Combination of diverse subword units in spoken term detection

Shi-wook Lee¹, Kazuyo Tanaka², Yoshiaki Itoh³

¹ National Institute of Advanced Industrial Science and Technology, Japan
² Tsukuba University, Japan
³ Iwate Prefectural University, Japan
s.lee@aist.go.jp, tanaka.kazuyo.qb@u.tsukuba.ac.jp, y-itoh@iwate-pu.ac.jp

Abstract

This paper focuses on the following two points: First, we try to clarify the effect of combination systems from two aspects, accuracy and heterogeneity. And then we evaluate our unique subword unit, called Sub-Phonetic Segment (SPS) to maximize performance improvement by combination. Combination systems usually yield higher performance than any individual system. When the systems being combined are individually accurate but also mutually heterogeneous, the improvement by combination can be maximized. From this consideration, we estimate heterogeneity by correlation of false alarm errors of combined systems and confirm that lower correlation of two systems yields the better performance improvement by combination. Comparative tests of several combination approaches are carried out on subword-based spoken term detection. Since subword-based systems use constrained linguistic knowledge, it is fairly straightforward to verify the heterogeneity of combined systems. Experimental results show that the most significant improvements can be achieved by combination of two different subword units, triphone and SPS, which are highly heterogeneous subword units with low correlation of false alarm detections.

Index Terms: spoken term detection, keyword search, system combination, phonetic recognition, diversity

1. Introduction

Spoken term detection (STD), one of the fundamental applications of Automatic Speech Recognition (ASR) is to find a query term from a huge spoken database. As multimedia is increasingly becoming the biggest big data, the need for accurate STD system is greatly increased. From the first STD project by the U.S. National Institute of Standards and Technology (NIST) in 2006 [1,2], many projects are interested in STD task, for example the IARPA’s Babel project [3,4,5,6], the DARPA’s RATS program [7], and the Japanese NTCI R project [8]. As a result, a number of studies have been reported over the past few years [1-13,16,17,18,20,22].

Since large vocabulary continuous speech recognition (LVCSR) is fundamentally based on a fixed vocabulary, any word in spoken data that does not exist in the lexicon, i.e., an out-of-vocabulary (OOV) word, will be misrecognized as an alternate that has similar acoustic features. So LVCSR systems cannot simply permit searching for OOV words. An alternative approach to solve the OOV problem is to use subwords [9], such as SPSs, phonemes, syllables, graphophones, and morphemes. There have been introduced two ways of making subword, subword phonetic recognition results [10,11] or subword description of the word-based LVCSR [12,13]. Here, subword-based STD is highly generalized without any restriction of a query. However, compared to word-based LVCSR, the poor accuracy of subword phonetic recognition has to be considered.

There has been considerable research in the possibility of building multiple systems and then combining them to obtain better performance in machine learning applications. The method of system combination has been adopted in LVCSR known as ROVER [14] and Statistical Machine Translation (SMT) [15]. Recently, a system combination method is also adopted in STD. There are two different approaches in system combination for STD, according to where the combination is carried out. One is that, previous to detection process, multiple hypotheses from one or more ASR systems are combined to make confusion networks [16,17]. The other is that, after detection is carried out in multiple systems, the results of multiple systems can be merged by calibrating score and reranking [5,6,11,18].

Many approaches have been focused on combining results by different systems, where ad-hoc combination strategies are used and empirically compared. However, there has been rare research why the combination can be helpful to improve performance and how to maximize the performance improvement by system combination. Niyogi [19] designed multiple systems through a procedure that directly minimizes the correlation of their respective errors. Mangu [20] exploited diversity for STD, where the system combination is carried out with diverse components, such as acoustic model, decoding strategy and audio segmentation. Here, we argue that diversity for combination is effective when two systems have similar high accuracy and high heterogeneity concurrently. High heterogeneity is evaluated by low correlation of false alarm errors in STD. In order to maximize performance improvement by combination, we propose a new structure of subword to satisfy the above two necessary conditions, high accuracy and high heterogeneity. And also, since subword based approaches are constrained to use linguistic knowledge, such as stochastic language model, combination of diverse subword units can be ultimate to solve the OOV problem.

The organization of this paper is as follows: Section 2 describes the continuous dynamic time warping as subword-based detection scheme and score fusion by linear interpolation. Section 3 describes diverse subword units and the other various factors for comparative evaluations. From experimental evaluation in section 4, combination with a new subword unit can maximize the diversity and yield better improvement than the other combination approaches, which are carried out by stochastic n-gram language model, different features and bagging. Finally, we conclude in Section 5.
2. Subword-based spoken term detection

2.1. Continuous dynamic time warping

Since subword phonetic recognition is generally less accurate than word-based recognition, continuous Dynamic Time Warping (DTW) has been applied to deal with recognition errors and has been proved to be suitable in STD [11,22]. The DTW matches subword sequence of a query to subword sequence of the spoken database to be searched. A query term is usually the part of sentence utterance, so the subword sequence of a query is much shorter than that of reference sentence. The DTW carries out the local matching process, of which the first subword of a query term is continuously shifted over the reference sentence. Dissimilarity between a query and reference is calculated as

\[
G(q,r) = \arg \min \left\{ \begin{array}{l}
G(q-1, r-1) + 3 \cdot D(s_q, s_r) \\
G(q-2, r-1) + 3 \cdot D(s_{q-1}, s_r) + 3 \cdot D(s_q, s_r) \\
G(q-1, r-2) + 2 \cdot D(s_q, s_{r-1}) + 1 \cdot D(s_q, s_r)
\end{array} \right. \tag{1}
\]

where \(G(q,r)\) denotes the cumulative dissimilarity of query subwords \(s_q\) up to reference subwords \(s_r\). At the last subword of a query, the portion that \(G(q,r)\) is less than a threshold value is detected as a spoken term. The three paths are concerned to substitution, deletion, and insertion and are weighted. And \(D(*)\) in Eq. (1) is the local dissimilarity between two subwords, which uses a previously calculated confusion matrix from an acoustic model. The Bhattacharyya distance is adopted to calculate the above distance \(D(*)\), which is a measure of the probabilistic similarity between Gaussian mixture models (GMM) and has been proven to be effective for STD [11,21]. Since the GMM and multi-states of a hidden Markov model (HMM) are used, many Bhattacharyya distances can be calculated between two subword HMMs. From the preliminary experiments [22], the distance of the minimum pair, i.e., the closest pair of Gaussian distributions, results in the best performance. This is because the minimum distance can be considered to represent the degree of confusion between two probabilistic mixture distributions. In the final scoring of the DTW, the score is first normalized by the number of subwords of a query, and then the detected candidates are ranked by their scores.

2.2. Score fusion by linear interpolation

Lee [25] showed that different systems retrieve similar sets of relevant documents but retrieve different sets of nonrelevant documents and then evaluated several score fusion. The key effect of combination is considered that since some of false alarm errors in STD have very different scores in each system, their scores should be lower by score fusion and the false alarm errors should be moved to a lower rank by the combination. For the analytic approach in this paper, linear interpolation of Eq. (2) is simply adopted for the combination of \(N\) systems.

\[
\hat{s}_t = \sum_{i=1}^{N} a_i \cdot s_{t,i}, \quad \text{where} \sum_{i=1}^{N} a_i = 1. \tag{2}
\]

Here, \(t\) is the detected candidate, \(i\) denotes the subsystem, \(s_{t,i}\) is the score of the detected candidate \(t\) for system \(i\), and \(\hat{s}_t\) is the merged score of the detected candidate \(t\). The weight, \(a\) is experimentally chosen for the best performance.

3. Diversity for combination

3.1. Heterogeneous subword units

The conventional context dependent phoneme, triphone is contextually expanded by contiguous phonemes but has the same duration of the phoneme. On the other hand, SPS becomes a more finely divided unit on the time axis, where one phoneme is divided into stationary and non-stationary parts. Fig. 1 shows the graphical description of phoneme, triphone, and SPS. It can be seen that the time division is different. As shown in the right side of Fig. 1, SPS is moved from the original phoneme and spans over two contiguous phonemes in the feature space, whereas triphone is slightly moved and is mainly located on the same region of the original phoneme. Here, three reasons can be considered why SPS is intuitively heterogeneous to triphone. First, though both SPS and triphone are modified and extended from phoneme, their areas are different in feature space. Triphone is moved slightly from the origin phoneme according to their contextual connection. The modification for SPS is that core-SPS of stationary part is focused on the centered area of phoneme, which is less influenced by coarticulation, while transition-SPS of non-stationary part is merged area of two connected phoneme in feature space. These different modifications in feature space lead to different acoustic models and different acoustic likelihood probabilities. Second, since SPS has shorter interval on the time axis, the recognition errors of SPS are substantially different from those of triphone. Third, the probabilities of subword language model are added more frequently in SPS.

3.2. Other methods for comparative experiments

3.2.1. Speech features

A simple approach for generating multiple systems to be combined is to use the different speech features, which can be available from different feature extraction scheme. Two most general speech features are Mel Frequency Cepstral Coefficients (MFCC) and Perceptual Linear Prediction (PLP) [23], and most of current ASR systems are developed on these
two features. Here, we first evaluate the system combination by using these two different speech features, where the subword phonetic recognitions are based on PLP and MFCC features, respectively.

3.2.2. Subword bigram language model

As one of diversity, we evaluate the performance whether subword n-gram language model is adopted or not. An n-gram model is widely used in LVCSR and is proven to improve the accuracy. Therefore, subword n-gram language model can also be used in subword phonetic recognition. In this work, subword bigram models are constructed by the subword description of speech database, which is used for training acoustic model. Furthermore, in order to solve the OOV problem, it is helpful to analyze the effect of using linguistic knowledge.

3.2.3. Bagging

The straightforward way of making complementary systems is to manipulate the training data, which is known as bagging, an abbreviation of bootstrap aggregating [24]. Bagging builds several subsystems on different, randomly selected subsets of the training data. In this work, we train 10 subsystems by randomly selected 10% samples of the original training data and then gradually combine those subsystems to the baseline system trained on all training data.

4. Experimental evaluation

4.1. Spoken term detection task

The NTCIR-10 SpokenDoc-2 STD task [8] is used for experimental evaluation. It consists of a total of 104 oral presentations (28.6 hours) for the spoken data to be searched. 100 queries (53 OOV and 47 in-vocabulary) and their relevant segments are also provided. Each query text can be uniformly converted into subword sequence, and the subword phonetic recognition is used to make subword sequence of the spoken data. To train GMM-HMM for the subwords, such as 463 SPSs and 10,325 triphones, 187 hours of speech from the corpus of spontaneous Japanese database are used [26]. Each state of a subword HMM both for MFCC and for PLP consists of 38 dimensional feature vector and 8 Gaussian mixtures.

4.2. Evaluation metric

For evaluating performance, we first compute the precision and recall rates for each query \( q \) and each rank \( j \). And then we summarize the above precision and recall rates into F-measure, which is defined as:

\[
F(q, j) = \frac{2 \cdot \text{Precision}(q, j) \cdot \text{Recall}(q, j)}{\text{Precision}(q, j) + \text{Recall}(q, j)}
\]

Finally, we average all maximum F-measure of queries and then multiply the result by 100 to obtain a single value as a percentage, which is referred to as average of maximum F-measure (AMF):

\[
AMF = 100 \cdot \frac{1}{Q} \sum_{q=1}^{Q} \max_{j} F(q, j)
\]

where \( Q \) is the number of queries and \( j \) is the rank of each detected result. Compared to Term Weight Value (TWV) [1], AMF is calculated without any weight and simply balanced between recall and precision.

4.3. Experimental results and discussion

Table 1 shows the baseline performance by single STD systems according to several factors, such as features, subwords and the usage of bigram language model. It can be seen that there is no significant performance difference between MFCC and PLP, and also between triphone and SPS. However, the results show that the performance is improved significantly by using a bigram language model. The best performance of 60.30 AMF is achieved by using SPS and MFCC with bigram language model.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Subword &amp; LM</th>
<th>AMF</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFCC</td>
<td>TRI</td>
<td>51.44</td>
</tr>
<tr>
<td></td>
<td>SPS</td>
<td>47.56</td>
</tr>
<tr>
<td></td>
<td>TRI_LM</td>
<td>58.86</td>
</tr>
<tr>
<td></td>
<td>SPS_LM</td>
<td>60.30</td>
</tr>
<tr>
<td>PLP</td>
<td>TRI</td>
<td>46.41</td>
</tr>
<tr>
<td></td>
<td>SPS</td>
<td>49.51</td>
</tr>
<tr>
<td></td>
<td>TRI_LM</td>
<td>56.82</td>
</tr>
<tr>
<td></td>
<td>SPS_LM</td>
<td>59.24</td>
</tr>
</tbody>
</table>

In the combining stage, the top 10,000 hypotheses of each query are selected from each system, and then the common set of the hypotheses is selected for the final combined hypotheses, where the selection is performed by time alignment with 60 millisecond margin of the start and end time and the score is calibrated by linear interpolation of Eq.(2).

First, the experiment by Bagging method is carried out by using MFCC, triphone and bigram language model. Ten subsystems trained by randomly selected 10% samples of the training data are gradually combined into the single baseline system in descending order of their AMF.

As shown in Fig. 2, the combined system is better than the single baseline system. The AMF is gradually increased as subsystems are combined, though all individual subsystems are worse than the single baseline system. However, the effect of system combination is becoming smaller and the
improvement is finally converged. The performance of 61.72 AMF from the combination of all 10 subsystems is better than the performance of 58.86 AMF of the single baseline system by 4.85% improvement. By the bagging method, each individual subsystem is slightly heterogeneous from each other, so the improvement can be achieved by majority voting.

Table 2. Experimental results by combination, focused on whether language model is adopted or not. The correlation coefficient (Corr.), performance (AMF) and improvement (Imp.) are shown. The improvement is calculated to the best AMF of subsystems.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Sys. #1 w/o LM</th>
<th>Sys. #2 with LM</th>
<th>Corr.</th>
<th>AMF</th>
<th>Imp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFCC</td>
<td>SPS</td>
<td>SPS</td>
<td>0.78</td>
<td>61.89</td>
<td>2.63</td>
</tr>
<tr>
<td></td>
<td>TRI</td>
<td>TRI</td>
<td>0.82</td>
<td>62.95</td>
<td>6.94</td>
</tr>
<tr>
<td></td>
<td>SPS</td>
<td>TRI</td>
<td>0.62</td>
<td>65.14</td>
<td>10.66</td>
</tr>
<tr>
<td></td>
<td>TRI</td>
<td>SPS</td>
<td>0.66</td>
<td>66.90</td>
<td>10.94</td>
</tr>
<tr>
<td>PLP</td>
<td>SPS</td>
<td>SPS</td>
<td>0.78</td>
<td>62.31</td>
<td>5.18</td>
</tr>
<tr>
<td></td>
<td>TRI</td>
<td>SPS</td>
<td>0.83</td>
<td>57.55</td>
<td>1.28</td>
</tr>
<tr>
<td></td>
<td>TRI</td>
<td>SPS</td>
<td>0.62</td>
<td>63.16</td>
<td>11.16</td>
</tr>
<tr>
<td></td>
<td>SPS</td>
<td>SPS</td>
<td>0.64</td>
<td>62.25</td>
<td>5.08</td>
</tr>
</tbody>
</table>

By the combination of whether subword bigram language is used or not, the performance is slightly improved using the same subword. However, with the different subwords, the improvements are significantly high, such as 66.90 AMF with MFCC and 63.16 AMF with PLP. Compared to the result of Bagging method in Fig. 2, the 66.90 AMF by the combination of TRI and SPS_LM is highly better than the 61.72 AMF of the final combined system of all 10 subsystems and also better than the 60.30 AMF of single system of SPS_LM in Table 1. The results show that the combination of using different subwords is quite effective.

Table 3. Experimental results by combination of different features and different subwords with language model.

<table>
<thead>
<tr>
<th>Sys. #1</th>
<th>Sys. #2</th>
<th>Corr.</th>
<th>AMF</th>
<th>Imp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFCC_TRI_LM</td>
<td>PLP_TRI_LM</td>
<td>0.81</td>
<td>60.73</td>
<td>3.18</td>
</tr>
<tr>
<td>MFCC_SPS_LM</td>
<td>PLP_SPS_LM</td>
<td>0.87</td>
<td>61.95</td>
<td>2.74</td>
</tr>
<tr>
<td>MFCC_TRI_LM</td>
<td>MFCC_SPS_LM</td>
<td>0.65</td>
<td>68.50</td>
<td>13.59</td>
</tr>
<tr>
<td>PLP_TRI_LM</td>
<td>PLP_SPS_LM</td>
<td>0.63</td>
<td>66.28</td>
<td>11.88</td>
</tr>
<tr>
<td>MFCC_TRI_LM</td>
<td>PLP_SPS_LM</td>
<td>0.64</td>
<td>67.44</td>
<td>13.84</td>
</tr>
<tr>
<td>MFCC_SPS_LM</td>
<td>PLP_TRI_LM</td>
<td>0.64</td>
<td>67.52</td>
<td>11.97</td>
</tr>
</tbody>
</table>

As shown in Table 3, the combinations of different subwords between triphone and SPS significantly improve the performance. However, with the same subword, the second and third row in Table 3, the performance is slightly improved, 3.18% for triphone and 2.74% for SPS. These results show that the heterogeneity between MFCC and PLP is low, and so the combination of these two different features is not so much effective. Finally, the best performance of 68.50 AMF can be achieved with the combination of SPS and triphone by 13.59% improvement over 60.30 AMF of single systems in Table 1.

From Table 2 and 3, the proposed SPS is very effective in combination to construct a complementary system. It can be also observed in the following Fig. 3. The figure shows that the correlation for query-to-query becomes lower from the black linear regression line by combining of the same subword to the red linear regression line by combining of different subwords. The lower correlation yields higher improvement.

Throughout all experiments in Fig. 4, it has been obviously examined that as the correlation of false alarm errors becomes lower, the performance improvement becomes much higher. Furthermore, the experimental results show that system combination is effective, not just because the systems are different, but because similar high accuracy and high heterogeneity of systems are provided.

5. Conclusions

Performance improvement in STD can be achieved by the combination of systems. The main contribution of this paper can be concluded as follows: first, high heterogeneity can be measured with low correlation of false alarm errors and maximize the improvement by combination. This result gives an idea for how to identify the best combination and determining whether to combine any system. Next, a new structure of subword, SPS is quite effective to improve the performance by combining with triphone. This is because SPS is expanded in time while triphone is expanded on linguistic concept, the speech feature space can be interpreted in very different ways and the effect of system combination can be maximized.

6. Acknowledgements

This research is partially supported by Grand-in-Aid for Scientific Research (C), KAKENHI, Project No. 15K00241.
7. References


