Evaluating Spoken Language Model Based on Filler Prediction Model in Speech Recognition

Kengo Ohta\textsuperscript{1}, Masatoshi Tsuchiya\textsuperscript{2}, Seiichi Nakagawa\textsuperscript{1}

\textsuperscript{1}Department of Information and Computer Sciences / \textsuperscript{2}Information and Media Center, Toyohashi University of Technology, Japan
kohta@slp.ics.tut.ac.jp, tsuchiya@imc.tut.ac.jp, nakagawa@slp.ics.tut.ac.jp

Abstract

We propose a method that uses a filler prediction model for building a language model that includes fillers from a corpus without fillers. In our method, a filler prediction model is trained from a corpus that does not cover domain-relevant topics. It recovers fillers in inexact transcribed corpora in the target domain, and then a language model that includes fillers is built from the corpora. The results of an evaluation of the Japanese National Diet Record showed that a model using our method achieves higher recognition performance than conventional ones.

Index Terms: Spoken Language, Filler, Language Model, Speech Recognition, Japanese National Diet Record

1. Introduction

The increasing number of digital audio archives of lectures, presentations, and meetings that span various domains requires an effective indexing system. For this, automatic speech recognition (ASR) of spontaneous speech is essential because transcriptions of recorded data are necessary for indexing. Standard modern ASR systems are based on statistical language models with large vocabularies. This necessitates the construction of a statistical language model that covers spoken-style expressions as well as domain-relevant topics. The simplest approach to constructing such a model is to train it from a large-scale corpus consisting of many faithful transcripts of spontaneous speech in the relevant domain. However, the available corpora are usually limited because they are quite expensive to prepare.

Several approaches have been proposed to address this problem. A typical one is combining a spoken language model that covers general spoken-style expressions with a written language model that covers domain-relevant topics. For example, a conversational telephone speech corpus is combined with meeting or Web corpora for speech recognition of meetings\cite{1}. Several similar methods that directly manipulate \(N\)-gram probabilities such as cache models\cite{2} have also been proposed. Class-based language models\cite{3} are also used for robust estimation of \(N\)-gram probabilities with limited or unmatched data. Stolcke and Shriberg\cite{4} proposed a method removing fillers from the language model history, but, their method achieved no significant impact on recognition accuracy. Akita and Kawahara\cite{5} proposed another approach that transforms an \(N\)-gram model trained from a written-style corpus of the target domain into an \(N\)-gram model covering spoken-style expressions by using a probabilistic transformation model trained from a parallel aligned corpus of the faithful transcripts and their written-style texts. However, it is quite difficult to obtain such an aligned corpus.

In this paper, we propose a new approach to this problem: using inexact transcribed corpora. Inexact transcribed corpora are widely produced in the form of shorthand notes or meeting records and are more freely available than faithful ones because their inexactness reduces transcription costs. For example, the Japanese National Diet Library\textsuperscript{1} publishes the Japanese National Diet Record (NDR), which contains numerous inexact transcriptions of debates in the Japanese National Diet for the past 60 years. Furthermore, such corpora can be more appropriate than the written-style corpora as a resource for training a spoken language model that covers domain-relevant topics because they include more spoken-style expressions in the target domain than the written-style corpora. Of course, unlike faithful transcribed corpora, inexact transcribed corpora lack almost all disfluencies, and therefore it is necessary to recover the disfluencies before using the corpora to train a spoken language model that covers disfluencies.

We focus especially on fillers because they occur much more frequently than other disfluency acts, such as word fragments and incorrect or reduced fragments\textsuperscript{6, 7}. This paper describes a method using a filler prediction model trained from a corpus that includes fillers but does not cover domain-relevant topics to recover fillers in inexact transcribed corpora in a target domain.

The remainder of this paper is organized as follows. Section 2 formalizes the filler prediction model. The procedure for using it to build a language model is described in Section 3. Section 4 describes an experiment using the NDR that shows that our proposed method achieves better performance than alternative methods. The paper concludes with final remarks in Section 5.

2. Filler Prediction Model

2.1. Model Description

There are two possible approaches to constructing a statistical language model including fillers from an inexact transcribed corpus with no fillers.

The first approach comprises two steps: constructing a model that does not cover fillers from the inexact transcribed corpus and then transforming it into a model with fillers using a transformation model. The second approach is to transform a corpus excluding fillers into one that includes fillers before constructing the model. Consider the following two sentences.

(1) This display shows that \(\cdots\)
(2) This uh display shows that \(\cdots\)

String (1) is an example sentence in the inexact transcribed corpus, whereas String (2) is the same sentence after the filler denoted by an underline has been recovered. Once we have obtained a corpus in which fillers have been recovered, as in String (2), constructing a model from it is quite easy.

While the transformation model used in the first approach depends heavily on the structure of the target language model, the language modeling and corpus transformation steps in the second approach are clearly distinct. This distinction means that the second approach can more easily incorporate advances

\textsuperscript{1}http://kokkai.ndl.go.jp/
in language modeling research. Because of this, we investigate the second approach in this paper.

The second approach requires a filler prediction model that predicts both the places where fillers should be inserted and the filler types for those places. An analysis by Maekawa [7] showed that many different fillers appear in real spontaneous speech, which may lead to data sparseness. To avoid this, we introduce the assumption that the prediction of filler places and the selection of filler types are independent. This means that our proposed filler prediction model consists of two sub-models: a filler insertion model that predicts places where fillers should be inserted and a filler selection model that selects appropriate fillers for the given places.

2.2. Filler Insertion Model

Given a certain word sequence, the filler insertion model predicts places where fillers would normally be inserted. We formalized this model as a sequence labeling problem as shown in Fig. 1, where BOS denotes “Beginning Of Sentence”.

The label F means that a filler should be inserted immediately after the labeled word, whereas the label O denotes the contrary.

<table>
<thead>
<tr>
<th>Word sequence</th>
<th>This display shows that...</th>
</tr>
</thead>
<tbody>
<tr>
<td>(BOS) pron noun verb pron</td>
<td>...</td>
</tr>
<tr>
<td>Label sequence</td>
<td>O F O O O ...</td>
</tr>
</tbody>
</table>

Figure 1: Example of Filler Insertion Labeling

We use a conditional random field (CRF) [8] model for this labeling problem. A CRF is a discriminative probabilistic model that offers several advantages over hidden Markov models and is used in several statistical natural language processing tasks, including language modeling [9].

When a certain word sequence X is given, the conditional probability of a label sequence Y is defined as follows:

\[
P(Y|X) = \frac{1}{Z(X)} \exp \left( \sum_{i=1}^{n} \lambda_a f_a(X_i, Y_i) \right),
\]

where \( n \) is the length of \( X \), \( f_a \) is a feature function, \( \lambda_a \) is the weight of the feature function, and \( Z(X) \) is the normalization factor.

2.3. Filler Selection Model

The filler selection model selects appropriate fillers for the given places. We simply employ the conditional distribution of fillers over contexts for this model.

Witten-Bell discounting [10] is used to estimate the conditional probability of the filler selection model. Suppose the index \( k \) is given, the conditional probability \( P_x(f|h) \) of the filler \( f \) given the context \( h \) as follows:

\[
P_x(f|h) = \begin{cases} \frac{c(h,f)}{c(h)+r(h)} & \text{if } c(h,f) > 0 \\ c(h)+r(h) & \text{otherwise} \end{cases},
\]

where \( c(h,f) \) is the frequency of filler \( f \) occurring in the context \( h \), \( c(h) \) is the frequency of \( h \), \( r(h) \) is the number of different fillers that appear immediately after \( h \), and \( h' \) is the reduced context of \( h \) (back-off).

3. Procedure to Build Language Model Using Filler Prediction Model

This section describes the procedure to build a language model using the filler prediction model. The proposed method basically consists of the following three steps.

1. First, train a filler prediction model from a faithful transcribed corpus (training corpus) that includes fillers but does not cover domain-relevant topics. This step consists of two sub-steps:
   (a) Training of a filler insertion model and
   (b) Training of a filler selection model.

2. Second, transform an inexact transcribed corpus (development corpus), which covers domain-relevant topics but includes no fillers, into a corpus that includes fillers as well as domain-relevant topics using the filler prediction model.

3. Third, build a statistical language model from the filler inserted development corpus resulting from the above step. In our experiment, a standard 3-gram model was used.

Our assumption of the filler prediction model described in Section 2 divides the training of the model into two steps: training of the filler insertion model and training of the filler selection model. Each training instance of the filler insertion model is composed of a word sequence excluding fillers and of the corresponding label sequence as shown in Fig. 1. Such instances are easily obtained from the faithful training corpus. The CRF that represents the filler insertion model is trained with the toolkit CRF++, which uses LBFGS, a quasi-Newton algorithm for large scale numerical optimization problems, to estimate parameters and a Gaussian prior to avoid over-fitting. We used a combination of the preceding two words, current words, succeeding two words, their POSs, and the preceding two moras as features of the CRF.

The filler selection model is obtained from \( N \)-gram statistics of the faithful training corpus as shown in Equation (2). However, there are many pronunciation variants of fillers. For example, there are 151 kinds of filler in the CSJ [7], many of which are due to the appearance or nonappearance of choked sounds, long vowels, or repetitions of syllables in the last portion. To obtain a robust estimation of \( P_x(f|h) \), we combined fillers with similar pronunciation while training the filler selection model. As a result, we reduced the kinds of fillers from 151 to 58 [11].

The next step is to transform a development corpus, which covers domain-relevant topics but includes no fillers, into a corpus that includes fillers as well as domain-relevant topics by using the filler prediction model. This procedure also comprises two steps: predicting places for fillers with the filler insertion model and choosing fillers for the selected places with the filler selection model. Suppose the index \( i \) ranges over all words in the development corpus. The probability that a filler should be inserted immediately after the \( i \)-th word is defined by

\[
P(y_i = f|X) = \sum_{Y: y_i = f} P(Y|X).
\]

When a uniform random variable \( Q_i \) (0 ≤ \( Q_i \) < 1) satisfies the condition \( Q_i < P(y_i = f|X) \), a filler is inserted immediately after the \( i \)-th word. Once a filler insertion place has been selected, the filler selection model is used to select an appropriate filler. When a uniform random variable \( Q_i' \) (0 ≤ \( Q_i' \) < 1) satisfies the equation

\[
\sum_{j=1}^{k} P_x(f_j|h_i) \leq Q_i' < \sum_{j=1}^{k} P_x(f_j|h_i),
\]

http://chasen.org/~taku/software/CRF++/
where $j$ is the index of all possible fillers and $h_i$ is the context around the $i$-th word, the filler $f_i$ is selected as an appropriate filler for the given position. Two random variables are introduced into the above procedure to simulate the process whereby humans include fillers. This process is not deterministic, but is obviously stochastic. As a result of these random variables, various corpora differing with respect to the inserted fillers are generated by the proposed method despite having been given the same training and development corpora. The average of 10 trials is therefore reported as the experimental results of the proposed method in Section 4.

Once a filler inserted development corpus including fillers as well as domain-relevant topics is generated by the above procedure, constructing a 3-gram model from it is quite easy. The vocabulary size of these models is 20 K.

4. Experimental Evaluation

In this section, we explain our experiments using the NDR, which contains real shorthand notes. The CSJ is employed as the training corpus of the filler prediction model and the NDR is employed as the development and test corpora.

4.1. Domains of NDR and CSJ

The NDR, published by the Japanese National Diet Library, consists of transcriptions of various debates in the Japanese National Diet for the past 60 years. These transcriptions are inexact with respect to most disfluencies, including fillers. The CSJ is a large-scale database of spontaneous Japanese containing four categories of speech: academic presentation speech (APS), simulated public speech (SPS), conversation, and reading. The NDR and CSJ share quite small domains, as shown in Table 1. The following experiments show that our method works well, despite this domain difference.

<table>
<thead>
<tr>
<th>Training corpus</th>
<th>Test corpus</th>
<th>OOV ratio</th>
<th>Fillers</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPS (1665 lectures)</td>
<td>SPS (50 lectures)</td>
<td>0.86%</td>
<td>5.28%</td>
</tr>
<tr>
<td>NDR (2 hours)</td>
<td>11.09%</td>
<td>8.00%</td>
<td></td>
</tr>
</tbody>
</table>

4.2. Evaluation of Filler Prediction

As previously described, NDR transcriptions are inexact with respect to most disfluencies, including fillers. An example (translated into English) follows:

- Under these circumstances, I think these budget debates should proceed on the basis of what you have claimed and what you are planning ...

The proposed method inserts fillers into such corpus as follows: (inserted fillers are underlined)

- Under these circumstances, well I think these budget debates should ah proceed on the basis of what you have well claimed and what you are planning ...

A manual transcription of the fillers is as follows: (fillers are double-underlined)

- Under these circumstances, well I think these ah budget debates should well proceed on the basis of what you have claimed and well what you are planning ...

As shown above, our proposed filler prediction model can properly insert fillers.

The prediction of such filler insertion is shown in Table 3. When we evaluated only filler insertion, our filler insertion model based on CRF (No.1) achieved a 23% F-measure, which is much better than that one based on unigram and trigram. On the other hand, when we evaluated both filler selection and insertion, our proposed model (No.4) achieved 6%.

This precision seems to be considerably poor, but it should be noted that a filler is a probabilistic phenomenon. (cf. The word hit rate of a 3-gram language model is approximately 17%[12].)

4.3. Evaluation of ASR

4.3.1. Experimental Setup

First, we prepared the test data from the NDR. In particular, we selected four debates in which fillers occurred as frequently as in the CSJ. A 5-minute long section was extracted from each debate, resulting in 20 minutes of data. All fillers and other disfluencies which occur adjacent to fillers were recovered manually by referring to the NDR video archives.

In all the experiments, we used the Julius decoder (ver 3.5.3) and a CSJ-APS,SPS acoustic model (a gender-independent triphone model), which was attached to the CSJ.

For a baseline model, we evaluated three language models. CSJ-Attached is the language model attached to the CSJ. The NDR model is trained from the NDR. The NDR+SPS model is trained from the combined corpora of the NDR and SPS (a similar method to [1]). We compared the language model based on the proposed method (Proposed) with these baseline models. In addition, we evaluated a model trained from a mixed corpora of filler-inserted NDR and SPS.

The details of each corpus are shown in Table 2.

Both the test-set perplexity and the adjusted test-set perplexity were used to evaluate each language model. When a test corpus $w^n_t$ is given, the cross entropy $H$ of the language model $L$ has the form

$$H(L) = \frac{1}{n} \sum \log P_L(w^n_t),$$

(4)

where $P_L(w^n_t)$ is the probability that the test corpus $w^n_t$ occurs over the language model $L$, and the test-set perplexity $PP$ is defined as

$$PP = 2^{H(L)}.$$  (5)

The adjusted test-set perplexity described by Ueberla [13] is the improved evaluation metric to consider OOVs in the test corpus and is defined as

$$\log_2 PP^* = \log_2 PP + \frac{o}{n} \log_2 m,$$  (6)

where $o$ is the number of OOVs in the test corpus and $m$ is the number of kinds of OOVs in the test corpus.

Unfortunately, we could not compare the CSJ-Attached model with the other models because it is based on the different part-of-speech system than other models.

Additionally, the word correct and the word accuracy were used as metrics to evaluate each recognition result overall and the recognition result around fillers (filler, preceding two words, and succeeding two words).

<table>
<thead>
<tr>
<th>Domain</th>
<th>Training corpus</th>
<th>Development corpus</th>
<th>Test corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPS</td>
<td>329.9</td>
<td>N/A</td>
<td>0.3</td>
</tr>
<tr>
<td>NDR</td>
<td>55 M</td>
<td>0</td>
<td>0.3 k</td>
</tr>
<tr>
<td>NDR+SPS</td>
<td>36 M</td>
<td>0</td>
<td>3.6 k</td>
</tr>
</tbody>
</table>
4.3.2. Experimental Results

The evaluation results of each language model are shown in Table 4. The columns of “NOT distinguished” show the results when all fillers are not distinguished and are unified into one symbol, and the ones of “distinguished” show the results when fillers are distinguished. As can be seen, the NDR trigram model that was simply trained from the NDR achieved a 87.2 PP and 133.1 PP*, respectively. When we used the NDR+SPS model, the PP and OOV rate substantially improved, while PP* degraded. On the other hand, the Proposed model achieved substantially better performance than the NDR+SPS model on all the metrics. The mixed corpus, used to train the NDR+SPS model, includes n-grams of fillers and out-of-domain words and ones of in-domain words, however, includes few n-grams including fillers and in-domain words. It means that obtaining the probabilities of n-grams that include both filler and in-domain words is difficult, so the performance of this model could be limited. In comparison, the Proposed model includes such n-grams and does not include out-of-domain words.

We can see that the word correct and the word accuracy have the same tendencies in all the models. When all fillers are not distinguished, the NDR+SPS model achieved the best performance among the baseline models. However, the Proposed model achieved substantially better performance. In particular, the improvement in the words around fillers is significant. We can see the same tendency when fillers are distinguished.

<table>
<thead>
<tr>
<th></th>
<th>Filler insertion model</th>
<th>Filler selection model</th>
<th>Precision</th>
<th>Recall</th>
<th>F-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CRF (Context: 2 words around)</td>
<td>(All fillers unified into one symbol)</td>
<td>0.26</td>
<td>0.21</td>
<td>0.23</td>
</tr>
<tr>
<td>2</td>
<td>Word Trigram</td>
<td></td>
<td>0.17</td>
<td>0.12</td>
<td>0.14</td>
</tr>
<tr>
<td>3</td>
<td>Unigram</td>
<td></td>
<td>0.06</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>4</td>
<td>CRF (Context: 2 words around)</td>
<td>Word Trigram</td>
<td>0.08</td>
<td>0.05</td>
<td>0.06</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>Unigram</td>
<td>0.06</td>
<td>0.04</td>
<td>0.03</td>
</tr>
<tr>
<td>6</td>
<td>Word Trigram</td>
<td>Word Trigram</td>
<td>0.05</td>
<td>0.03</td>
<td>0.04</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>Unigram</td>
<td>0.04</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>8</td>
<td>Unigram</td>
<td>Unigram</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
</tr>
</tbody>
</table>

### Table 3: Evaluation Results of Filler Prediction

### Table 4: Evaluation Results of ASR

<table>
<thead>
<tr>
<th>LM</th>
<th>Vocabulary size</th>
<th>PP</th>
<th>PP*</th>
<th>OOV (%)</th>
<th>NOT distinguished</th>
<th>distinguished</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Overall (%)</td>
<td>Around fillers (%)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Overall (%)</td>
<td>Around fillers (%)</td>
</tr>
<tr>
<td>CSJ</td>
<td>2-gram</td>
<td>25 000</td>
<td>—</td>
<td>—</td>
<td>49.0</td>
<td>40.4</td>
</tr>
<tr>
<td>NDR</td>
<td>2-gram</td>
<td>20 000</td>
<td>101.0</td>
<td>154.2</td>
<td>9.88</td>
<td>53.8</td>
</tr>
<tr>
<td>NDR+SPS</td>
<td>2-gram</td>
<td>20 000</td>
<td>110.2</td>
<td>130.7</td>
<td>4.03</td>
<td>56.1</td>
</tr>
<tr>
<td>Proposed</td>
<td>2-gram</td>
<td>20 000</td>
<td>99.2</td>
<td>116.5</td>
<td>3.86</td>
<td>61.3</td>
</tr>
<tr>
<td>Proposed+SPS</td>
<td>2-gram</td>
<td>20 000</td>
<td>93.2</td>
<td>110.6</td>
<td>4.03</td>
<td>60.3</td>
</tr>
</tbody>
</table>

5. Conclusion

In this paper, we proposed a novel method using a filler prediction model for constructing a spoken language model including fillers from an inexact transcribed corpus excluding fillers. The filler prediction model consists of two sub-models: the filler insertion model and the filler selection model. The filler insertion model predicts places where fillers should be inserted, and is represented by a CRF. The filler selection model selects appropriate fillers for given places and is represented by a simple conditional distribution. The experiment using the National Diet Record showed that the model constructed using the proposed method achieved substantially better performance than conventional models in speech recognition. In future work, we plan to improve the proposed method so that the process to insert fillers into the development corpus is skipped.

### References


