognizers. To be precise, we took the Microsoft Speech API as a starting point. It provides an XML description scheme for the definition of context-free grammars, the words themselves being the terminals of the language. However, in Hungarian listing all the agglutinated forms of a word stem is intractable. As luck would have it, Hungarian morphology can be well modeled by finite state systems. We observed that the agglutinated forms of a stem can be stored in a much smaller space with transducers than with a traditional compression algorithm. This led us to extend the SAPI description so that transducers could be embedded in the place of terminals. This results in a context-free grammar with its terminals being the words recognized by the transducer. Further compression can be achieved by applying special automaton compression algorithms which create the smallest possible transducer that models the same language [15]. Additional savings in storage are possible by storing the resulting transducer with a special data structure [16].

The SAPI handles probabilities by allowing the user to associate weights with the right hand side alternatives of a rule. The transducers embedded in our extended scheme also allow the weighting of the transitions. So, by combining the two levels, the system is able to associate a probability with any phoneme sequence.

The interface of the language model is adjusted to suit the requirements of the decoding algorithm. During the extension of a hypothesis the algorithm asks for the possible extensions of a phoneme sequence, so the task of the language model is to return all the possible subsequent phonemes of a prefix. Based on this, the interface of the language model consists of two functions, together making it possible to iteratively traverse all the phoneme sequences of the model. These functions are:

`Enter:` Returns the first possible extension of a prefix, along with its probability (or returns a null pointer if there is no extension).

`Next:` Return the next possible extension of the same prefix, along with its probability (or returns a null pointer if there are no more extensions).

As regards the technical details, the implementation of the storage and traversal of the transducers was relatively easy to do. Managing the context-free grammar, however, required the implementation of a stack automaton. We also had to store the actual values of the stack, which led to further technical complications.

5.4 The Combining of the Knowledge Sources

The decoding scheme of algorithm 1 is quite general and thus can be easily extended for the combining of more knowledge sources. An important practical issue is that the more sources we combine the more complex the problem of finding the optimal combination becomes. Fortunately, the problem of knowledge source combination has recently become an active research area. In addition, optimization techniques that support discriminative modelling are getting evermore popular in speech recognition [24]. One such possibility is the Discriminative Model Combination scheme of Beyerlein [1], which optimizes a combination scheme of the form:

$$P(F|A, L_1, ..., L_r) \approx \max_S \prod_i P(f_i|A, S)^{\alpha_0} P(f_i|L_1)^{\alpha_1} \cdots P(f_i|L_r)^{\alpha_r}, \qquad (3)$$

where we have r knowledge sources $L_1, ..., L_r$ voting on the symbols f_i in the form of posterior probabilities. Combining is then performed by raising the values to a power and multiplying them.

The OASIS system uses Eq. (3) for the combination of the three components. The optimal exponents of Eq. (3) are found by a global optimization algorithm called SNOBFIT [14]. To make the recognition process more efficient we apply multi-stack decoding with several search tree pruning heuristics [9].

6 Experimental Results on the MTBA Database

The goal of this section is to demonstrate the effectiveness of the OASIS system on a real recognition task. For this we chose the phonetically rich sentences of the MTBA Hungarian Telephone Speech Database, because it presents a very general and challenging problem for the acoustic component of any recognizer. Furthermore, this is currently the largest available speech corpus for Hungarian, and very few results have been reported on it so far. Unfortunately, this recognition task is too general in the sense that there was no way of applying any complex language model. Hence, the tests reported leave the language model component of the system practically unexploited, and assess the performance of the acoustic components only.

6.1 The Corpus

The MTBA Hungarian Telephone Speech Database is the result of an IKTA project carried out in 2001-2003 by the Department of Informatics, University of Szeged, and the Department of Telecommunications and Media Informatics, Technical University of Budapest [28]. Besides other recordings, the database contains 6000 manually labeled and segmented sentences. This part of the corpus was designed so that the phonetic transcript of the sentences contains all possible phoneme pairs that occur in Hungarian. Moreover, the phone callers were organized so that the recordings covered the whole of the country, and the callers were distributed in age and gender. These factors altogether present a very difficult and general recognition task.

For the experiments we first selected those sentences from the database that contained no significant noise and/or half-cut phonemes (denoted by [spk] and [cut] symbols in the phonetic transcript). From the remaining sentences 1367/687 randomly chosen ones were used for training and testing, respectively. These contained 68333/34532 phoneme instances.

6.2 Acoustic Features and Phoneme Classification Scores

For the classification of segments the system applies a 3-layer feed-forward neural net with 200 hidden neurons and a softmax output layer. The net is trained with the minimum cross-entropy training criterion, and training is stopped according to a cross-validation criterion [2]. To find a proper segmental feature set we started from a rather simple representation and gradually extended it with further features. The findings were as follows.

Baseline features. As a traditional frame-based representation, energies in 18 Bark-bands were calculated (via FFT, with triangular weighting and cube root compression) at a frame rate of 333 frames/sec[3]. As the neural net used for segmental classification requires a fixed number of inputs, a conversion is necessary into a fixed-dimensional segmental feature set. At this stage we followed the very simple idea of the SUMMIT system [7]: the band energies were averaged over phoneme thirds, which means a kind of non-uniform smoothing. We may say that the inputs to the neural net are really just average energies in cells that tile the time-frequency space in a special manner.

The Importance of Phoneme Duration. In Hungarian most phonemes have a 'short' and a 'long' counterpart, thus duration seems to be a vital piece of information. To model the duration we extended the baseline feature set with

[3] This is about three times more than the usual 100 frames/sec. We used this increased value because in many experiments we found that it resulted in a slightly better classification performance.

a further feature containing the length of the segment. This way the neural net had the opportunity to form any kind of durational description, according to the data. The introduction of the duration feature resulted in a significant error rate reduction, as shown in the table below.

Classification error rate	
Baseline features	Baseline plus duration
47.72%	42.15%

Channel Normalization and Gain Control. The variance in the transfer characteristics of telephone lines is known to have a detrimental effect on speech recognition. A somewhat similar issue is the varying amplitude of the signal. Many normalization techniques have been suggested to counter these effects. Some of them are off-line, which means that they work *after* the whole signal has been recorded (and, consequently, are not suitable for real-time recognition). The on-line algorithms base their processing on the last couple of (centi)seconds. The methods studied do this by means of a 1-pole lowpass filter with time-constant τ.

As the results in the table below show, off-line methods performed slightly better than on-line ones. Out of the on-line methods the non-linear AGC was the best, with a time-constant of 1 second.

Off-line methods	CER%
Mean and dev. normalization (full spectrum)	40.27%
Mean and dev. normalization (per channel)	37.75%
On-line methods	
RASTA filtering	43.86%
Mean and dev. norm. (per channel, $\tau = 250ms$)	41.12%
Mean and dev. norm. (per channel, $\tau = 1sec$)	40.36%
Nonlinear AGC (per channel, $\tau = 250ms$)	39.64%
Nonlinear AGC (per channel, $\tau = 1sec$)	38.49%

Adding Observation Context. In fluent (and fast) speech phones may become so short that they cannot be recognized without their observation context. Auditory research suggests that approximately a 220-250 ms interval contains information about the identity of a phone, but some researchers use observation windows as large as one second [10]. We tried three different settings of the observation length, defined as the phoneme length plus the context length. This means the a variable-sized observation context was considered, depending on the segment size. The context was represented by its average energy values in each Bark-band, thus resulting in two additional feature 'columns' on both sides of the phonemes. As the table shows below, the shortest observation length (150ms) performed best, but this might be due to the large variance of the context over the training set.

	Classification error rate		
Normalization	$\tau = 150msec$	$\tau = 250msec$	$\tau = 1sec$
Off-line mean and dev. norm.	33.18%	34.49%	36.12%
Nonlinear AGC (*1sec*)	33.51%	34.85%	36.25%

Adding Onset and Offset Detectors. Human hearing has cells tuned to detect signal onsets and offsets. These onset and offset detectors may play an important role in the segmentation of a sound stream, especially in finding the boundaries of (certain) phonetic segments. So we implemented an algorithm to simulate these detectors, based on the directions described in [4]. Our detectors calculate the derivatives of the Bark-band energy trajectories and sum them over 3 (6-Bark wide) channels. These curves were evaluated at the phone start and end points and their values were added to the feature set as further features.

We sought to combine these features only with the best feature set found so far. The result shown below indicates that these new features brought only a marginal improvement in the classification scores. We should mention, however, that they proved very important in the phone/antiphone component of the recognition system.

CER with onset/offset feat. (off-line norm., 150ms obs.cont.)	32.17%

6.3 Modeling Anti-Phones

For separating real phonemic segments from the anti-phones, a two-class neural network was used. The segmental feature set was similar to that of the phoneme classifier, but instead of the means of the band energy averages, in this case the *variances* were used. This was a new idea and brought a slight improvement over our previous setup that utilized the same feature set in both the phoneme classifier and the anti-phone model. In all other respects the generation of the anti-phone training examples and their utilization in the decoding process followed the scheme that we presented in [22]. Hence we will refrain from repeating the mathematical formulation here.

Although the anti-phone model could be evaluated in isolation if we generated test examples similar to the generation of the training data, its effect can only really be assessed by its influence on the decoding process. Our results for this will be given in the next subsection.

6.4 Language Model

As the vocabulary of the sentences in the corpus is not restricted in any sense, there was no option of applying any sophisticated (word or morpheme-based) language model. The only thing we could do was to work with a statistical model, e.g. phoneme N-grams. From these we chose the simplest possible one, that is every phoneme was allowed at every position and with the same probability.

6.5 Phoneme Recognition Results

The evaluation of the recognition results is performed by comparing the manual phonetic transcription of a sentence to the transcription generated by the recognizer. Clearly, the recognizer output may contain substitution, insertion and deletion errors as well. To count these the two strings are matched by calculating their edit distance with weights (4,3,3) for substitutions, insertions and deletions, respectively. These weights were proposed by the HTK toolkit [29]. The scores reported below were calculated using the formula

$$Correct = \frac{N - S - D}{N}, \tag{4}$$

where N is the number of all phoneme instances and S and D are the number of substitutions and deletions, respectively. Obviously the recognizer can increase this value by producing many insertion errors as the number of insertions is not accounted for in the formula. To prevent this, the number of insertions was forced to stay around 10-12% by suitably punishing phone transitions in the aggregation formula. This value was suggested by [20].

The table below lists the recognition scores obtained with and without applying the anti-phone model. The figures clearly show the importance of the anti-phone component. Whether these scores are good or not is difficult to judge per se, so in the following section we will furnish some possible bases of comparison.

Sentence-Level Recognition Scores	
Without anti-phones	With anti-phones
53.44%	61.34%

7 Related Work

To our knowledge, apart from us only three teams have used the MTBA corpus so far. Unfortunately, the TSP Lab of the Technical University of Budapest and Hexium Ltd. have performed only isolated word or connected word recognition tests over a restricted vocabulary [5][25]. Although the LSA Lab of the Technical University of Budapest has experimented with a task similar to our setup, in their tests both the train/test division of the data and the phonetic label set were slightly different. Hence, the phoneme recognition score of 55-60% they reported [26] allows only a gross comparison.

To obtain a more precise base for comparison we trained the HTK Toolkit [29], which is a freely available HMM-based recognizer, and is very frequently used to obtain a baseline result when evaluating new technologies. The HTK recognizer was trained with 3-state monophone phoneme models, all having 15 diagonal Gaussian components (this was reported to be about optimal in [5]). Naturally the same train/test setup and phonetic labeling was used as with the OASIS system, and the language model was also set up in a similar way. For signal processing we applied the standard 39-component MFCC vector proposed by the HTK manual. With these settings HTK recognized 61.60% of the phonemes correctly, with an insertion error rate very close to the one obtained with the OASIS system. This means that our system is capable of practically the same recognition performance as other common recognizers.

Unfortunately, HTK cannot measure phoneme classification directly, so we could not obtain comparative scores to assess the performance of the phoneme classifier module in isolation. However, in an earlier paper we had the option of comparing the phoneme classifier of our system to an HMM-based recognizer for a number recognition task [17]. Moreover, in another paper we tested our phoneme classifier on the TIMIT corpus, for which several classification results are available in the literature [19]. In both cases we found that our segmental representation (along with an ANN or SVM classifier and suitably chosen transformation methods) yields slightly better results than the conventional HMM technology.

8 Discussion and Conclusions

The basic motivation for segmental speech modeling is to replace the simple and incorrect independence assumption of HMM's with a more sophisticated combination scheme. The phoneme classification results indeed show that HMM can be outperformed by even a very simple segmental representation. When it comes to recognition, however, one finds that segmental models need an additional component to handle outlier segments. In our system this problem is handled by the anti-phone models. Our recognition results show that this component can bring up the performance of the system to the level of a traditional HMM recognizer. However, it appears that the gain of better phoneme classification is still lost during decoding, so further developments are required to outperform the current technology. To improve the recognition scores we experiment with alternative anti-phone modeling techniques [23]. Besides this, the addition of further knowledge sources along with the automatic tuning of the parameters in Eq. (3) seems a promising direction. Decomposing the phoneme classifier into many localized experts may improve both performance and robustness, and is a very real trend in speech recognition. Our generalized decoding algorithm provides a good framework for these topics of study, so we plan to investigate them in the near future.

References

[1] P. Beyerlein, Discriminative Model Combination, Proc. ICASSP'98, pp. 481-484., 1998.

[2] C. M. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, 1995.

[3] P. Clarkson and P. J. Moreno, On the Use of Support Vector Machines for Phonetic Classification, Proceedings of ICASSP'99, pp. 585-588, 1999.

[4] E. F. Evans, Modelling Characteristics of Onset-I Cells in Guinea Pig Cochlear Nucleus, Proceedings of the NATO ASI on Computational Hearing, pp. 1-6, 1998.

[5] T. Fegyó, P. Mihajlik and P. Tatai, A Comparative Study on Hungarian Acoustic Model Sets and Training Methods, Proc. Eurospeech 2003, 2003.

[6] H. Gish, K. Ng, A Segmental Speech Model With Applications To Word Spotting, Proceedings of ICASSP'93, pp. 447-450, 1993.

[7] J. R. Glass, A probabilistic framework for feature-based speech recognition, Proceedings of ICSLP'96, pp. 2277-2280, 1996.

[8] Y. Glass, J. P. Haton, Stochastic Trajectory Modeling For Speech Recognition, Proceedings of ICASSP'94, pp. 57-60, 1994.

[9] G. Gosztolya and A. Kocsor, Improving the Multi-stack Decoding Algorithm in a Segment-based Speech Recognizer, Proc. 16th Int. Conf. on IEA/AIE 2003, LNAI 2718, pp. 744-749, Springer Verlag, 2003.

[10] H. Hermansky, Modulation Spectrum In Speech Processing, In: A. Prochazka et al. (eds.), Signal Analysis and Prediction, Birkhauser, pp. 385-398., 1998.

[11] W. J. Holmes, M. J. Russel, Probabilistic-trajectory Segmental HMMs, Computer Speech and Language, V.13, pp. 3-37, 1999.

[12] K. S. Van Horn, A Maximum-entropy Solution to the Frame-dependency Problem in Speech Recognition, Tech. Rep., Dept. of Computer Science, North Dakota State Univ., Nov. 2001.

[13] X. D. Huang, A. Acero and H-W. Hon, Spoken language processing, Prentice Hall, 2001

[14] W. Huyer, A. Neumaier, SNOBFIT - Stable Noisy Optimization by Branch and Fit, Submitted

[15] A. Kertész-Farkas, Z. Fülöp and A. Kocsor, Magyar nyelvű szótárak tömör reprezentációja nemdeterminisztikus automatákkal, Proc. MSZNY, pp. 231-236, 2003. (in Hungarian)

[16] G. A. Kiraz, Compressed Storage of Sparse Finite-State Transducers, *Proc. of WIA'99* , LNCS Vol. 2214, pp. 109-122, Springer, 2001.

[17] A. Kocsor, L. Tóth, A. Kuba Jr., K. Kovács, M. Jelasity, T. Gyimóthy, J. Csirik, A Comparative Study of Several Feature Space Transformation and Learning Methods for Phoneme Classification, Int. J. Speech Technology, Vol. 3, 3/4, pp. 263-276, 2000.

[18] A. Kocsor and L. Toth, Application of Kernel-Based Feature Space Transformations and Learning Methods to Phoneme Classification, accepted for Applied Intelligence

[19] A. Kocsor and L. Toth, Kernel-Based Feature Extraction with a Speech Technology Application, accepted for IEEE Transactions on Signal Processing

[20] K.-F. Lee and H.-W. Hon, Speaker-Independent Phone Recognition Using Hidden Markov Models, IEEE Trans. ASSP., Vol. 37, No. 11, Nov. 1989.

[21] M. Ostendorf, V. Digalakis and O. A. Kimball, From HMMs to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition, IEEE Trans. ASSP, 4:360–378., 1996.

[22] L. Tóth, A. Kocsor and K. Kovács, A Discriminative Segmental Speech Model and its Application to Hungarian Number Recognition, Proc. TSD'2000, Springer Verlag LNAI Vol. 1902, pp. 307-313, 2000.

[23] L. Tóth and G. Gosztolya, Replicator Neural Network for Outlier Modeling in Segmental Speech Recognition, Proc. ISNN 2004, Springer Verlag LNAI Vol. 3173, pp. 996-1001., 2004.

[24] R. Schlüter, W. Macherey, B. Müller and H. Ney, Comparison of discriminative training criteria and optimization methods for speech recognition, Speech Communication, Vol. 34., pp. 287-310., 2001.

[25] Csaba Szepesvári, personal communication

[26] Szabolcs Velkei, personal communication

[27] J. Verhasselt et al., Assessing the Importance of the Segmentation Probability in Segment-Based Speech Recognition, Speech Recognition, Vol. 24, (1), pp. 51-72, 1998.

[28] K. Vicsi, L. Toth, A. Kocsor, G. Gordos, J. Csirik, MTBA - magyar nyelvű telefonbeszéd-adatbázis, Híradástechnika, Vol. LVII, NO.8, 2002 (in Hungarian)

[29] S. Young et al., The HMM Toolkit (HTK) (software and manual), http://htk.eng.cam.ac.uk/