Subword Speech Recognition for Detection of Unseen Words

Ivan Bulyko, José Herrero, Chris Mihelich and Owen Kimball

Raytheon BBN Technologies, Cambridge MA 02138, USA
{ibulyko,jherrero,cmihelic,okimball}@bbn.com

Abstract

We present a novel approach to building a subword speech recognizer for the task of phonetic keyword search. The recognizer, which uses short fixed-length phonetic units, is trained with phonetic transcripts that are segmented into all possible substrings of 1, 2 and 3 phones, using a lattice representation to accommodate the overlapping units. We compare the keyword search accuracy of the proposed system with systems that use words, graphones, variable-length phonetic units and context-dependent phones. Experiments with Spanish CTS data show that the proposed subword recognizer outperforms other subword systems in terms of phonetic keyword search accuracy measured on queries that consist of words not present in the training data.

Index Terms: speech recognition, keyword search, OOV

1. Introduction

Spoken term detection (STD) or keyword search (KWS) is an important application of speech recognition technology. A substantial research literature supports the use of large vocabulary continuous speech recognition (LVCSR) for accurate keyword search. Specifically, using lattices generated by an LVCSR system has been shown to be crucial for achieving high recall and improved accuracy in KWS [1,2]. In addition to accuracy, fast retrieval speed is important in KWS, but searching lattices presents a computational challenge. Consensus networks [3] offer a compact representation of the lattice, encapsulating the full richness of recognition hypotheses in a structured linear form, which can be exploited by search algorithms [4]. In the past, consensus networks have been shown to perform as well or better than lattices in KWS tasks [5]. The keyword search system described in this paper uses consensus networks because of the computational advantages.

Out-of-vocabulary (OOV) words, i.e., those not in the recognizer’s dictionary, can never appear in the lattice and therefore must be retrieved using subword units such as syllables and phones. Word lattices can be converted to their corresponding subword units, or one can build a subword or a hybrid word-subword recognition system [6] that generates lattices with subword units. Alternatively, one can build a generic word model to phonetically represent the whole class of OOV words [7].

Various types of subword units have been proposed in the past. Graphone units [8] were successfully used in open-vocabulary speech recognition and text-to-phoneme conversion. Variable-length phonetic units or multigrams have been applied to a variety of tasks that include OOV detection and recovery in speech recognition [9], vocabulary-independent voice search [10] and spoken term detection [6]. In [9], following the work of [11], the authors define the unit inventory by training an n-gram phone LM of high order and pruning the LM using the relative entropy criterion. In [10] the multiphone units are found by computing the mutual information between the phonetic subsequences and the word labels. In [6] the authors use constrained multigram units, where unit segmentation is chosen to maximize the likelihood of the training data.

Several studies have compared different subword units in terms of their effectiveness in OOV detection and recovery [12], voice search [13], and STD [14]. There appears to be a consensus that a hybrid recognizer, that contains words and subword units outperforms a pure word-based or a subword-only recognizer on these tasks. In this work we focus on detection of words that are not observed in training. We introduce a novel subword recognizer that exceeds the keyword search accuracy of graphone-based and variable-length phonetic unit recognizers when tested with queries that contain out-of-training keywords.

The remainder of this paper is organized as follows. Section 2 describes the keyword search system. In Section 3, we present the various types of recognition units used in our experiments, including the proposed new type of fixed-length units. Our experimental setup is explained in Section 4, followed by the definitions of the evaluation metrics in Section 5. Experimental results are presented in Section 6. We conclude with a summary and some future directions in Section 7.

2. Keyword Search System

Our keyword search is based on BBN’s Byblos LVCSR system [15], which uses state-of-the-art discriminatively trained acoustic models and performs MLLR speaker adaptation. The output of the recognizer is a word lattice, which we transform into a consensus network [3] for the purpose of running keyword search. Words present in the recognizer’s dictionary (IV terms) can be retrieved directly from these consensus networks.

For OOV queries we employ phonetic search. The recognition models can use units of any granularity, such as words, subwords and/or phones. We convert the lattices produced by the recognizer to phone level by splitting each lattice arc into a sequence of phone arcs that correspond to the unit of the original arc. We then convert the resulting phone lattices to phonetic consensus networks.

Phonetic search is accomplished by finding the query term (represented by a sequence of phones) in the phonetic consensus network. A match is returned when we find a sequence of arcs in the consensus network that correspond to the target phone sequence. We used a trainable text-to-phoneme (T2P) model to generate pronunciations for OOV queries. The T2P model was trained in three steps (see [16] for details): 1) the characters and phonemes in the training dictionary are aligned using an iterative EM-based algorithm, 2) contextual features are extracted from the alignments and 3) a decision tree mapping characters to phonemes is built based on the contextual features.
In our previous work [4] we showed that recall can be improved significantly by allowing approximate matches within the consensus network to be returned, where insertions, deletions or substitutions within the target are evaluated with some predefined probability of mismatch \( P_m \). We align the target query represented by a sequence of phones to the consensus network using a dynamic programming (DP) algorithm similar to the one used in computing word error rate. The algorithm returns the best alignment, measured by the product of 1) consensus network arc posteriors (including skip arcs) used by the alignment, and 2) probability of mismatch \( P_m \) for each instance of deletion, insertion or substitution.

Each hit returned by the DP search is assigned a confidence score which is computed using a Generalized Linear Model (GLM). The input to GLM is a fixed-length vector of numerical features. For each search result, the GLM computes a weighted sum of the input features and applies the sigmoid function to the sum. We used features that are language-independent and easy to compute from the consensus networks during the DP search: number of phonemes in the target; the geometric mean of confidences and the product of confidences in the best alignment, either including or excluding phoneme errors; the number and fraction of target phonemes that were matched; and the number and fraction of phonemes that did not align to the consensus network. We have also tried using phonetic and lexical features but only saw benefit in cases where query words were very short (e.g. 3 phonemes or less) or very frequent.

### 3. Recognition Units

In this work we built and tested several keyword search systems varying in the size and type of recognition units being used. We evaluated five types of basic units: 1) words (i.e. the standard LVCSR models), 2) context-dependent phones, 3) hybrid word-graphone units, 4) variable-length phonetic units, and 5) fixed-length phonetic units with flexible segmentation. The latter represents a novel approach to building a subword recognition system. Below we describe the different types of units in more detail.

**Context-dependent (CD) phones.** The recognizer’s dictionary consists exclusively of the individual phonemes. Context dependency is modeled by the underlying triphone acoustic models and the 3-gram phonetic language model.

**Graphone units** were introduced in [8] for text-to-phone conversion and for open-vocabulary speech recognition. Each unit is a pair of a letter sequence and a phoneme sequence of possibly different lengths. Units of varying lengths are automatically learned from the dictionary and then a trigram joint-sequence model is used to segment infrequent words in the training data into subword units. Finally, this flat hybrid word-graphone segmentation is used for training of the STT models (see example in Figure 1a). In our implementation we set the maximum graphone length to 3 and we kept the 2000 most frequent words intact in the training data. (The first three words in Figure 1a are frequent words left as-is in the graphone unit inventory.)

**Variable-length phonetic unit** inventory is constructed by iteratively joining most frequent unit bigrams, starting with single phonemes and stopping when all unit pairs are below some minimum frequency threshold (in our experiments we set the threshold to 50 occurrences) or when the desired number of units is reached. Note that unlike the graphone units, the variable-length units are not constrained by word boundaries, hence a phone sequence that spans two or more words may become a single unit, as illustrated in Figure 1b, where the 3-word phrase “YO SOY DE” became a single phonetic unit.

**Fixed-length units** are phonetic sequences with a limited maximum length. Choosing the desired unit length entails the tradeoff between the extended context span of long units and better ability to generalize with short units. When training resources are limited long units may be susceptible to data sparsity, hence shorter units are needed in the ASR dictionary to generate new sequences in the output. In this work we introduce a novel approach to building a fixed-length subword recognition system, where we segment the phone sequence in each training utterance into all possible substrings of 1, 2 and 3 phones, without any constraints. All units found in the training data are added to the recognizer’s dictionary.

In order to accommodate all segmentations, including overlapping units, we use the lattice representation, as illustrated in Figure 1c. Flexible lattice-based segmentation allows a seamless integration of units of differing granularity into a single language model, where 3-phone units are modeled in the context of 2- and 1-phone units and vice versa. Thus the model can benefit from the increased length of context provided by the longer units, and, at the same time, leave room for greater generality offered by shorter units, which is especially useful for recognition of unseen words. Finally, considering all segmentations of the phone sequence reduces data fragmentation and allows units of each length to take advantage of the entire training set.

#### Table 1. Characteristics of different unit inventories.

<table>
<thead>
<tr>
<th>Units</th>
<th>Dictionary size</th>
<th>Average phones/token in training</th>
</tr>
</thead>
<tbody>
<tr>
<td>Words</td>
<td>30801</td>
<td>3.64</td>
</tr>
<tr>
<td>Graphones</td>
<td>14282</td>
<td>3.12</td>
</tr>
<tr>
<td>Var.-length units</td>
<td>33332</td>
<td>3.12</td>
</tr>
<tr>
<td>Fixed-length units</td>
<td>5048</td>
<td>2.00</td>
</tr>
<tr>
<td>CD Phones</td>
<td>33</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Two basic characteristics of the different unit inventories: the dictionary size and average token length in training, are presented in Table 1. As may be expected, the word dictionary is the largest and leads to the longest tokens. It is interesting to note, that graphones and variable-length units resulted in the same average token length, but the graphone inventory is significantly larger, because we kept all subword units derived from the dictionary regardless of whether they appeared after segmenting words in the training transcripts.
4. Experimental Setup

The acoustic and language models used in these experiments were trained with 100 hours of the Fisher Spanish conversational telephone speech (CTS). We used the 4-hour Spanish CallHome (Eval97) test set to measure the recognition and keyword search accuracy. On this test set the LVCSR system achieved a 1-best WER of 49.39% and a lattice-oracle WER of 26.08%. For the phonetic, subword and hybrid systems we will report recognition performance only in terms of phone error rate (PER).

For keyword search query terms, we used single words from the test set reference transcripts. The query terms were grouped into four disjoint sets, based on the LVCSR dictionary and the word frequency in training:

- **IV-rare5**, rare in-vocabulary words that occur between 5 and 10 times in the training data;
- **IV-rare1**, rare in-vocabulary words that occur only 1 or 2 times in training;
- **IV-OOT**, in-vocabulary words that do not have any instances in training (out-of-training);
- **OOV**, out-of-vocabulary words that also do not occur in training.

Table 2 shows various characteristics of the query lists. Since all queries are rare words, they have very few occurrences in the test set. The average length of queries (in phonemes) does not vary significantly across the four lists. We used a trainable text-to-phoneme (T2P) model to generate pronunciations for OOV queries. T2P phone accuracy was 97%, measured on the full 30K-word dictionary using 5-fold cross-validation.

Table 2. Various characteristics of the query lists.

<table>
<thead>
<tr>
<th>Recognition units</th>
<th>IV-rare5</th>
<th>IV-rare1</th>
<th>IV-OOT</th>
<th>OOV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of terms</td>
<td>355</td>
<td>379</td>
<td>224</td>
<td>412</td>
</tr>
<tr>
<td>Avg. occurrences in train.</td>
<td>6.82</td>
<td>1.41</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Avg. occurrences in test</td>
<td>1.61</td>
<td>1.34</td>
<td>1.37</td>
<td>1.33</td>
</tr>
<tr>
<td>Avg. term length in phones</td>
<td>7.14</td>
<td>7.28</td>
<td>7.29</td>
<td>7.34</td>
</tr>
</tbody>
</table>

5. Keyword Search Evaluation

For each set of queries, we report keyword search accuracy in terms of three different metrics: **AUC**, **ATWV** (Actual Term Weighted Value) with automatically set confidence threshold, and **OTWV** (TWV with optimally set threshold). **AUC**, which stands for “area under curve”, is a variant of the mean average precision (MAP) metric, defined as

\[
AUC = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{N(q)} \sum_{r \in R(q)} p(q, r)
\]

where \( Q \) is the set of queries with non-empty references, \( N(q) \) is the number of possible hits for query \( q \), \( R(q) \) is the set of ranks of correct hits for \( q \), and \( p(q, r) \) is precision for query \( q \) at rank \( r \). AUC differs from NIST MAP in that normalization factor \( N(q) \) is the total number of possible correct hits for query \( q \) instead of the number of returned correct hits, so that AUC penalizes queries with no returned hits.

AUC measures the quality of ranking of search results, is not sensitive to the absolute value of hit confidences and does not require setting a confidence threshold. AUC, which takes on values in the range \([0, 1]\), is high when correct hits are ranked high in the list of results. Note that AUC is not weighted by the word frequency, i.e. both rare and common words contribute equally to the overall score.

**ATWV**, which stands for “actual term-weighted value”, incorporates the relative costs of misses and false alarms. It was introduced by NIST in the 2006 STD evaluation [17]. It requires the system to make a hard decision by setting a confidence threshold. Setting the threshold correctly for each query term (see [1] for details) requires an accurate estimate of the expected count for that term as well as accurate estimation of word confidences.

\[
ATWV = \frac{1}{|Q|} \sum_{q \in Q} \left( \frac{N_{\text{correct}}(q)}{N(q)} - \beta \frac{N_{\text{spurious}}(q)}{T - N(q)} \right)
\]

where the search term \( q \) occurs \( N(q) \) times in the reference transcript and the system returns \( N_{\text{correct}}(q) \) correct and \( N_{\text{spurious}}(q) \) incorrect search results for \( q \). \( T \) is the total duration of the speech in the corpus in seconds. The parameter \( \beta \) incorporates the relative costs of misses and false alarms and was set to 999.9 (same as in the STD06 evaluation).

**OTWV** (or optimal TWV) is the maximum ATWV that can be achieved with the given search results when the threshold is set optimally. Another way one can think of it is as the value of the search results when the system is not required to make a hard decision. This metric is more stable than ATWV and thus preferable for comparing performance across different systems, because it depends on the ranking of hits and not the absolute value of the confidence scores which can vary significantly across systems.

6. Results

Results of the experiments are presented in Table 3. One can see that word-based recognition leads to the lowest phone error rate (PER) followed by graphones and variable-length units. PER appears to correlate with the average unit length (see Table 1) which reflects the extent of the context that the language model is able to capture. LVCSR word recognition, followed by word-based search (row 1) achieves the highest AUC and ATWV on all types of IV queries, although for IV-rare1 and IV-OOT terms the advantage in AUC and ATWV over phonetic search is small and other systems are able to get higher OTWV. Comparing the three types of IV query, the KWS accuracy of the word-based system appears to
consistently improve when more training samples of the keywords are available, while the other systems are fairly invariant to the number of keyword samples in training.

Context-dependent phone models lead to the worst performance across the board. Among the subword models, variable-length units lead to the best accuracy on IV terms that are present in training, while the fixed-length units achieve the best scores on out-of-training and OOV terms. These observations are also supported by the term-weighted DET curves for OOV and IV-rare1 terms in Figures 2 and 3, respectively.

The approach to building a subword recognizer presented in this work is not bound to the maximum unit length of three phonemes. In the future we plan to evaluate units of varying lengths. We also expect the variety of subword systems to offer a great potential for system combination, which we plan to explore in future work.

8. References


