IEICE TRANS. INF. & SYST., VOL.E89–D, NO.3 MARCH 2006

PAPER Special Section on Statistical Modeling for Speech Processing

Trigger-Based Language Model Adaptation for Automatic Transcription of Panel Discussions

Carlos TRONCOSO[†a)], ***Student Member and*** **Tatsuya KAWAHARA**[†], ***Member***

SUMMARY We present a novel trigger-based language model adaptation method oriented to the transcription of meetings. In meetings, the topic is focused and consistent throughout the whole session, therefore keywords can be correlated over long distances. The trigger-based language model is designed to capture such long-distance dependencies, but it is typically constructed from a large corpus, which is usually too general to derive task-dependent trigger pairs. In the proposed method, we make use of the initial speech recognition results to extract task-dependent trigger pairs and to estimate their statistics. Moreover, we introduce a back-off scheme that also exploits the statistics estimated from a large corpus. The proposed model reduced the test-set perplexity considerably more than the typical trigger-based language model constructed from a large corpus, and achieved a remarkable perplexity reduction of 44% over the baseline when combined with an adapted trigram language model. In addition, a reduction in word error rate was obtained when using the proposed language model to rescore word graphs.

key words: *speech recognition, language model, trigger-based language model, TF/IDF*

1. Introduction

In automatic speech recognition (ASR), the most widely used language model is the n-gram model, where n typically ranges from 2 (bigram) to 4 (4-gram). The n-gram language model estimates the occurrence probability of n consecutive words in the text. This model is known to be effective, but it is apparently limited in scope, because it is unable to model dependencies longer than n.

Alternative approaches have been proposed to try to broaden the scope of the n-gram by modeling long-distance dependencies between words. These include the trigger-based language model [1]–[4], the cache-based language model [5], [6], latent semantic analysis-based language models [7], and structured language models [8]. This work focuses on the trigger-based language model.

The trigger-based language model uses a set of correlated word pairs, known as trigger pairs, to raise the probability of the words "triggered" by others in the word history. The conventional trigger-based language model, however, has some limitations. This model has been mostly applied to corpora of newspaper articles. This kind of corpora are usually too general in topic and do not closely match the specific test data. Moreover, it has been observed that much of the potential of trigger-based language models lies in words that trigger themselves, called *self-triggers*. Self-triggers are virtually equivalent to the cache-based language model, so the original trigger-based language model does not significantly outperform the cache-based model.

This paper addresses an effective implementation of the trigger-based language model mainly targeting at a meeting transcription task to overcome the model's limitations. The transcription of meetings and lectures is one of the promising applications of large vocabulary continuous speech recognition. The subject matter in a meeting is fairly homogeneous during it, so we can expect to find keywords related in their topic throughout the whole session. The trigger-based language model could be used to capture these constraints, but typical large corpora such as newspapers are too general to extract task-specific trigger pairs and their statistics. On the other hand, the data from a single meeting session can be used to extract trigger pairs, and we expect that the probabilities of the trigger pairs can also be estimated from these data.

In the proposed approach, we regard a meeting session as a document unit, and the trigger pairs are extracted from its initial speech recognition results. The initial transcription, though containing errors, can provide useful information about the topic and speaking style of the meeting. We introduce several techniques that filter this useful information from the initial transcription and also exploit a large corpus based on a back-off scheme. The resultant model is used for rescoring the initial speech recognition results.

The rest of this paper is organized as follows. Section 2 describes the task addressed in this work, as well as the proposed approach. Section 3 deals with the extraction of trigger pairs from the initial transcription. Then, their probability estimation and an enhancement based on a back-off scheme using a large corpus are explained in Sect. 4. The perplexity evaluation of these models in a panel discussion transcription task is presented in Sect. 5, as well as a further enhancement by combining the model with n-gram model adaptation. Speech recognition evaluation is portrayed in Sect. 6.

2. Trigger-Based Language Model Adaptation

2.1 Description of Task and Corpora

The target task in this work is the transcription of panel discussions from a Japanese TV program called "Sunday Discussion" broadcasted by NHK [9]. This program consists

Manuscript received July 11, 2005.
Manuscript revised October 5, 2005.
[†]The authors are with the School of Informatics, Kyoto University, Kyoto-shi, 606–8501 Japan.
a) E-mail: carlos@ar.media.kyoto-u.ac.jp
DOI: 10.1093/ietisy/e89–d.3.1024

of discussions on current political and economic issues by politicians, economists and other experts in the field. A specific agenda is given for each session of the discussions. A chairperson also takes part and prompts the speakers. The duration of each session is one hour. Ten programs recorded from June 2001 to January 2002 were used in this work. These programs were chosen arbitrarily to cover diverse topics and a sufficient variety of speakers. The average number of utterances and words per program is 550 and 14 K, respectively. The total number of words in the test set is 134,405.

We also make use of a large corpus of the minutes of the National Diet (Congress) of Japan [9] from 1999 to 2002. We selected this corpus because of its similarity in topic with the panel discussion programs, since both corpora mainly deal with politics and economics. The total number of words in the corpus is 71 M. Documents in this corpus are divided by the kind and date of meetings, and the total number of documents is 2866. Among them, we select 671 documents from the year 2001 as a portion similar to the test set.

2.2 Proposed Approach

Since each session of the discussions focuses on a particular topic, we expect to find topic-related words during the whole program. In order to capture these long-distance dependencies, we propose to use the trigger-based language model. This model, however, is usually trained from large corpora such as newspapers. These corpora are too general in topic, so the resulting trigger pairs are not task-dependent.

We propose an adaptation paradigm in which the trigger pairs are extracted, and their probabilities are estimated from the initial speech recognition results. The initial transcription, although erroneous, contains many of the keywords whose dependencies we want to model. Therefore, it is a good source for deriving task-dependent trigger pairs, which we expect to have a significant effect on perplexity and speech recognition accuracy in a rescoring framework. To the best of our knowledge, this is the first work on constructing a trigger-based language model from the initial transcription.

This approach, however, poses two problems. The first one is that the size of the training data, that is, the size of the initial transcription, is much smaller than that of a large corpus, so it might be insufficient to extract enough trigger pairs and to reliably estimate their probabilities. The second problem is that, since the initial transcription contains errors, we may obtain erroneous triggers in addition to correct trigger pairs. These erroneous trigger pairs can have a harmful effect, increasing the probabilities of wrong words.

In order to cope with the first problem, instead of extracting the trigger pairs from a window of fixed length with the average mutual information, we use the term frequency/inverse document frequency measure to find keywords from the whole document, and then we let any combination of two keywords be a candidate trigger pair. In this way, not only do we boost the possible number of trigger pairs, but we also capture topic constraints global to the document. In addition, since the probability estimates derived from the initial transcription might not be reliable, we propose a back-off scheme that incorporates statistics from a large corpus to the model.

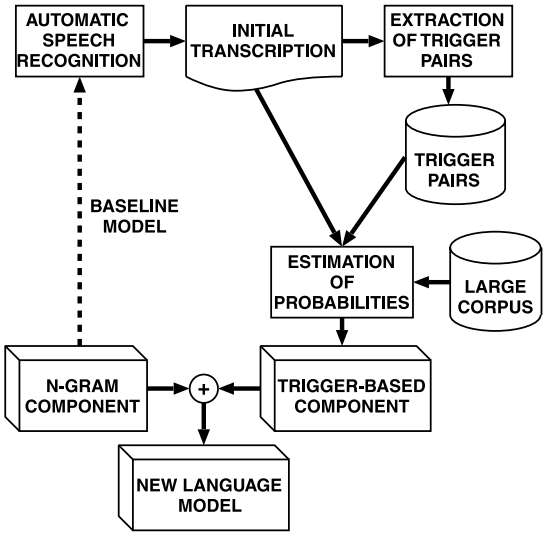

Fig. 1 Outline of the proposed approach.

As for the second problem, we use a confidence measure score to get rid of those trigger pairs whose component words are not reliable, while we assume that correct trigger pairs have a greater confidence score and consistently appear throughout the session. In this way we expect to minimize the number of incorrect trigger pairs.

Figure 1 illustrates the outline of the proposed approach. First, ASR is performed with a standard n-gram as the baseline language model, yielding the initial speech recognition results. The trigger pairs are then extracted and their probabilities are estimated from the initial transcription, as well as from a large corpus. Finally, the resulting trigger-based component is combined with the n-gram component to produce a new language model for the second pass of speech recognition.

3. Extraction of Trigger Pairs from Initial Transcription

A trigger pair is a pair of content words that are semantically related to each other. Trigger pairs can be represented as $A \rightarrow B$, which means that the occurrence of word A "triggers" the appearance of word B, that is, if A appears in a text, it is likely that B will come up afterwards.

This section details the extraction of trigger pairs from the initial speech recognition results.

3.1 Extraction Based on TF/IDF Instead of Mutual Information

Task-dependent trigger pairs are extracted from the initial

transcription, namely the K-best ASR hypotheses. For the selection of pairs, instead of the average mutual information (AMI) used in [1], [2], we use the term frequency/inverse document frequency (TF/IDF) measure [10]. We employ this measure because it is document-based, that is, it lets us extract the trigger pairs from a whole document, rather than from a text window of the corpus. In this way, we can capture global constraints from each document. This measure is also chosen because of its simplicity.

The TF/IDF value of a term t_k in a document D_i is computed as follows:

$$V_{ik} = \frac{tf_{ik} \log(N/df_k)}{\sqrt{\sum_{j=1}^{T} (tf_{ij})^2 [\log(N/df_j)]^2}}, \quad (1)$$

where tf_{ik} is the frequency of occurrence of t_k in D_i, N is the total number of documents, df_k is the number of documents that contain t_k, and T is the number of terms in D_i.

Since the initial transcription intuitively consists of only one document, the TF part (tf_{ik} and T) is calculated from the K-best hypotheses, whereas the IDF part (N and df_k) is computed from a fraction of a large corpus similar to the target task.

3.2 POS and Stop Word Filtering

We create all possible word pairs, including pairs of the same words (self-triggers), with the base forms and parts of speech (POS) of all content words with a TF/IDF value above a threshold. By regarding any combination of content words as a trigger pair, even though the size of the initial transcription is small, we obtain a large list of candidate trigger pairs. By using base forms we avoid same-root triggers, and we can apply the trigger pair when a word is presented with any inflection, while by using the POS information we distinguish between homonyms with different POS when applying the trigger pairs.

POS-based filtering is introduced to discard function words, and a stop word list with the most frequent words is used to ignore them during the extraction.

Table 1 shows some examples of trigger pairs extracted from the initial transcription of the target task.

3.3 Filtering with Confidence Score and Large Corpus

In order to minimize the adverse effect of erroneous trigger pairs, we introduce two methods to get rid of as many incorrect triggers as possible. First, we use the confidence score that is provided by the ASR system to eliminate the trigger pairs whose component words have a confidence score lower than a threshold.

Then, we compare the trigger pairs extracted from the initial transcription with pairs extracted from a large corpus, and we discard the trigger pairs which are not present in the second set.

With these methods, we can extract reliable trigger pairs, which are matched to the target domain.

Table 1 Example of extracted trigger pairs.

Triggering word	Triggered word
roudou (work)	*shifuto* (shift)
ame (rain)	*kasa* (umbrella)
shishutsu (expenses)	*kyasshu* (cash)
juutaku (housing)	*yachin* (rent)
isuramu (Islam)	*shuukyou* (religion)
mukashi (past)	*juurai* (former)
sodateru (to bring up)	*kyouiku* (education)
risuku (risk)	*kaihi* (avoidance)
teate (allowance)	*kyuufu* (payment)
kokusai (international)	*seiji* (politics)

4. Probability Estimation and Back-off Method

This section describes the probability estimation of the trigger pairs from the initial transcription, as well as a back-off scheme to incorporate trigger-based statistics derived from a large corpus.

4.1 Probability Estimation from Initial Transcription

The probabilities of the trigger pairs are estimated from the K-best ASR hypotheses by using a text window to calculate the co-occurrence frequency of the pairs inside it. Given a trigger pair $w_1 \rightarrow w_2$, this text window consists of the L words preceding w_2.

The probability of each trigger pair is computed as follows:

$$P_{TP}^{IT}(w_2|w_1) = \frac{N(w_1, w_2)}{\sum_j N(w_1, w_j)}, \quad (2)$$

where $N(w_1, w_2)$ denotes the number of times the words w_1 and w_2 co-occur within the text window, and j runs throughout all words triggered by w_1.

4.2 Proposed Trigger-Based Language Model

The proposed trigger-based language model is then constructed by linearly interpolating the probabilities of the trigger pairs with those of the baseline n-gram model, so that both long and short-distance dependencies can be captured at the same time.

The probability of the proposed language model for a word w_i given the word history $H = w_{i-L}, \ldots w_{i-1} \stackrel{\text{def}}{=} w_{i-L}^{i-1}$ is computed in the following way:

$$P_{LM}(w_i|H) = \frac{1}{L} \sum_{j=i-L}^{i-1} P_{LM}(w_i|w_j)$$

$$P_{LM}(w_i|w_j) = \begin{cases} P_{NG}(w_i|w_{i+1-n}^{i-1}), \text{ if } P_{TP}^{IT}(w_k|w_j) = 0, \ \forall k \\ \lambda P_{NG}(w_i|w_{i+1-n}^{i-1}) + (1-\lambda) P_{TP}^{IT}(w_i|w_j), \text{ else} \end{cases} \quad (3)$$

Here L is the number of words in the history H; P_{NG} is the probability of the n-gram component, which uses only the last $n-1$ words of H (i.e. $n \ll L$); P_{TP}^{IT} is the probability of the trigger-based component, computed by Eq. (2); and λ is the language model interpolation weight. When there are no words triggered by w_j, the n-gram model alone is used. Otherwise, the n-gram probabilities are linearly interpolated with the probabilities from the trigger pairs.

4.3 Back-off Method Using Statistics from Large Corpus

Since the amount of data provided by the initial transcription may be insufficient to obtain reliable probability estimates, a back-off scheme is introduced to combine the proposed model with the statistics estimated from a large corpus.

Another set of trigger pairs is extracted with the TF/IDF measure from a fraction of the large corpus similar to the target task. Then, the probabilities of the trigger pairs are computed from the whole corpus. We previously demonstrated that the method that selects trigger pairs from a matched corpus and estimates their statistics with a larger corpus is effective [11]. The resulting trigger pairs are similar to those used in the conventional trigger-based language model, except that the trigger pairs presented here are derived with the TF/IDF measure, instead of the AMI, and that they are extracted from a matched portion of the large corpus, instead of from the whole training set.

Then, we make use of this model to complement the proposed trigger-based language model described in Sect. 4.2. We have two different sets of trigger pairs: the trigger pairs constructed from the initial transcription (hereafter trigger set IT), and the trigger pairs extracted from the large corpus (hereafter trigger set LC). The trigger set IT provides a probability distribution more faithful to the target domain, whereas the trigger set LC offers a more reliable distribution that can cope with the problem of data sparseness that we discussed in [11].

The probability of the enhanced language model based on the back-off scheme $P_{BO}(w_i|w_j)$ is calculated in the following way:

$$\begin{cases} P_{NG}(w_i|w_{i+1-n}^{i-1}), \text{ if } P_{TP}^{IT}(w_k|w_j) = 0, P_{TP}^{LC}(w_l|w_j) = 0, \ \forall k, l \\ \lambda P_{NG}(w_i|w_{i+1-n}^{i-1}) + (1-\lambda) P_{TP}^{LC}(w_i|w_j), \text{ if } P_{TP}^{IT}(w_k|w_j) = 0, \ \forall k \\ \lambda P_{NG}(w_i|w_{i+1-n}^{i-1}) + (1-\lambda)\left(\delta P_{TP}^{LC}(w_i|w_j) + (1-\delta) P_{TP}^{IT}(w_i|w_j)\right), \\ \qquad \text{otherwise} \end{cases} \quad (4)$$

Here, P_{NG} is the probability of the n-gram component; P_{TP}^{IT} is the probability of the trigger set IT; P_{TP}^{LC} is the probability of the trigger set LC; λ is the language model interpolation weight; and δ is the trigger set interpolation weight. When there are no words triggered by w_j in either of the two trigger sets, the n-gram model alone is used. When there are no trigger pairs for w_j in the trigger set IT, the n-gram probabilities and the trigger set LC probabilities are linearly interpolated. Otherwise, all language models are linearly interpolated.

Note that if the trigger set IT is empty, that is, if we do not use the trigger pairs extracted from the initial transcription, the resulting model (first two entries in Eq. (4)) is similar to the conventional trigger-based language model, that is, a model whose trigger pairs are constructed from a large corpus. The differences are those we have just discussed. Hereafter we call this model the quasi-conventional trigger-based language model.

5. Perplexity Evaluation

In this section we present the experimental evaluation of the proposed language model by test-set perplexity.

5.1 Experimental Setup

The ASR system Julius 3.5-rc2 [12] was used for speech recognition. The baseline language model was a linear interpolation of word trigram models constructed from the Corpus of Spontaneous Japanese (CSJ) [13] (3.5 M words) and the minutes of the National Diet of Japan (71 M words) with an interpolation weight of 0.5. The size of the vocabulary was 30 K words, and the out-of-vocabulary (OOV) rate was 1.56%. The acoustic model was a shared-state triphone HMM trained with the CSJ [14]. The average word recognition accuracy with this baseline model was 55.2%. We obtained this relatively low accuracy because the utterances are truly spontaneous and often uttered very fast.

The minutes of the National Diet from the year 2001 (17 M words) were used for calculating the IDF part used in the trigger pair extraction of the set IT and also to extract the trigger pairs of the set LC.

5.2 Parameter Optimization

The parameters of all models were optimized by dividing the test set into two. The first 5 programs were used to empirically tune the parameters used in the other 5 programs and vice versa. The parameters were optimized by means of the perplexity.

The resulting optimal number of hypotheses from the initial transcription K used for extracting the trigger pairs and estimating their likelihood was 2. The threshold for the TF/IDF value was 0.0005. The average word history size L was 26, and the trigger set interpolation weight δ was 0.07. Figures 2 and 3 show the perplexity for different values of K and L, respectively. We can see that the perplexity is not sensitive to these values.

The optimal language model interpolation weight λ was, on average, 0.55 for the proposed trigger-based model (Eq. (3)), 0.67 for the quasi-conventional model (Eq. (4) without last entry), and 0.56 for the back-off method (Eq. (4)). The value of λ is larger for the quasi-conventional

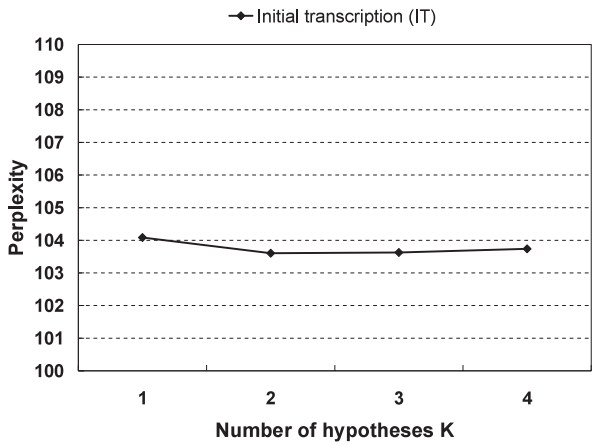

Fig. 2 Perplexity for different values of the number of hypotheses K.

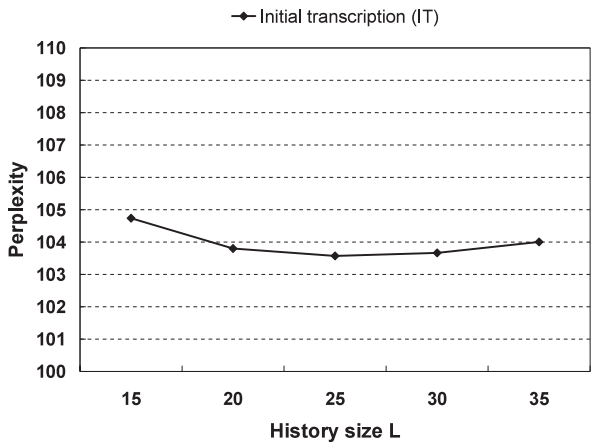

Fig. 3 Perplexity for different values of the history size L.

model than for the proposed models, because the trigger pairs are not task-dependent in the former model and, therefore, less beneficial in the interpolation.

In the experiments of perplexity evaluation, it turned out, after optimization, that the best performance was obtained when filtering with stop words, confidence score, and large corpus were not incorporated.

5.3 Experimental Results

We evaluated the test-set perplexity for the 10 programs by three different models: the quasi-conventional trigger-based model using only a large corpus (LC), the proposed trigger-based language model using only the initial transcription (IT), and the back-off method (IT+LC). For reference, we also evaluated the model constructed by deriving the trigger pairs from the correct transcription.

The perplexity and its reduction averaged over the 10 programs are shown in Table 2. The proposed language model (IT) achieved a reduction of 30.66% over the baseline, much greater than the reduction obtained with the quasi-conventional model (LC). This demonstrates the effectiveness of the proposed approach.

Table 2 Comparison of trigger-based language models constructed by different methods.

Model	Perplexity	Reduction (%)
Baseline	150	–
Large corpus (LC)	121	19.33
Initial transcription (IT)	104	30.66
Back-off model (IT+LC)	102	32.00
(cf.) Correct transcription	73	51.33

The back-off method improved the perplexity slightly, but not significantly. This suggests that the initial transcription provides trigger pairs that are much more adapted to the task than those constructed from the large corpus, so the benefit obtained from the latter is minimal. We expect that the proposed back-off scheme can be useful when the initial transcription is smaller in size.

The perplexity reduction by the proposed method was smaller than that obtained with the model that used the correct transcription. The baseline word recognition accuracy is 55.2%, meaning that about half of the initial transcription is erroneous, so the results are consistent with this fact.

We also constructed a trigger-based language model from the initial transcription by using the AMI [1], [2], instead of the TF/IDF measure. The perplexity was 104, which is comparable to that obtained when using the TF/IDF measure. It was observed that more trigger pairs were extracted by the TF/IDF measure, so we expect that this measure should be more effective for shorter discussions.

We also investigated the improvements for correctly recognized words and incorrectly recognized ones in the initial transcription. The average perplexity for correctly recognized words was 75 by the baseline model and 49 by the proposed model, whereas, for the incorrectly recognized words, the perplexity was 408 and 298, respectively. That is, we obtained a reduction of 34.66% for the correctly recognized words and an also significant 26.96% reduction for the incorrectly recognized ones. The fact verifies that the perplexity was also improved significantly for incorrect words, showing a potential of improvement in speech recognition accuracy.

The average number of trigger pairs was 128 K in the trigger set IT, 9158 K in the trigger set LC, and 71 K from the correct transcription. The average hit rate of the trigger pairs in the test set was 31% for the first case, 33% for the second, and 35% for the third. We can see that the set IT efficiently covers the test set with a much smaller number of trigger pairs than the set LC. This is because the pairs from the set LC are not task-dependent. The back-off method had slight impact on the perplexity because the hit rate by using the set LC is only a little greater than that by the set IT.

The model constructed from the initial transcription used 606 self-triggers on average during the test-set perplexity evaluation, while 26,555 non-self-triggers were used. This is a significant difference with the conventional works on trigger-based language models, where non-self-triggers offered little benefit over self-triggers. In contrast to previous works, the trigger pairs in the proposed approach are

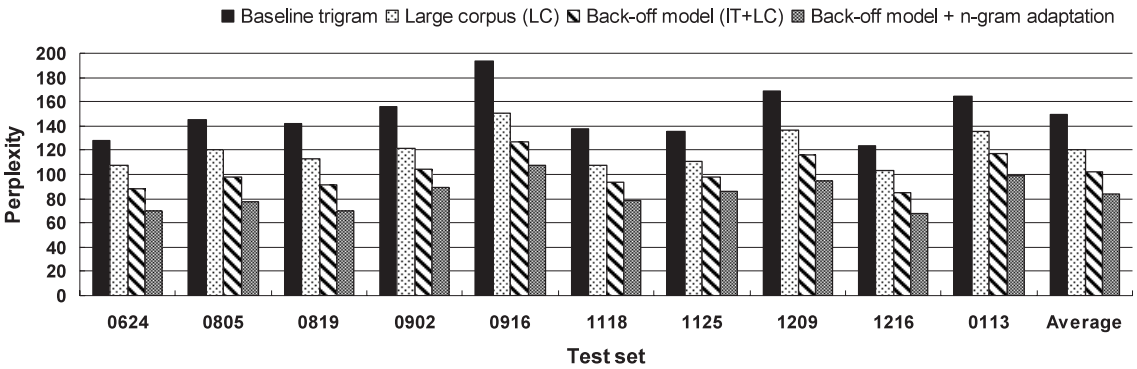

Fig. 4 Perplexity evaluation of trigger-based language models for different topics.

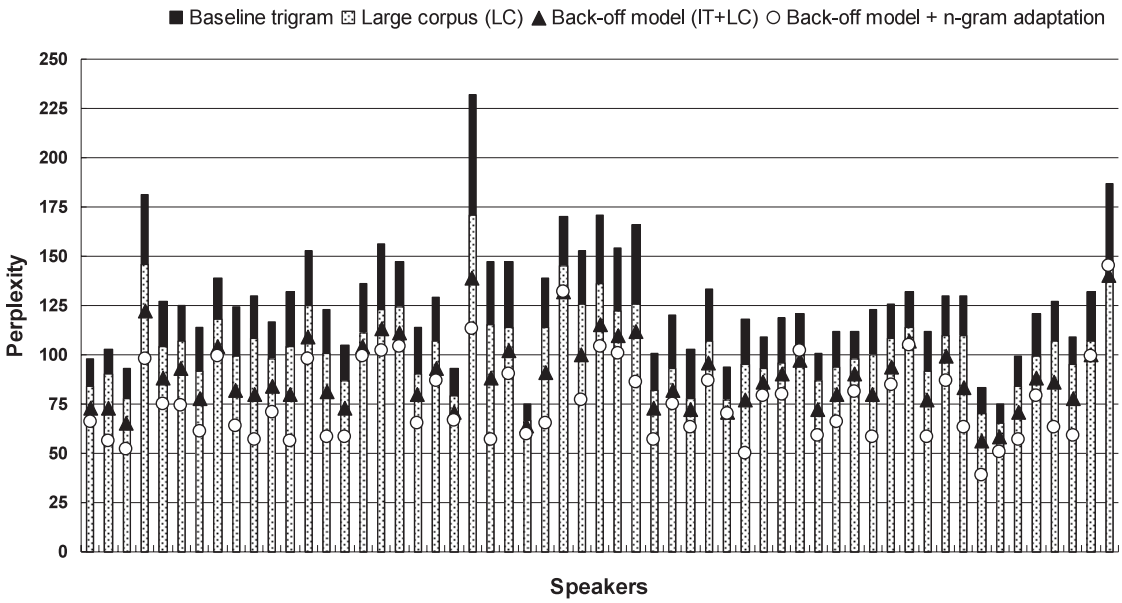

Fig. 5 Perplexity evaluation of trigger-based language models for different speakers.

task-dependent and make a better match for the target task.

5.4 Comparison and Combination with n-gram Model Adaptation

Next, we use the initial transcription also to create an adapted n-gram language model in order to compare its performance with that of the proposed approach. We then combine this with the proposed model for further improvement.

We take the J-best hypotheses from the initial transcription for creating a back-off n-gram model. A trigram model was constructed from each of the 10 test sets, and then interpolated with the baseline trigram model. The value J was optimized with the method discussed in Sect. 5.2, yielding the value 10.

The resulting interpolated trigram was then combined with the trigger-based language model. Table 3 shows the results of the perplexity evaluation. The perplexity reduction by the n-gram adaptation is smaller than that by the proposed trigger-based adaptation, and their combination achieved a notable maximum perplexity reduction of 44% over the baseline trigram model. Although the improvement is not additive, the n-gram model adaptation serves as a good complement for the proposed approach.

Table 3 Comparison and combination of the proposed method with the adapted n-gram.

Model	Perplexity	Reduction (%)
Baseline	150	–
Adapted n-gram	119	20.66
+ Initial transcription (IT)	87	42.00
+ Back-off model (IT+LC)	84	44.00

Figures 4 and 5 show the perplexity by several of the constructed language models for each of the topics (test discussions) and speakers, respectively. As can be observed, the results are fairly consistent across the different topics and speakers.

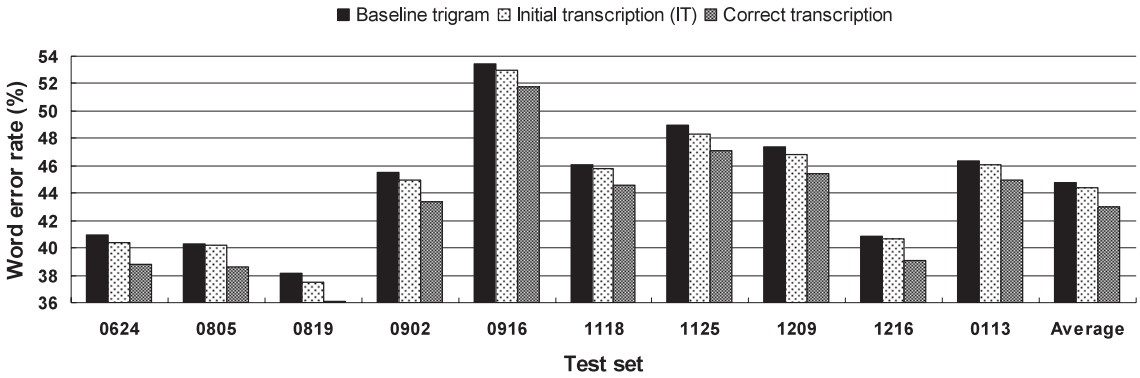

Fig. 6 Word error rate improvement by the trigger-based language model.

6. Speech Recognition Evaluation

This section presents a scheme for rescoring word graphs by the proposed language model and the experimental results in terms of speech recognition accuracy.

6.1 Word Graph Rescoring

The ASR system Julius generates a word graph with acoustic, language, and confidence scores for each node. The experimental setup is the same as in Sect. 5.1.

Then, we use a stack decoding search for parsing the word graph to find the most likely sentence hypothesis [15]. During the search, we use the proposed trigger-based language model to recalculate the language model scores, by discounting the baseline language model probability from the per-node combined score and then adding the proposed language model probability. The word history is formed with the 1-best hypotheses of the preceding utterances and with the words that make up the partial path in the current utterance.

6.2 Experimental Results

We evaluated the word error rate (WER) for each of the 10 programs of the test set. In this section, filtering with stop words, confidence score, and large corpus were incorporated. Here also, we conducted the two-fold cross validation described in Sect. 5.2. The resulting average confidence threshold was 0.05, and the average word history size was changed to 43.

Figure 6 shows the results obtained by the proposed language model (IT) and those by the model constructed from the correct transcription. We obtained a relative 0.98% improvement in WER. This improvement, although small, is statistically significant, with a p-value of 0.022.

We also examined the WER when using the AMI instead of the TF/IDF measure, and we obtained no significant difference. In addition, we investigated the WER without using the confidence score filtering. In this case, we obtained a 0.91% improvement, so the confidence score filtering has some effect in reducing erroneous trigger pairs.

The reasons why the obtained improvement in WER is much smaller than the perplexity reduction by the proposed language model are presumed as follows. First, although the reduction in perplexity for incorrectly recognized words is significant, the perplexity value is still very large (reduced from 408 to 298), so it is hard to improve the recognition accuracy. Second, when we calculate the perplexity, the word history does not contain any errors, so the predictors are much better than those used in the speech recognition experiments. Conversely, the word history contains errors during the word graph rescoring, thus a history size greater than that used in the perplexity evaluation was needed. Finally, the word graph we rescore has the apparent limitation that the correct words might not be in any of the nodes. We expect that a re-decoding scheme with the adapted model would realize a greater improvement as shown in [9], [16], whose perplexity reductions are much smaller than the one obtained in this work. With the correct transcription, the relative WER improvement was 4.07%, much greater than that obtained with the initial transcription, so we anticipate better results in tasks with higher baseline ASR performance.

7. Conclusion

We have presented a novel trigger-based language model adaptation based on initial speech recognition results. A significant improvement in perplexity was achieved over both the baseline trigram and the typical trigger-based model constructed from a large corpus. A further improvement was obtained by combining with n-gram model adaptation. In addition, the speech recognition accuracy was also improved with the proposed language model.

The proposed approach is particularly useful in tasks where large amounts of training data are not readily available but the test set is long, since we have observed that the initial transcription is a good source for deriving the trigger pairs. This is specifically true for many transcription tasks.

Acknowledgments

We would like to thank Dr. Yuya Akita of Kyoto University for his priceless help throughout this research, and Dr. Shinsuke Mori of IBM Japan and Dr. Hirofumi Yamamoto of ATR for their useful comments and suggestions.

References

[1] R. Lau, R. Rosenfeld, and S. Roukos, "Trigger-based language models: A maximum entropy approach," Proc. ICASSP, vol.2, pp.45–48, 1993.
[2] R. Rosenfeld, "A maximum entropy approach to adaptive statistical language modeling," Comput. Speech Lang., vol.10, pp.187–228, 1996.
[3] C. Tillmann and H. Ney, "Selection criteria for word trigger pairs in language modeling," International Colloquium on Grammatical Inference, pp.95–106, 1996.
[4] C. Tillmann and H. Ney, "Word triggers and the EM algorithm," Proc. ACL Special Interest Group Workshop on Computational Natural Language Learning, pp.117–124, 1997.
[5] R. Khun and R. De Mori, "A cache-based natural language model for speech recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol.12, no.6, pp.570–583, 1990.
[6] P. Clarkson and A. Robinson, "Language model adaptation using mixtures and an exponentially decaying cache," Proc. ICASSP, vol.2, pp.799–802, 1997.
[7] J. Bellegarda, "Exploiting latent semantic information in statistical language modeling," Proc. IEEE, vol.88, no.8, pp.1279–1296, 2000.
[8] C. Chelba and F. Jelinek, "Recognition performance of a structured language model," Proc. Eurospeech, vol.4, pp.1567–1570, 1999.
[9] Y. Akita and T. Kawahara, "Language model adaptation based on PLSA of topics and speakers for automatic transcription of panel discussions," IEICE Trans. Inf. & Syst., vol.E88-D, no.3, pp.439–445, March 2005.
[10] G. Salton, "Developments in automatic text retrieval," Science, vol.253, pp.974–980, 1991.
[11] C. Troncoso, T. Kawahara, H. Yamamoto, and G. Kikui, "Trigger-based language model construction by combining different corpora," IEICE Technical Report, SP2004-100, 2004.
[12] T. Kawahara, A. Lee, K. Takeda, K. Itou, and K. Shikano, "Recent progress of open-source LVCSR engine Julius and Japanese model repository," Proc. ICSLP, pp.3069–3072, 2004.
[13] K. Maekawa, H. Koiso, S. Furui, and H. Isahara, "Spontaneous speech corpus of Japanese," Proc. LREC, vol.2, pp.947–952, 2000.
[14] T. Kawahara, H. Nanjo, T. Shinozaki, and S. Furui, "Benchmark test for speech recognition using the corpus of spontaneous Japanese," Proc. ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition, pp.135–138, 2003.
[15] T. Kawahara, C.-H. Lee, and B.-H. Juang, "Flexible speech understanding based on combined key-phrase detection and verification," IEEE Trans. Speech Audio Process., vol.6, no.6, pp.558–568, 1998.
[16] H. Nanjo and T. Kawahara, "Language model and speaking rate adaptation for spontaneous presentation speech recognition," IEEE Trans. Speech Audio Process., vol.12, no.4, pp.391–400, 2004.

Carlos Troncoso received the B.S. degree in Computer Science in 1999 from the University of Seville, Seville, Spain, and the M.S. degree in Information Science in 2003 from the Japan Advanced Institute of Science and Technology, Ishikawa, Japan. He is currently in the Japan Advanced Institute of Science and Technology as a Ph.D. candidate, and as a research student in Kyoto University, Kyoto, Japan. From 2003 to 2004, he was a student intern at ATR Spoken Language Translation Research Laboratories, Kyoto, Japan. His research interests include statistical language modeling and automatic recognition and understanding of speech. He is a member of the Acoustical Society of Japan.

Tatsuya Kawahara received the B.E. degree in 1987, the M.E. degree in 1989, and the Ph.D. degree in 1995, all in Information Science, from Kyoto University, Kyoto, Japan. In 1990, he became a Research Associate with Department of Information Science, Kyoto University. From 1995 to 1996, he was a Visiting Researcher at Bell Laboratories, Murray Hill, NJ, USA. Currently, he is a Professor with Academic Center of Computing and Media Studies, Kyoto University. He is also an Invited Researcher at ATR Spoken Language Translation Research Laboratories. He has published more than 100 technical papers covering speech recognition, confidence measures, and spoken dialogue systems. He has been managing several speech-related projects in Japan, including a free large vocabulary continuous speech recognition software project (http://julius.sourceforge.jp). Dr. Kawahara received the 1997 Awaya Memorial Award from the Acoustical Society of Japan and the 2000 Sakai Memorial Award from the Information Processing Society of Japan. Since 2003, he has been a member of the IEEE SPS Speech Technical Committee.