A Pointwise Approach to Pronunciation Estimation for a TTS Front-end

Shinsuke Mori, Graham Neubig

Kyoto University, School of Informatics

Abstract

In this paper, we propose a pointwise approach to the Japanese TTS front-end. In this approach, phoneme sequence estimation of sentences is decomposed into two tasks: word segmentation of the input sentence and phoneme estimation of each word. Then these two tasks are solved by pointwise classifiers without referring to the neighboring classification results.

In contrast to existing sequence-based methods, an $n$-gram model based on sequences of word-phoneme pairs for example, this framework enables us to use various language resources such as sentences in which only a few words are annotated, or an unsegregated list of compound words, among others.

In the experiments, we compared a joint tri-gram model with the combination of a pointwise word segmenter and a pointwise phoneme sequence estimator. The results showed that our framework successfully enables a TTS front-end to refer to a partially annotated corpus and/or a word sequence list annotated with phoneme sequences to realize a far larger improvement in accuracy.

1. Introduction

A text-to-speech (TTS) system consists of two modules. One is a front-end, which takes a sentence as its input and returns a phoneme sequence annotated with accent information of the sentence. The other is a back-end, which converts the output of a front-end into sound. For a front-end, the vital part is phoneme sequence estimation, as the intelligibility depends on its correctness. To estimate the correct phoneme sequence of a sentence, we need to recognize words and determine their phoneme sequences.

For English, there have been attempts at solving this problem by neural networks [1]. In English there are few words of multiple pronunciations, such as read, and they can be easily distinguished by their part-of-speech information, the research focus has been shifted to so-called G2P, phoneme sequence estimation from graphemes especially for unknown words [2] [3].

In some languages such as Finnish, Turkish, and Korean, character sequences, which are separated by whitespaces, are combinations of prefixes, a stem, and suffixes, so they are not be able to be covered with an ordinary vocabulary. To cope with this problem, there is data-driven research for these languages such as [4], which describes a data-driven method for Korean language pronunciation estimation.

In some other languages such as Japanese or Chinese, which we focus on in this paper, there is no whitespace between words. Thus phoneme sequence estimation of a sentence in these languages, we must segment the input sentence into words and annotate them with phoneme sequences. There have been some attempts at solving this problem with a joint $n$-gram models which use word/phoneme pairs as units [5]. This is a natural extension of a morphological analyzer based on word/part-of-speech pairs [6, 7].

This sequence-based modeling, however, requires a fully annotated corpus, a collection of sentences divided into word sequences annotated with phonemes, and is not capable of referring to a variety of language resources, such as a partially annotated corpus or a dictionary containing word sequences annotated with phonemes. In this paper, we propose a pointwise approach to phoneme sequence estimation for Japanese. In this approach, the phoneme sequence estimation task for sentences is decomposed into two tasks: one is word segmentation of the input sentence and the other is phoneme estimation of each word. Then these two tasks are solved by pointwise classifiers without referring to the neighboring classification results. This framework allows us to use various language resources such as sentences annotated only with word boundary information or word sequences from sentences partially annotated with word boundary information and a phoneme sequences [8, 9, 10].

In the experiments, we compared a joint tri-gram model with a combination of our pointwise word segmenter and our pointwise phoneme sequence estimator. The results showed that our framework successfully enables a TTS front-end to refer to a partially annotated corpus and/or a compound word list with a phoneme sequence to realize further improvement in accuracy.

2. A Data-driven Approach to TTS Front-end

As described in the previous section, a front-end of a TTS system for Japanese has to solve the word segmentation problem and phoneme sequence estimation for segmented words. In this section, we describe the joint $n$-gram approach [5, 2, 3], as a representative of the existing data-driven methods.

2.1. Front-end Based on a Joint $N$-gram Model

A statistical morphological analyzer [6] takes word/part-of-speech (POS) pairs as the unit of a joint $n$-gram model and searches for the word-POS pair sequence with the highest generative probability for a given sentence. Under the condition that the concatenation of the words in the sequence is equal to the input sentence. Inspired by this research, a TTS front-end based on the same framework has been proposed in [5]. This research proposes to use pairs of a word $w$ and its phoneme sequence $y$ as the unit of an $n$-gram model, $u = (w, y)^1$. Then the probability of a unit sequence $u = u_1 u_2 \cdots u_n$ by an $n$-gram model

---

1. In the original paper [5], a quadruplet of spelling of a word, its POS, its phoneme sequence, and its accent sequence is used.
Corpora must be fully annotated.

Figure 1: The sequence-based approach.

\[ M_{n,u}(u_1 u_2 \cdots u_h) = \prod_{i=1}^{h+1} P(u_i | u_{i-n+1} \cdots u_{i-2} u_{i-1}). \]  

(1)

where \( u_i \) (\( i \leq 0 \)) is a special symbol indicating the beginning of the sentence, and \( u_{h+1} \) is another special symbol indicating the end of the sentence. They are introduced just for a notation simplicity.

Similarly to the morphological analyzer, the statistical front-end, given a character sequence \( x = x_1 x_2 \cdots x_h \) as an input sentence, searches for the unit sequence \( u \) with the highest probability under the constraint that the concatenation of the spellings \( w = u_1 u_2 \cdots u_h \) is equal to the input sentence:

\[ \hat{u} = \underset{x \rightarrow w}{\operatorname{argmax}} M_{n,u}(u_1 u_2 \cdots u_h). \]  

(2)

The search problem is solved efficiently using dynamic programming [11].

2.2. Unknown Word Model

In real applications, unknown words are not avoidable. To estimate a pronunciation of an unknown word, the pair-based n-gram approach [5] uses an n-gram model based on pairs of character and phoneme sequence \( v = (x, y) \)

\[ M_{n,v}(u) = \prod_{i=1}^{h+1} P(v_i | v_{i-n+1} \cdots v_{i-2} v_{i-1}). \]  

(3)

where the spelling of \( u \) is equal to \( x_1 x_2 \cdots x_h \), which is the concatenation of characters of \( v_1 v_2 \cdots v_h \). Note that this unknown word module is very similar to the ones proposed for English unknown words [2] [3], since one Japanese character usually corresponds to one or two syllables.

2.3. Parameter Estimation

The parameters of n-gram models \( M_{n,u} \) and \( M_{n,v} \) are estimated from a corpus. In the corpus, sentences must be segmented completely into words and all the words must be annotated with a phoneme sequence. This annotation work is costly, especially considering domain adaptation. In an adaptation situation, we need annotators who know the segmentation standard well, for example, a sequence of a verbal stem and its ending followed by a sequence of auxiliary verb expressions and the pronunciations of special place names or technical terms in the medical domain. Putting it in another way, in an adaptation situation, we need to find specialists who are familiar with the target domain and the annotation standard at the same time. So we need a framework which allows us an easy and fast adaptation.

3. A Pointwise Approach

As we pointed out in the previous section, sequence-based methods require fully annotated training sentences, which are very costly to prepare. In order to overcome this shortcoming, we propose a pointwise approach, in which the phoneme sequence estimation task of a sentence is divided into two steps as shown in Figure 2:

1. Word Segmentation (WS): Dividing an unsegmented character string into appropriate units.
2. Pronunciation Estimation (PE): Estimation of the pronunciation of each segmented unit.

In the subsequent part, we explain WS and PE based on a pointwise classifier.

3.1. Word Segmentation for a Sentence

The word segmentation problem is defined as putting white-spaces at all points between two characters belonging to different words and putting nothing between characters belonging to the same word according to a predefined word segmentation standard. Given an unsegmented character string \( x = x_1 x_2 \cdots x_h \) as input

\[ x = \text{ 大分は今日は快晴です}, \]

the characters are segmented into words by estimating whether a boundary between \( x_i \) and \( x_{i+1} \) exists for each \( i, 1 \leq i < h \).

Using this information, we can acquire a segmented word string \( w \).

\[ w = \text{大分は今日は快晴です}. \]

The pointwise method assumes that every decision about a segmentation point is independent from the other decisions. For example, the decision whether a word boundary lies between characters \( x_i \) and \( x_{i+1} \) can depend on any number of features based on the surrounding characters, but not on whether a boundary lies between characters \( x_{i-1} \) and \( x_i \). Putting it in another way, the classifier uses features of input characters but not of the output tags.

We propose, as possible features of the classifier, all of the character and character type n-grams (\( n \leq 3 \)) contained by

\[ 2 \text{In this paper, we use the term pronunciation as the same meaning of phoneme sequence.} \]
In order for our word segmenter to exploit a dictionary containing word sequences, our word segmenter also checks out whether the character n-grams with/without a word boundary at the corresponding position appear in the dictionary or not.

### 3.2. Pronunciation Estimation for a Word

The above word segmentation process outputs a word sequence. Then, we annotate each word in the sequence with possible pronunciation information. This is used in the second part of the experiments.

As classifiers for word segmentation and phoneme sequence estimation, we use linear SVMs [13].

#### 4. Evaluation

As an evaluation of our framework, we measured the phoneme sequence estimation accuracies for Japanese sentences of the joint tri-gram model and the pointwise method mainly in a domain adaptation case. In this section we show the results and evaluate our new framework.

### 4.1. Experimental Conditions

As a general domain corpus, we used the Balanced Corpus of Contemporary Written Japanese (BCCWJ) [12] which consists of sentences extracted from various sources. We held out every 10th sentence for our test set. The sentences are segmented into words and annotated with pronunciation manually (see Table 1). The corpus in the target domain is composed of articles extracted from newspapers specialized in the economy. The test corpus in this domain, made by taking every 10th sentence, is annotated with word boundary information and each word is annotated with a phoneme sequence to measure accuracies. The learning corpus is, however, annotated partially with this information. This is used in the second part of the experiments.

As classifiers for word segmentation and phoneme sequence estimation, we use linear SVMs [13].

#### 4.2. Evaluation Criterion

As an evaluation criterion we follow [5] and use precision and recall based on mora. First the longest common subsequence (LCS) is found between the correct answer and system output. As classifiers for word segmentation and phoneme sequence estimation, we use linear SVMs [13].

#### 4.3. Learning from a Fully Annotated Corpus

First we compared the accuracies in the general domain and the target domain of two methods. Table 2 shows the accuracies in the general domain and Table 3 shows those in the target domain. From a comparison between the recalls in Table 2, we see that about 16% errors in the pair tri-gram model were eliminated by our method. The difference is statistically significant with a level of 1%. Table 3 shows that the improvement is far larger in the target domain. From these results, we can say that our pointwise method outperforms the joint tri-gram method, and is particularly robust to out-of-domain text.

---

### Table 1: Corpora.

<table>
<thead>
<tr>
<th>usage</th>
<th>domain</th>
<th>#sentences</th>
<th>#words</th>
<th>#chars</th>
</tr>
</thead>
<tbody>
<tr>
<td>learning</td>
<td>balanced</td>
<td>33,147</td>
<td>899,025</td>
<td>1,292,249</td>
</tr>
<tr>
<td>learning</td>
<td>newspaper</td>
<td>9,023</td>
<td>–</td>
<td>398,570</td>
</tr>
<tr>
<td>test</td>
<td>balanced</td>
<td>3,681</td>
<td>98,634</td>
<td>141,655</td>
</tr>
<tr>
<td>test</td>
<td>newspaper</td>
<td>1,002</td>
<td>29,038</td>
<td>43,695</td>
</tr>
</tbody>
</table>

### Table 2: Accuracy in the general domain.

<table>
<thead>
<tr>
<th>Model</th>
<th>precision</th>
<th>recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pair tri-gram</td>
<td>99.07%</td>
<td>99.025</td>
</tr>
<tr>
<td>Pointwise</td>
<td>99.19%</td>
<td>99.26%</td>
</tr>
</tbody>
</table>

---

**Figure 3:** The characters to be referred to in word segmentation.

**Figure 4:** The characters to be referred to in pronunciation estimation of a word.
Table 3: Accuracy in the target domain before adaptation.

<table>
<thead>
<tr>
<th>Model</th>
<th>precision</th>
<th>recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pair tri-gram</td>
<td>97.83%</td>
<td>97.23%</td>
</tr>
<tr>
<td>Pointwise</td>
<td>98.04%</td>
<td>97.48%</td>
</tr>
</tbody>
</table>

Table 4: Accuracy in the target domain. PAC stands for a partially annotated corpus and DWS stands for a dictionary of word sequences.

<table>
<thead>
<tr>
<th>Model</th>
<th>PAC</th>
<th>DWS</th>
<th>precision</th>
<th>recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pair tri-gram</td>
<td>No</td>
<td>No</td>
<td>97.83%</td>
<td>97.23%</td>
</tr>
<tr>
<td>Pair tri-gram</td>
<td>Yes</td>
<td>Yes</td>
<td>98.02%</td>
<td>97.51%</td>
</tr>
<tr>
<td>Pointwise</td>
<td>No</td>
<td>No</td>
<td>98.04%</td>
<td>97.48%</td>
</tr>
<tr>
<td>Pointwise</td>
<td>Yes</td>
<td>No</td>
<td>98.27%</td>
<td>98.09%</td>
</tr>
<tr>
<td>Pointwise</td>
<td>No</td>
<td>Yes</td>
<td>98.07%</td>
<td>97.51%</td>
</tr>
<tr>
<td>Pointwise</td>
<td>Yes</td>
<td>Yes</td>
<td>98.29%</td>
<td>98.12%</td>
</tr>
</tbody>
</table>

4.4. Using Various Language Resources

One of the advantages of our framework is the ability to use various language resources, such as a partially annotated corpus, a dictionary containing words or word sequences with/without a phoneme sequence. In order to attest to this advantage experimentally, we built pointwise pronunciation estimators using the following language resources in the target domain:

PAC: a partially annotated corpus.

We annotated 1,366 words in the learning corpus in the target domain with word boundary information and phoneme sequences. They are selected by the classifier estimated from the corpus in the general domain according to active learning based on classifier uncertainty [8].

DWS: a dictionary of word sequences with phoneme sequences.

We selected 1,060 most frequent compound words in a dictionary appearing in the learning corpus in the target domain and annotated them with word boundary information and phoneme sequences. As a result the dictionary contained 1,928 words (the average length is 1.82 words).

In addition we built a joint tri-gram, referring to the additional language resources (PAC and DWS) as a part of the learning corpus. Annotated sequences are added directly to the learning corpus.

Table 4 shows the accuracies in the target domain of the pronunciation estimators built from various combinations of the language resources. By referring to the additional language resources, the joint tri-gram increased the accuracy, but it is comparable to the pointwise method without any additional language resource. By referring to the PAC or DWS, the pointwise method improves the accuracy. PAC yields, however, a much larger improvement than DWS. The reason is that PAC uses the active learning criterion to select the words to be annotated, but DWS takes only the frequencies of the compound word candidates into account. In the pointwise framework, when we use two language resources together, we obtain a further improvement and the accuracy was highest.

From the above observations, we can conclude that our pointwise framework is better than the existing pair n-gram based method and is capable of utilizing a partially annotated corpus or a word sequence list to yield a higher accuracy.

5. Conclusion

In this paper we proposed a pointwise approach to phoneme sequence estimation for a TTS front-end in Japanese. Instead of modeling a sentence as a word/phoneme-sequence pair and solving the problem simultaneously, we decomposed the task into two steps (word segmentation and pronunciation estimation) to enable us to utilize various language resources, such as partially annotated sentences or a compound word list with and without phoneme sequences. Experiments comparing an existing pair-based n-gram method and our pointwise method using the same language resources showed that our pointwise approach outperforms an existing method even when the same data are used. Then we showed that the pointwise model is capable of referring to a partially annotated corpus and/or a compound word list with a phoneme sequence to realize further improvement in accuracy. These results show that our pointwise approach is better than the existing one.

6. References


