Effect of Confusion Network Combination on Speech Recognition System for Editing

Satoshi Ishimaru*, Hiromitsu Nishizaki† and Yoshihiro Sekiguchi†
* Department of Education Interdisciplinary Graduate School of Medicine and Engineering, University of Yamanashi, Japan
E-mail: ishimaru@alps-lab.org Tel/Fax: +81-55-220-8361/8776
† Department of Research Interdisciplinary Graduate School of Medicine and Engineering, University of Yamanashi, Japan
E-mail: {hnishi,sekiguti}@yamanashi.ac.jp Tel/Fax: +81-55-220-8361/8776

***Abstract*—This paper describes confusion network combination (CNC), which integrates multiple confusion networks, and its effectiveness on a system for editing transcription by a speech recognizer. It has been proposed that integration by CNC produces better recognition performance. We believe that this improves the working efficiency of a human editor for correcting errors. We utilized confusion networks from two recognition systems for CNC. Integration by the CNC method is performed by combining the networks based on posterior probabilities attached to each word. The experimental results showed that the improvement of recognition performance by the CNC method could reduce the working time of human editors by 5.1 seconds, on average, compared to the working time required without using this method.**

I. Introduction

Recently, applications using speech recognition technologies have been developed and used practically. For example, some movies on YouTube have captions that are automatically added by speech recognition systems [1]. In addition, Japan Broadcasting Corp. (called "NHK" in Japan) has started captioning all news programs using speech recognition technologies [2].

We are developing an automatic captioning system for classroom lectures for students with hearing loss. There are some problems in automatic captioning of speech. One of them is the speech recognition error problem. If some speech recognition errors occur in a caption, accurate information is not carried to students, and the students may misunderstand the information. Therefore, to avoid this problem, it is important to refine the speech recognition technologies. However, it is impossible to completely eliminate recognition errors.

Errors must be corrected by a human (or humans) for a captioning system, and the human needs to correct errors fast in the case of a real-time system. It is necessary to develop an editing system with a user-friendly interface to obtain an error-free transcription by a speech recognizer. For example, Ogata et al. [3] developed "PodCastle," a social annotation system of Podcast speeches. The system provides the error correction interface and everyone can edit transcriptions of Podcast speeches through the Internet.

In this paper, we describe both the development of an editing system for correcting recognition errors and a confusion network combination (CNC) method. The goal of this paper is to reduce the working time of humans required for correcting errors by introducing the proposed CNC method.

One of the main factors for reducing correction time is the ability to display as many word candidates as possible on an edit screen. Then, the editor can correct errors by only using mouse or performing touch operation. In other words, edit time can be reduced without keyboard operation. However, too many word candidates may confuse an editor. Therefore, suitable candidates which are displayed on the screen must be selected.

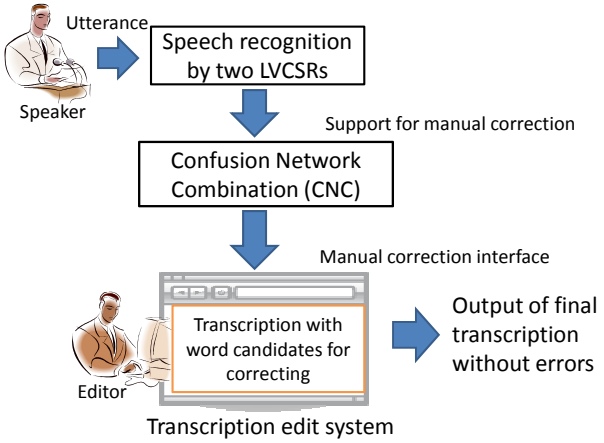

Fig. 1. Outline of error correction with CNC.

Our CNC method uses two types of speech recognizer, because it is known that using multiple speech recognizers, such as "ROVER" [4] and CNC [5], improves speech recognition performance. The two recognizers produce two types of transcription, each of which uses a different word set. Our method combines the confusion networks output by the two speech recognizers based on posterior probabilities attached to each word.

In the correction experiment, our CNC method improved speech recognition performance. This made it possible for a human editor to correct recognition errors faster.

II. Manual Correction Interface

Figure 1 shows an outline of an error correction framework using a transcription edit system with CNC for reducing speech recognition errors.

First, an utterance is recognized by two speech recognition systems. We commonly used Julius [6], an open source of a large vocabulary continuous speech recognition engine, as a decoder in the two recognition systems. A language model is also commonly used in the systems. The two recognition systems differ according to the type of acoustic models used. We prepared two types of acoustic model.

Next, our CNC method combines the confusion networks derived by the two recognizers. The arranged transcription

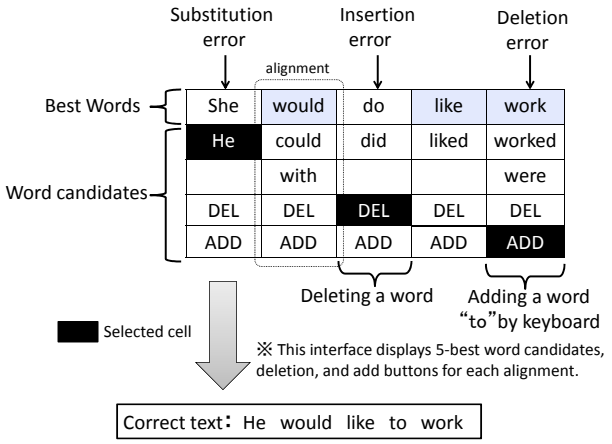

Fig. 2. Example of correction interface.

created by the method is displayed on the graphical user interface of the editing system. An editor can find the recognition errors and correct them by replacing wrong words with correct words.

Figure 2 shows a screenshot example of our editing system. The word sequence on the row labeled "Best Words" is the best recognition hypothesis (1-best) made by our CNC method. The words in the "Word candidates" rows are candidates for making corrections, and each candidate is aligned to each 1-best word. The interface can display a maximum of 5-best candidates for each alignment because too many candidates may damage the efficiency of making corrections. Each alignment has "DEL" and "ADD" buttons. An alignment is deleted from the "Best Words" line by touching (or clicking) the DEL button. If an editor touches (or clicks) on a candidate cell, the 1-best word corresponding to the touched (or clicked) cell is replaced with the word in the cell.

By repeating this action, the editor can obtain an error-free transcription of the utterance. However, if the correct word is not in any cell, the editor has to input the correct word by using a keyboard. The form to input a word is represented on the screen by pushing the ADD button, and the word is added when an editor finishes inputting the word. Using a keyboard increases the working time of an editor. Therefore, it is important to display as many candidates as possible on the screen. This reduces the correcting time and the work load of the editor.

Our proposed CNC method can achieve this. We explain the method in the following section.

III. Confusion Network Combination

Our proposed CNC method is based on a combination of confusion networks derived by the two speech recognizers. The Julius decoder can recognize an input utterance, and can also output a confusion network formed transcription. This method has the following steps:

Step(1): Preparing two types of confusion network by two different speech recognizers.
Step(2): Performing a sub-network-based alignment between the confusion networks.
Step(3): Composing a new confusion network by combining the two networks, based on posterior probabilities.

The details are explained in following sections.

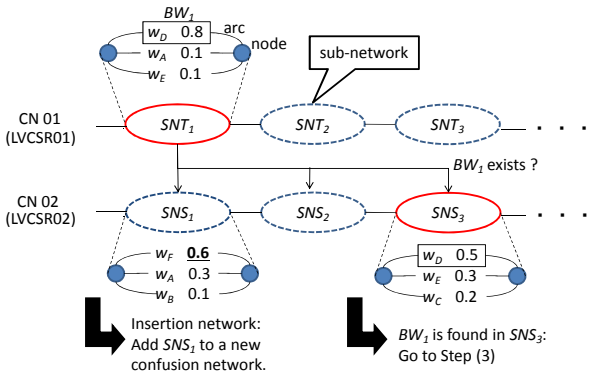

Fig. 3. Alignment example, where BW_i is found in SNS_j.

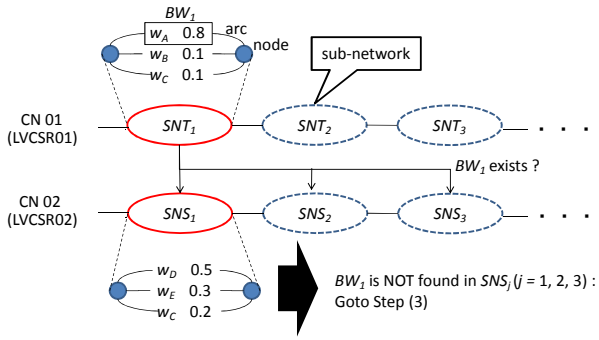

Fig. 4. Alignment example, where BW_i is NOT found in SNS_j.

A. Step(1): Two confusion networks

The first step of our correction method is to prepare two types of confusion network derived by the two different speech recognition systems. As mentioned above, we prepared two types of acoustic model: triphone-based Hidden Markov Model (HMM) and syllable-based HMM. However, the language model and the recognition dictionary are commonly used in the same recognition decoder Julius.

It is known that different types of phonological modeling unit provide a different recognition result [7]. Therefore, using two recognizers may form a better transcription (confusion network) than that formed by using only one recognizer.

Figure 3 shows an example of confusion networks. There are some arcs between two successive nodes. Each arc has a word with posterior probability. In this paper, we call the arc set between two successive nodes a "sub-network."

B. Step(2): Sub-network alignment

The second step of our CNC method is to perform a sub-network-based alignment between the two confusion networks. The confusion network derived by the recognizer with triphone-based HMM provides the basis for the alignment process.

We define "CN01" and "CN02" as the confusion network derived by the recognizer with triphone-based HMM and syllable-based HMM, respectively. Suppose that the number of sub-networks of CN01 and CN02 is I and J, respectively. The i-th sub-network of CN01 and the j-th sub-network of CN02 are denoted as "SNT_i" and "SNS_j," respectively. The process has the following steps:

1) For $i = 1, 2, \cdots, I$, the following Step (2) to (5) are repeated.

2) The Best Word BW_i, which has the highest probability among SNT_i, is searched in SNS_j ($j = p, p+1, p+2$). p is a pointer, indicating the alignment point and is dynamically updated, depending on the alignment of SNT_i and SNS_j. The initial value of p is 1.
3) If BW_i is found in any SNS_j ($j = p, p+1, p+2$, $1 \leq j \leq J$), SNT_i corresponds to SNS_j, which is the sub-network including the same word of BW_i. Then, p is updated to j.
4) If BW_i is NOT found in any SNS_j, SNT_i corresponds to SNS_j, which is regarded as a substitution error of SNT_i. Then, p is updated to j.
5) p is incremented.

All SNT_i are not aligned to one SNS_j and all SNS_j are not aligned to one SNT_i. We deal with these sub-networks as insertion sub-networks.

Figure 3 shows an alignment case where the BW_1 is found in SNS_3. In this case, SNS_1 is the insertion network in that the maximum posterior probability is 0.5 or more. In our correction method, all insertion networks are added to the new confusion network created by Step(3).

On the other hand, Figure 4 shows an alignment case where the BW_1 is not found in any SNS_j. However, we assume that the SNT_1 corresponds to the SNS_p ($p = i = 1$) because the SNS_p ($p = i$) is likely to be the substitution error of SNT_1.

C. Step(3): Composing new confusion network

The final step is to merge the two confusion networks and create a new confusion network. The merging process is performed by calculating the average posterior probability of the words in the aligned sub-networks' pair.

Figure 5 shows an example of merging the sub-networks that correspond to each other, from the previous Step(2). Figure 5 shows a case where SNT_i and SNS_j correspond. Each word in the sub-networks has a posterior probability. The new, marged sub-network is defined as "SNC." The probabilities of words belonging to the SNC_i are calculated by averaging the probabilities of the same word in SNT_i and SNS_j. If the word is only in one of the sub-networks, its probability is halved in the new sub-network.

Finally, the word w_B, which has the highest probability, is likely to be the correct word. The remaining words such as w_A are candidates for correction by an editor. In our editing system, a maximum of 5-best words are displayed on the interface. In the case of Figure 5, the word w_D is not provided.

D. Advantages of CNC method

The CNC method has three advantages for our editing system.

The first advantage is correcting substitution errors in our system. Our method can replace a wrong word on the Best Word position with a correct word with a lower posterior probability. In the case of Figure 5, w_B does not have the highest probability in the two sub-networks. However, it can be in the Best Word position when considering the two sub-networks.

The second advantage is recovering deletion errors. As described in Section III-B, our method adopts two types of confusion network. This can prevent some deletion errors because all insertion networks are merged into the new confusion network. On the other hand, it may increase insertion errors and confuse a human editor. However, the insertion errors can be easily removed by pushing "DEL" button on the interface.

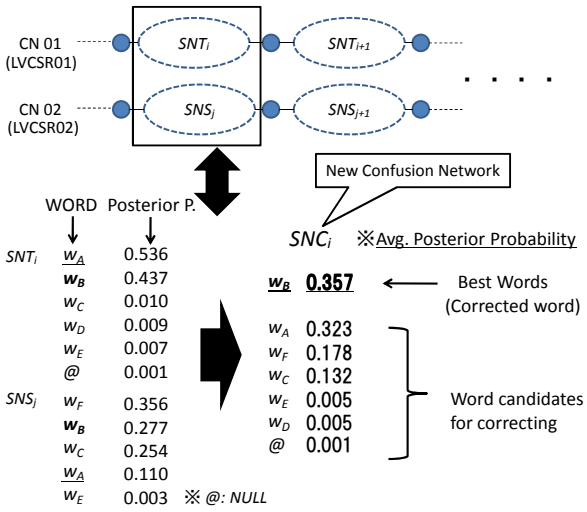

Fig. 5. Compose a new confusion network.

The final advantage is being able to form transcriptions with greater confidence. After processing CNC, all words on the Best Words line have a high posterior probability. In addition to this, if a word is recognized and belongs to the two sub-networks with a high probability, we can assume that the word has a high degree of confidence and the word may be correct. It can contribute to the reduction of cost of editing by a human editor.

IV. Error correction experiment

A. Experimental setup

The main purpose of the error correction experiment was to investigate the effectiveness of our CNC method on the editing transcription system for the corrections made by editors. Therefore, we evaluated how much the improvement in speech recognition performance the CNC method achieves. In addition to this, we evaluated the time required for error correction using the editing system both with and without the CNC method. In other words, we evaluated whether the error correction reduced an editor's burden or not.

The sample of subjects consisted of 14 students who were used to keyboard operation. The experimental procedure was performed in the following order:

1) A subject utters something to the system.
2) The CNC method is performed when the system gets the two types of confusion network.
3) The transcription with correction candidates is displayed on the interface of the system.
4) The subject corrects the errors included in the transcription using the editing system.

Each of the subjects uttered a total of 15 sentences, which were selected from the Japan Newspaper Article Speech (JNAS) corpus provided by the Acoustic Society of Japan (ASJ) [8]. The duration of the sentences varied from 4 seconds to 6 seconds, and the sentences consisted of about 12 words.

TABLE I
SPEECH RECOGNITION RATE BEFORE THE SUBJECTS CORRECT RECOGNITION ERRORS.

Utterance Group	Hypothesis	Corr.[%]	Acc.[%]	Sub	Del	Ins	Cover_Rate[%]
Group1	Triphone	67.8	61.5	34.1	3.6	7.3	72.2
	Syllable	70.9	63.9	30.9	3.1	8.3	74.6
	CNC	**71.2**	61.5	31.3	**2.4**	11.3	**77.8**
Group2	Triphone	36.2	18.9	36.7	2.9	10.7	38.2
	Syllable	39.2	18.7	35.4	2.3	12.7	41.2
	CNC	38.3	10.8	36.3	**2.0**	17.0	**42.2**

Ten of the 15 sentences had fewer than 2 or no words that were out-of-vocabulary (OOV). We classify these as "Group1" utterances. The others have about 4 OOV words (the OOV rate is about 30%). These are called "Group2" utterances.

Both types of acoustic model come from the JNAS corpus [8]. A word trigram-based language model, with 20,000 words of vocabulary, was derived from Mainichi newspaper articles.

B. Experimental result and discussion

Table I shows the recognition rates of each utterance group (Group1, Group2). All rates are averaged by the number of subjects. "Corr." means word correct rate, which does not consider any insertion errors, while "Acc." is word accuracy rate, which considers insertion errors (Acc.$= 1-$WER). "Cover_Rate" means the coverage of correct words displayed on the correction interface of the editing system. A higher Cover_Rate makes an editor's work easier. The lines labeled "Triphone" and "Syllable" represent the performance of the confusion networks derived by the recognizer with triphone-based HMM and syllable-based HMM, respectively. On the other hand, the lines labeled "CNC" show the performance of a confusion network composed by our CNC method. Note that "Corr." and "Acc." are calculated on the basis of Best Words sequences.

Figure 6 shows the editing time for correction work. Our CNC method made a reduction of 5.1 seconds in error correction time, on average, for the Group1 sentences. This is why it obtained a 3.2% improvement in Cover_Rate.

However, in the case of Group2 setences, these do not seem to be a benefit of using the CNC method, although the Cover_Rate was slightly improved. In the context of recognizing an utterance including too many OOV words, these words cannot be recognized correctly. The editor has to edit them using a keyboard. It takes time to correct errors. Therefore, there was no difference between the editing duration with and without use of the CNC method.

We suggest that there are two effects of the CNC method of speech recognition errors on the editing system, and that these are as follows:

- The CNC method makes a significant contribution to a human editor when OOV rate is low.
- The CNC method can improve the Cover_Rate.

These contributions have led to a reduction in recognition errors, and have prevented the editors from having to input correct words using a keyboard. Another contribution is the reduction in time spent in editing errors.

V. CONCLUSION

In this paper, we have presented both an editing system for correcting errors in transcription formed by speech recognizers and a CNC method. Our CNC method uses two speech recognizers, and corrects errors by combining confusion networks based on posterior probability. The result of the error correction experiment showed that our CNC method worked efficiently, and reduced the editing time for the sentences containing 2 or fewer OOV words.

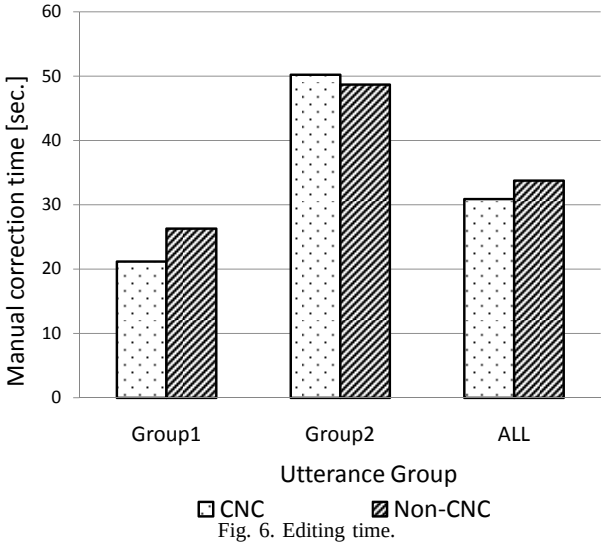

Fig. 6. Editing time.

In future work, we intend to improve the user interface of the editing system for making speedy corrections. For example, each candidate can be color-coded according to the level of confidence. This enables an editor to find correction words of more easily. In addition, we are going to refine our CNC framework. Using multiple recognizers' outputs has possibilities of reducing recognition errors.

REFERENCES

[1] K. Harrenstien, "Automatic captions in YouTube," The Official Google Blog, Online: http://googleblog.blogspot.com/2009/11/automatic-captions-in-youtube.html, accessed on 28 Mar 2011.

[2] S. Homma, A. Kobayashi, T. Oku, S. Sato, T. Imai, and T. Takagi, "New Real-Time Closed-Captioning System for Japanese Broadcast News Programs," in *ICCHP 2008*, vol. LNCS 5105, 2008, pp. 651–654.

[3] J. Ogata, M. Goto, and K. Eto, "Automatic Transcription for a Web 2.0 Service to Search Podcasts," in *INTERSPEECH 2007*, 2007, pp. 2617–2620.

[4] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)," in *IEEE ASRU Workshop*, 1997, pp. 347–352.

[5] G. Evermann and P. Woodland, "Posterior probability decoding, confidence estimation and system combination," in *NIST Speech Transcription Workshop*, 2000, college Park, MD.

[6] A. Lee, T. Kawahara, and K. Shikano, "Julius — an open source real-time large vocabulary recognition engine," in *EUROSPEECH 2001*, 2001, pp. 1691–1694.

[7] T. Utsuro, Y. Kodama, T. Watanabe, H. Nishizaki, and S. Nakagawa, "Multiple LVCSR Model Combination by Machine Learning," in *HLT-NAACL 2004*, 2004, pp. 13–16.

[8] K. Itou, M. Yamamoto, K. Takeda, T. Takezawa, T. Matsuoka, T. Kobayashi, K. Shikano, and S. Itahashi, "The Design of the Newspaper-Based Japanese Large Vocabulary Continuous Speech Recognition Corpus," in *5th ICSLP*, 1998, pp. 3261–3264.