

PHONEME SEGMENTATION OF CONTINUOUS SPEECH USING MULTI-LAYER PERCEPTRON

Youngjoo Suh and Youngjik Lee

Electronics and Telecommunications Research Institute
Yusong P. O. Box 106, Taejeon, 305-600, KOREA
E-mail: yjsuh@zenith.etri.re.kr

ABSTRACT

In this paper, we propose a new method of phoneme segmentation using MLP(multi-layer perceptron). The structure of the proposed segmenter consists of three parts: preprocessor, MLP-based phoneme segmenter, and postprocessor. The preprocessor utilizes a sequence of 44 order feature parameters for each frame of speech, based on the acoustic-phonetic knowledge. The MLP has one hidden layer and an output layer. The feature parameters for four consecutive inter-frame features (176 parameters) are served as input data. The output value decides whether the current frame is a phoneme boundary or not. In postprocessing, we decide the positions of phoneme boundaries using the output of the MLP.

We obtained 84 % for 5 msec-accuracy and 87 % for 15 msec-accuracy with an insertion rate of 9 % for open test. By adjusting the threshold value of the MLP output, we achieved higher accuracy. When we decreased the threshold by 0.4, we obtained 5 msec-accuracy of 92 % with insertion rate of 3.4 % for the insertions that are more than 15 msec apart from phoneme boundaries.

1. INTRODUCTION

Segmentation of continuous speech into its corresponding phonemes has become very important issue in many speech processing areas such as speech recognition, speech analysis, speech synthesis, and speech database. Accurate and reliable automatic phoneme segmentation is the crucial factor in satisfying the application requirements. A variety of methods have been proposed to accomplish this phoneme segmentation [1][2][3][4]. Although some of them showed acceptable performances, most of the methods rely heavily on a series of rules derived from acoustic phonetic knowledge. However, because these rule-based methods are very complex and hard to optimize their parameters efficiently, the performances degrade severely in the real application. In order to overcome these drawbacks, another method of phoneme segmentation adopting neural networks has been proposed. This neural network-based approach has several advantages over the conventional rule(or knowledge)-based method. Because it is not a parametric model, it produces robust performance under the unexpected environmental variations or the presence of noise, and also needs

not make assumptions about the underlying analysis target. With these advantages, several neural network based attempts have been made for the phoneme segmentation and obtained some encouraging results[5][6].

In this paper, we propose a new method of MLP-based phoneme segmentation which utilizes the differences between adjacent speech frames and adopts the modification of MLP learning algorithm. The organization of this paper is as follows. We begin by introducing the architecture and learning algorithm of the proposed MLP-based phoneme segmenter for continuous speech in section 2. In section 3, we describe the experimental procedure and about the results with discussion about the implication of these results. Finally, a brief conclusion including a further study is given in section 4.

2. MLP-BASED PHONEME SEGMENTATION

In order to improve the drawbacks of already existing rule-based methods, we adopted an MLP-based neural network approach in the phoneme segmentation of continuous speech. We regarded the detection of phoneme boundary as a kind of pattern classification. MLP-based approaches have shown their notable nonlinear discrimination capability in this pattern classification area[7]. To use this ability, we attempted to introduce well-defined features to make patterns representing the boundary and nonboundary of phonemes. We then trained the MLP to learn the capability of detecting these boundaries and segmenting continuous speech into their corresponding phonemes. The followings are the details of the proposed algorithm.

2.1. An MLP Architecture for Phoneme Segmentation

The architecture of our MLP-based phoneme segmenter consists of three parts: preprocessor, the MLP, and postprocessor. The preprocessor extracts feature parameters and has two stages of the following procedure. The first stage is to extract the features from each speech frame and the second stage is to re-extract the final features as the differences of two adjacent frame features. We restricted all features to the FFT-derived features. This is because we began to develop the phoneme segmenter to be

served as a part of our segmentation-based phoneme recognizer, and thus we needed to use the same kind of features both in phoneme segmentation and recognition, to avoid excessive computational loads for feature extraction. To handle the phonemes having different duration and abruptness, we used 10 msec frames in addition to 16 msec frames, and applied Hamming window with a shift rate of 10 msec. Then, a sequence of 44 order features suitably representing the acoustic-phonetic characteristics of human speech is extracted from each frame. These primary features are as follows.

- MFE: FFT based 16 order mel-scaled filter bank energies(16 order).
- ENG_FRM: Normalized frame energy(1 order).
- ENG_RTO: A ratio of low(0-3000 Hz) to high frequency band(3000-7500 Hz) energy (1 order).
- F_POS: The position of the first(F1), second(F2), third(F3), and fourth(F4) formant - residing mel band(4 order).
- F_AMP: The amplitude of F1, F2, F3, and F4 residing mel band energy(4 order).

To derive the formant-residing frequency bands, we defined the ranges of each formant frequency as follows. Based on the acoustic-phonetic knowledge, the F1, F2, F3, and F4 reside in the frequency bands of 0-1000 Hz, 1000-2400 Hz, 2400-3000 Hz, and 3000-4000 Hz, respectively. From these, the corresponding formant-residing mel-scaled bands are 1-8 for F1, 9-18 for F2, 19-21 for F3, and 22-24 bands for F4 in our extended 31 mel-scaled frequency bands.

From the above primary features, we re-extract the final features for phoneme segmenter input. Since signal variations are more prominent at the phoneme boundary, these variations can be good cues in the phoneme segmentation. To use this fact, we choose the final features, inter-frame features, as the differences between two adjacent frame features. All these inter-frame features are then normalized to lie between -1 and +1 to be used in the MLP.

The MLP in the MLP-based phoneme segmenter has one hidden layer and an output layer. The 176 feature parameters for four consecutive inter-frame features are finally served as input data because of their superior performance in the experiments. The output layer has a single node that decides whether the current frame, that is, the frame between the second and third inter-frame, is phoneme boundary or not. In the hidden and output layer, we use sigmoid as an activation function. In the hidden layer, the number of hidden nodes are changed through the experiments.

In postprocessing, the positions of phoneme boundaries are decided using the output value of the MLP. When the output of the MLP is greater than the threshold value, the position of the third frame, that is, the position between the second and third inter-frame, is regarded as a phoneme boundary.

2.2. Learning Algorithm

We adopted the modified Error Back Propagation method as a learning algorithm[8]. This algorithm has the same criterion of minimizing the mean-squared error but converges much faster than the commonly used Error Back Propagation method. The target data have the value of +1 at the phoneme boundary and -1 or similar value in other position. Four consecutive frame features are applied to the MLP and then shifted by one frame to learn all cases of speech input patterns. The learning rate is set to 0.0005 and the initial weight values are randomly generated with the range of $-5.0E-7$ to $5.0E-7$ for all cases.

3. EXPERIMENTAL PROCEDURE AND RESULTS

The accuracy and consistency of training data affect the output performance considerably in neural network based experiments. Thus, to achieve good results, large amount of elaborate training data is highly recommended. In our case, the training data is the speech database segmented by phonemic unit. In the following section, we will start by describing this speech data used in our experiments. We then explain the details of the experimental procedure and conclude with the results achieved by the proposed method.

3.1. Speech Database

To train and test the MLP based phoneme segmenter, we used a Korean read speech database. This database is uttered by one female speaker and then manually labeled by phoneticians. It contains 156 sentences, 2185 words, and 16,057 phones. All the data were sampled at 16 kHz sampling rate with 16 bit resolution per sample and processed to extract feature parameters. We used about 88 % (14,227 phones) of the total speech data in the training and the remaining data (1,830 phones) in the evaluation of the segmenter.

3.2. Segmentation Experiment and Results

In the earlier stage of our experiments, we implemented a baseline phoneme segmenter that has smaller number of input nodes and simple standard structure. To improve the performance of the segmenter, we added more useful features and modified the learning method. The three different types of feature parameters used in our experiments are as follows.

- FEA_A(18): MFE, ENG_FRM, ENG_RTO from 16 msec frames.
- FEA_B(26): MFE, ENG_FRM, ENG_RTO, F_POS, F_AMP from 16 msec frame.
- FEA_C(44): MFE, ENG_FRM, ENG_RTO from 16 and 10 msec frames, and F_POS, F_AMP from 16 msec frames.

The number in the parenthesis after the name of features represents the order of each feature. In order to detect the boundaries of short duration phonemes, we introduced a sequence of features extracted from the frame of 10 msec length and added to make FEA_C. With the baseline segmenter having 13 hidden nodes, we made experiments with respect to the different feature types. The results are given in Table 1. We obtained 15 msec accuracy of 53.6 % with insertion rate of 3.5 % for the baseline segmenter and this represents the addition of short term features is very helpful in improving performance. The accuracy is defined as the ratio of the number of correctly detected boundaries to that of total boundaries. The insertion rate is the ratio of the number of frames incorrectly decided as boundary to that of total nonboundary frames.

Feature type	5 msec accuracy (%)	15 msec accuracy (%)	Insertion (%)
FEA_A	35.2	46.6	5.1
FEA_B	36.7	48.2	3.9
FEA_C	42.8	53.6	3.5

Table 1: The segmentation accuracy with different type of features

We introduced two methods to improve the accuracy of our baseline segmenter. The first method is the modification of target data form in learning procedure. Because the transition from one phoneme to its following phoneme takes at least a couple of frames in normal cases, the assignment of the target value of 1 at the frame of phoneme boundary and -1 for the adjacent frame is inappropriate. To accommodate this slowly varying transition effect, we modified the target data by the following form. Instead of applying -1 to the adjacent frame directly contacting with the frame of phoneme boundary, we assigned -0.01 to train the effect of phonemic transition near the boundary. So the target data form of three consecutive frames having phoneme boundary at their center position is (-0.01, 1.0, -0.01) compared with the previous values of (-1.0, 1.0, -1.0). The results from this modification of the target data form are given in Table 2

Feature type	5 msec accuracy (%)	15 msec accuracy (%)	Insertion (%)
FEA_A	58.4	69.9	16.9
FEA_B	62.4	73.0	14.3
FEA_C	70.3	79.5	9.4

Table 2: The segmentation accuracy with the modified form of target data (-0.01, 1.0, -0.01)

Compared with the results in Table 1, the segmentation accuracy was increased more than 23 % but the insertion rate was increased by three times. When we analyzed the MLP output pattern, we found that considerable amount of insertions was due to the consecutive insertions near the correct phoneme boundary.

The second method for the improvement of segmentation

accuracy is the adjustment of boundary position in the target data at the learning procedure. This method is conceived by the fact that the segmentation is done manually by phoneticians. Although these phonetic experts segment speech accurately with useful acoustic-phonetic knowledge, it is really difficult to do the task without any loss of consistency. To solve this problem, we moved the position of phoneme boundary in the target data. The detailed algorithm is as follows. After the predetermined learning iteration, the output scores of the MLP are computed for three consecutive frames including phoneme boundary at their center (second) position. The position producing the highest score is then assigned as a new phoneme boundary. This readjustment is performed at every learning iteration based on the updated MLP model. In the case of the repetition of successive boundaries with very short phoneme duration, we did not apply the above readjustment algorithm. We made a series of experiments changing the iteration time of starting the adjustment. The best result was obtained at the second iteration. The results for these experiments are given in Table 3 and 4 for two different forms of target data .

Feature type	5 msec accuracy (%)	15 msec accuracy (%)	Insertion (%)
FEA_A	61.6	64.5	12.8
FEA_B	62.5	65.7	11.0
FEA_C	70.2	72.7	7.1

Table 3: The segmentation accuracy using the adjustment of boundary position in target data with target data form of (-1.0, 1.0, -1.0)

Feature type	5 msec accuracy (%)	15 msec accuracy (%)	Insertion (%)
FEA_A	70.6	75.9	13.3
FEA_B	72.6	77.3	12.4
FEA_C	81.6	85.5	8.7

Table 4: The segmentation accuracy using the adjustment of boundary position in target data with target data form of (-0.01, 1.0, -0.01)

From the results in Table 3 and 4, we know that adjusting the position of phoneme boundary in target data has the effect of increasing the segmentation accuracy as well as reducing the insertion rate. By applying the proposed three methods we obtained much higher performance than that of the baseline segmenter.

All the above results are evaluated after 100 learning iteration of the MLP training. When we increased the iteration to 3000, the best result was 84.2 % and 87.2 % for 5 and 15 msec accuracy with insertion rate of 9.4 % respectively. The segmentation decision is made by the threshold value of the MLP output and lower threshold value produces higher segmentation accuracy at the expense of increased insertion rate. When we decreased the threshold value by 0.4, we obtained 92.3 % with an insertion rate of 16.8 %. Since the phoneme transition can endure more than

