Error Correction of Automatic Speech Recognition Based on Normalized Web Distance

E. Byambakhishig, K. Tanaka, R. Aihara, T. Nakashika, T. Takiguchi, Y. Ariki

Graduate School of System Informatics, Kobe University, Japan

byamba@cs.scitec.kobe-u.ac.jp, takigu@kobe-u.ac.jp, ariki@kobe-u.ac.jp

Abstract

In this paper, we focus on the problems associated with error correction of automatic speech recognition (ASR) based on confusion networks. The problems discussed are the availability of corpus in terms of calculating the semantic score and performance degradation for error correction using N-gram due to the null transitions in the confusion networks. In attempt to solve these problems, first, we employ Normalized Web Distance as a measure for semantic similarity between words that are located far from each other. The advantage of Normalized Web Distance is that it may use the Internet and so on for learning semantic similarity, which might solve the problem of corpus availability. Secondly, an error correction model without null nodes in confusion networks is trained using conditional random fields in order to improve the performance of error correction using N-grams.

Index Terms: confusion network, conditional random fields, word-error correction, normalized web distance

1. Introduction

Speech technology is now widely used in the field of speech archiving, such as PodCastle [1] on WWW or MIT lecture browser [2]. In these systems, to read the speech in words or to retrieve the proper passages using keywords, a low word-error rate (WER) is necessary. A language model can contribute to selecting the most plausible words among the candidates presumed by the acoustic model. However, if the acoustic score of the false word is high, it may be selected irrespective of the language model.

To solve this problem, some methods have been proposed to learn and correct whether each utterance is linguistically natural or not, using a discriminative model. In a discriminative model, features for learning and testing are vital for the performance and N-gram features and confidence scores are often used as features for ASR error corrections. Although N-gram features only consider the few words around a corresponding word, but not the words located far from the word in utterance. Moreover, the degradation of N-gram correction is blatant, if there are many recognition errors and null transitions in the confusion networks. There are some methods that consider the relevance with the words located far in utterance. However, there are problems such as availability of corpus and the computational complexity caused from the corpus size increase [3].

To solve these problems, we employ Normalized Web Distance (NWD) [4] as a measure for semantic similarity between words that are located far from each other. The advantage of Normalized Web Distance is that it uses the Internet, search engines, and transcripts as a database, which can solve the problem of corpus availability and the computational complexity. In the proposed method, we begin by correcting the speech recognition errors based on long-distance and short-distance context using the Normalized Web Distance score. Then we delete the null transitions in the confusion networks from its output to make N-grams effective for learning and correcting for its second run. In this paper, error correction is done by using conditional random fields (CRF) [5], and a confusion network [6] is used as the competition hypotheses. A confusion network was proposed for compact representation of the speech recognition results. This paper is constructed as follows. In Section 2, the flow of the proposed method is discussed. In Sections 3 and 4, long contextual information and a word-error correction method are described, respectively. In Section 5, the experiment results are shown. The conclusion is described in Section 6.

2. Flow of proposed method

Figure 1 shows the flow of the proposed method. The "Learning detection model" process shows the learning process of the error detection model based on Normalized Web distance. First, speech data are recognized and recognition results are output as a confusion network. Second, each word on the confusion network is labeled as false or true after the similarity scores of the words are computed using Normalized Web Distance. Then the error detection model is trained by CRF using features of unigram, bigram, trigram, and posterior probability on the confusion network and NWD similarity score. We obtain two types of error correction models during this process: the "Error detection model with null nodes", which we obtain without deleting the null transitions in the confusion network, and "Error correction model without null nodes", which we obtain by deleting all the null corrections from the training data. In the "Test" process, the confusion network is produced in the same way from the input speech and the NWD score is computed. Then word re-ranking is carried out on the confusion network using the first "Error detection model with null transitions". After that, null transitions that are labeled True are deleted from the output of the first re-ranking result, and the second re-ranking is carried out using the "Error detection model without null transitions".

3. Long contextual information

3.1. Normalized Web Distance

NWD is a method that has been proposed to determine the similarity between words and phrases, and is derived from Normalized Information Distance. Normalized Information Distance includes Kolmogorov complexity in its definition. However Kolmogorov Complexity is not computable for all given inputs, which leads to computability problems when working with Normalized Information Distance. Normalized Web Dis-

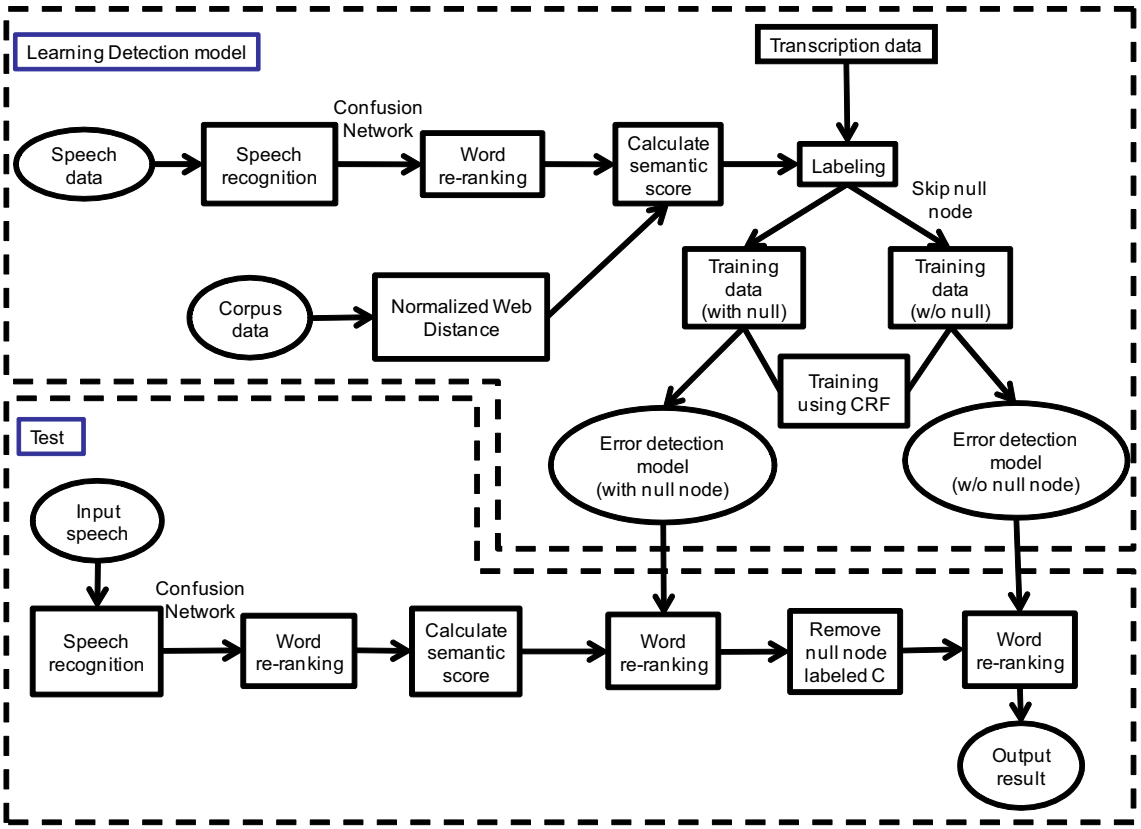

Figure 1: *Flow of proposed method*

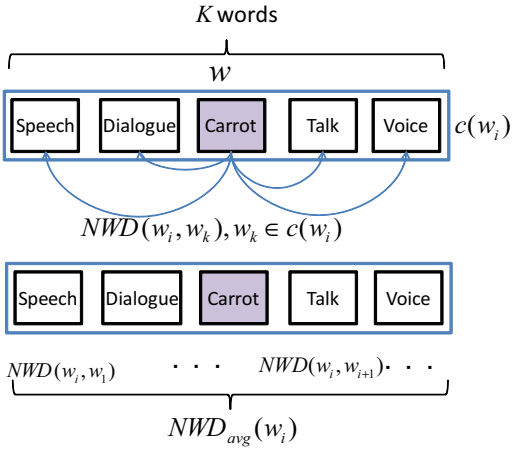

Figure 2: *Computation of semantic score*

tance solves this problem by approximating the Kolmogorov complexity using the hit numbers of search engines. We can calculate the Normalized web distance between word x and y by the equation below.

$$NWD(x,y) = \frac{\max(\log f(x), \log f(y)) - \log f(x,y)}{\log N - \min(\log f(x), \log f(y))} \quad (1)$$

Here, $f(x)$ represents the number of pages containing x, $f(y)$ represents the number of pages containing y, $f(x,y)$ represents the number of pages containing both x and y, and N is the sum of all indexed pages on the search engine.

Although the range of the NWD (with its definition) is in between 0 and ∞, in most cases, its value falls 0$\sim$1. If the two words x and y never occur together on the same web page, but do occur separately, the NWD between them is infinite. If both terms always occur together, their NWD is zero, or equivalent to the coefficient between x squared and y squared.

3.2. Algorithm

Focusing on the content words such as nouns, verbs and adjectives, we calculate the semantic score using the Normalized Web Distance equation above. For convenience, if the NWD is infinity, we calculated the semantic score by replacing it with 1. The semantic score of a recognized word w_i is calculated as follows:

(1) Context $c(w_i)$ of the content word w_i is formed as the collection of the content words around w_i not including itself as shown in Figure 2.

(2) For w_i, NWD(w_i, w_k) is calculated as the distance between each word w_k of $c(w_i)$.

(3) The average of NWD (w_i, w_k) is computed as NWD$_{avg}(w_i, w_k)$ and is allocated to w_i as its similar-

ity score.

$$\mathrm{NWD}_{avg}(w_i) = \frac{1}{K}\sum_{k} NWD(w_i, w_k) \quad (2)$$

The smaller the value of $\mathrm{NWD}_{avg}(w_i)$ is, the more the word w_i is semantically similar to the context.

4. Error correction

4.1. Conditional Random Fields

Conditional Random Fields (CRF) is one of a number of discriminative language models. CRF processes a series of data, such as sentences, and is represented as the conditional probability distribution of output labels when input data are given. The model is trained from a series of data and labels. The series of labels that the model estimates are output when test data are given. Then, labels optimizing individual data are not assigned to each data, but labels optimizing a series of data are assigned to them. In short, CRF can also learn the relationship between data.

In this paper, we use CRF to discriminate the unnatural N-gram from the natural N-gram. In short, we use CRF to detect recognition errors. This kind of discriminative language model can be trained by incorporating the speech recognition result and the corresponding correct transcription. Discriminative language models, such as CRF, can detect unnatural N-grams and correct the false word to fit the natural N-gram.

In the case of CRF, the conditional probability distribution is defined as

$$P(y \mid x) = \frac{1}{Z(x)} \exp(\textstyle\sum_a \lambda_a f_a(y, x)) \quad (3)$$

where x is a series of data and y denotes output labels. f_a denotes feature function and λ_a is the weight of f_a. Furthermore $Z(x)$ is the partition function and is defined as

$$Z(x) = \textstyle\sum_y \exp(\sum_a \lambda_a f_a(y, x)). \quad (4)$$

When training data $(x_i, y_i)(1 \leq i \leq N)$ are given, the parameter λ_a is learned in order to maximize the log-likelihood of formula (5)

$$\mathcal{L} = \textstyle\sum_{i=1}^{N} \log P(y_i \mid x_i). \quad (5)$$

L-BFGS algorithm [7] is used as a learning algorithm.

In the discrimination process, the task is to compute optimum output labels $\hat{y}$ for given input data x by using the conditional probability distribution $P(y|x)$ calculated in the learning process. $\hat{y}$ can be computed as formula (6) using the Viterbi algorithm.

$$\hat{y} = \operatorname*{argmax}_{y} P(y \mid x) \quad (6)$$

4.2. Confusion Network

The proposed system detects recognition errors by CRF, and corrects errors by replacing them with other competing hypotheses. We use a confusion network to represent competing hypotheses.

A confusion network is the compact representation of the speech recognition result. Figure 3 shows an example of a confusion network generated from the speech "Watashi tachi wa (We are)" in Japanese. The transition network enclosed by the dotted line includes the competitive word candidates with the confidence score and is called the confusion set. In this figure, four confusion sets are depicted. The null transition shown by "-" indicates there is no candidate word.

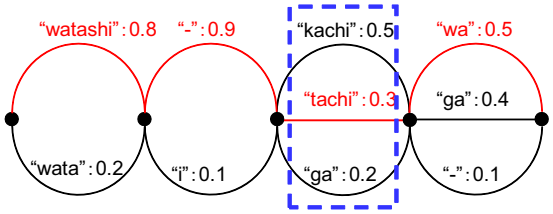

Figure 3: *An example of confusion network*

Table 1: *Number of training data and test data*

	Context model	Test
Number of lectures	150	301
Number of words	311,374	113,289

4.3. Error Correction Algorithm

In this paper, as mentioned previously, recognition errors are corrected using CRF. Word-error correction can be achieved in the confusion set by selecting the word with the highest value of the following linear discriminant function. The features that are learned are mentioned in Section 5. After the learning process is finished, recognition errors are corrected twice using the algorithm below.

First, we correct using "Error detection model with null nodes" :

(1) Convert syllable/word recognition of test data into confusion network.
(2) Extract the best likelihood words of the confusion network, and detect the recognition error using CRF.
(3) Check the confusion set in order of time series. The words identified as correct data are left unchanged. The words identified as a misrecognition are replaced with the next likelihood word in the confusion set. After that, detect recognition errors again using CRF.
(4) Select the best likelihood word in the confusion set if the word identified as correct data does not exist.
(5) Repeat processes (3) and (4) for all confusion sets in turn.
(6) Repeat processes (2) to (5) for all confusion networks in turn.

Next, we correct using "Error correction model without null nodes":

(1) Delete the null transitions that are labeled True from the first correction result and make it the test data.
(2) Repeat the process steps 2, 3, 4, 5, 6 of the above algorithm.

Using this algorithm, CRF distinguishes correct words from misrecognitions, and all the words identified as misrecognitions are corrected. Because the word bigram and trigram are used as features by CRF, the correct or misrecognized label of the word may change to the other when a preceding word is corrected. This is the reason we mentioned "in order of time series" in the algorithm (3).

Table 2: *Features used in each model*

	N-gram	Confidence score	LSA score	NWD score	Null node skip
Recognition result	×	×	×	×	×
N-gram model	◯	◯	×	×	×
LSA context model (Baseline)	◯	◯	◯	×	×
NWD context model w/ null (1)	◯	◯	×	◯	◯
NWD context model w/o null (2)	◯	◯	×	◯	×
Proposed method (1 + 2)	◯ ◯	◯ ◯	× ×	◯ ◯	◯ ×

Table 3: *Evaluation of each method*

	SUB	DEL	INS	COR	WER [%]
Recognition result	28,446	5,453	14,751	63,871	42.94
NWD context model w/o null	23,088	6,966	9,625	67,416	35.02
N-gram model	21,522	7,848	8,204	68,400	33.17
LSA context model (Baseline)	21,049	8,324	7,757	68,397	32.77
NWD context model w/ null (Yahoo)	20,469	10,130	5,316	67,171	31.70
NWD context model w/ null (CSJ)	18,073	11,524	4,597	67,873	30.18
Proposed method NWD w/ null + NWD w/o null	15,118	13,534	3,431	68,794	28.32

5. Experiment

5.1. Experiment Conditions

In order to generate the confusion network from speech data, we employed Julius-4.1.4. The acoustic model was trained using 953 lectures (male:787 lectures, female:166 lectures) from the CSJ speech database. MFCC (12 dim.) + ΔMFCC (12 dim.) + log power are used in the experiment.

The number of training and test data for the error detection model using CRF is shown in Table 1. For calculating the NWD score, we employed two types of corpora: CSJ transcript data including 2,672 lectures and Japanese Yahoo! Answers' 50 million answer datasets from April 2004 to April 2009. The context length K described in Figure 2 is set to three utterances around the current one.

5.2. Experiment Results

Table 3 shows the word error rate and evaluation with error types. "SUB", "DEL" and "INS" denote the number of substitution errors, deletion errors and insertion errors, respectively.

We carried out five experiments for comparison. The first was the general speech recognition experiment denoted as "Recognition result". The second was "N-gram model ", where word errors are corrected by using the N-gram and confusion network likelihood features. The third was the baseline "LSA context model" with the semantic score based on Latent Semantic Analysis (LSA) [8] incorporated into the "N-gram model". The fourth, the "NWD model w/ null", is a model employing Normalized Web Distance instead of LSA, and the fifth, "NWD context model w/o null", is a model that uses the same features as above, but differs because of the null transitions deleted from training data. Table 2 shows features that are used by each model. ◯ and × each denotes if the specific feature is used or not. All of the above models are trained and tested on the data shown in Table 1.

In the "Proposed method", we combine two types of detection models: First, we correct the errors by using "NWD context model w/ null". After deleting the null transitions that are labeled True from the result, we then correct the errors using "NWD context model w/o null". The results show that by employing the Normalized Web Distance, the word error rate is reduced by 1.07 points and 2.59 points compared to the LSA model, either with the corpus Yahoo! Answers and CSJ. Moreover, the substitution and insertion errors of the proposed method decreased, compared with the others. As a result, the word-error rate of the proposed method also shows the best value. Compared with the baseline, the word-error rate of the proposed method was reduced by 4.45 points from 32.77 % to 28.32 %.

6. Conclusion

In this paper, we proposed the fully automatic word error correction on the confusion network by combining the N-grams and semantic score based on Normalized Web distance. The proposed method can efficiently decrease errors, reducing the recognition errors and null transitions, which degrade the effectiveness of N-grams on the first correction, and making the further correction possible for the second run. As a result of the experiment, the proposed method achieved a 4.45-point improvement to the baseline.

7. Acknowledgements

This research is supported by Yahoo!Answers. We would like to show our gratitude to NII and Yahoo! for providing QA data.

8. References

[1] M. Goto, J. Ogata, and K. Eto, "Podcastle: A web 2.0 approach to speech recognition research," in *Proc. Interspeech2007*, 2007, pp. 2397–2400.

[2] J. Glass, T.J. Hazen, S. Cypher, I. Malioutov, D. Huynh, and R. Barzilay, "Recent progress in the mit spoken lecture processing project," in *Proc. Interspeech2007*, 2007, pp. 2553–2556.

[3] Ryohei Nakatani, Tetsuya Takiguchi, Yasuo Ariki, "Two-step correction of speech recognition errors based on n-gram and long contextual information," in *Proc. Interspeech2013*, pp. 3747–3750, 2013.

[4] Cilibrasi, R.L., P.M.B. Vitanyi, "Normalized Web Distance and Word Similarity," Handbook of Natural Language Processing, 2nd ed, pp. 293–314, 2010.

[5] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. ICML*, pp. 282–289, 2001.

[6] Lidia Mangu, Eric Brillx, Andereas Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech and Language*, vol. 14, pp. 373–400, 2000.

[7] J. Nocedal, "Updating Quasi-Newton Matrices with Limited Storage," in *Mathematics of Computation*, pp. 773–782, 1980.

[8] Thomas Landauer, Peter W. Foltz, Darrell Laham, "Introduction to Latent Semantic Analysis," in *Discourse Processing*, pp. 259–284, 1998.