LETTER

Estimation of Phone Mismatch Penalty Matrices for Two-Stage Keyword Spotting

Chang Woo HAN[†a)], **Shin Jae KANG**[†b)], ***Nonmembers***, ***and*** **Nam Soo KIM**[†c)], ***Member***

SUMMARY In this letter, we propose a novel approach to estimate three different kinds of phone mismatch penalty matrices for two-stage keyword spotting. When the output of a phone recognizer is given, detection of a specific keyword is carried out through text matching with the phone sequences provided by the specified keyword using the proposed phone mismatch penalty matrices. The penalty matrices associated with substitution, insertion and deletion errors are estimated from the training data through deliberate error generation. The proposed approach has shown a significant improvement in a Korean continuous speech recognition task.

key words: *phone mismatch penalty matrices, multistage keyword spotting, phone recognition*

1. Introduction

Currently, there is a growing interest in multistage approaches to automatic speech recognition (ASR) and keyword spotting [1]–[8]. In a multistage system, N-best phone sequences, phone lattices, or confusion networks are obtained at the first stage followed by a lexical search applying specialized decoding steps, or using more detailed information, e.g., morphological and domain-dependent knowledge.

The multistage systems are not only flexible in altering keywords but also useful to build a vocabulary-independent system. We only need to modify the lexical decoding part while the first part, the phone recognition module, is independent of the desired keywords. Furthermore, the decoding part of the system requires only a small computation compared with the conventional techniques for keyword spotting, even though the vocabulary size is huge. Since handheld devices such as PDAs, e-books, or mobile phones have low computing power and small memory size, this keyword spotting approach can be usefully implemented on such devices.

The performance of the decoding module at the second stage is mainly dependent on the phone mismatch penalties imposed to the substitution, insertion and deletion errors. There have been several studies on determining these penalties for multistage keyword spotting [2]–[6]. In [4] and [5], the penalties for substitution are decided on the basis of some rules defined over the broad acoustic-phonetic classes, and the penalties for insertion and deletion are fixed to constant values. In [6], the substitution penalties are automatically derived from the phone confusion matrix of the recognizer, while the insertion and deletion penalties are still set to fixed constants.

In this letter, we propose a new method to estimate the phone mismatch penalty matrices for two-stage keyword spotting. In the proposed approach, the phone mismatch penalties are estimated from the training data while considering all possible types of phone recognition errors. We apply a two-stage keyword spotting system based on the multi-pass phone recognition results, e.g. N-best phone sequences or phone lattices, to verify the performance of the proposed technique. The keyword spotting system using the proposed penalty matrices shows better performance than those using other penalties when evaluated on a Korean continuous speech recognition task.

2. Two-Stage Keyword Spotting

The overall block diagram of the implemented two-stage keyword spotting system is shown in Fig. 1. In our implementation, we use the mel-frequency cepstral coefficients (MFCCs) as the basic feature vectors. At the first stage, multi-pass phone recognition outputs, such as the N-best phone sequences or phone lattices, are generated by performing the conventional phone recognition based on the hidden Markov model (HMM). At the second stage, keywords are detected by comparing the generated recognition results with the hypothesized lexical phone sequences of the keywords in the lexical decoding block. The decoding operation can be regarded as a simple string match algorithm. To measure the similarity between each pair of phone strings, the phone mismatch penalties are defined to consider each type of errors. The multi-pass phone recognition outputs are decoded by applying a dynamic programming algorithm.

Let $Q^{(n)} = (q_1^{(n)}, q_2^{(n)}, \cdots, q_{N_{Q^{(n)}}}^{(n)})$ be one of the phone sequences obtained from the multi-pass phone recognizer, and

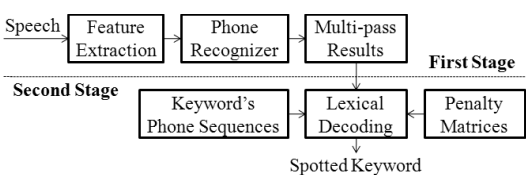

Fig. 1 Overall block diagram of the implemented two-stage keyword spotting system.

Manuscript received January 13, 2010.
Manuscript revised April 3, 2010.

[†]The authors are with the School of Electrical Engineering and the INMC, Seoul National University, Seoul 151–742, Korea.
a) E-mail: cwhan@hi.snu.ac.kr
b) E-mail: sjkang@hi.snu.ac.kr
c) E-mail: nkim@snu.ac.kr

DOI: 10.1587/transinf.E93.D.2331

$P = (p_1, p_2, \cdots, p_{N_P})$ be the lexical phone sequence of a hypothesized keyword where $N_{Q^{(n)}}$ and N_P denote the number of phones of $Q^{(n)}$ and P, respectively. Then, we can compute the sequence mismatch distance $\mathfrak{D}(P, Q^{(n)})$ by applying a dynamic programming technique similar to the dynamic time warping (DTW) algorithm [3]. Finally, keyword spotting is accomplished according to the following decision rule:

$$\min_{1\le n\le N} \mathfrak{D}(P, Q^{(n)}) \underset{H_0}{\overset{H_1}{\lessgtr}} \gamma \tag{1}$$

where N is the number of possible phone sequences from the multi-pass outputs, γ is a prespecified threshold and the two hypotheses H_1 and H_0 respectively indicate the presence and absence of the target keyword.

In order to carry out dynamic programming, we need a set of penalties that measure the degree of phone sequence mismatch. For this, we introduce three penalty matrices for the three types of phone errors: substitution, insertion and deletion. Let $\Psi = \{\phi_1, \phi_2, \cdots, \phi_{N_\phi}\}$ be the set of all phone identities, where N_ϕ is the total number of phones. The penalty matrix for each type of phone errors is a $N_\phi \times N_\phi$ matrix. Let $PM_{sub}(\phi_i, \phi_j)$ be the (i, j)th element of the penalty matrix for substitution. $PM_{sub}(\phi_i, \phi_j)$ represents a penalty imposed when the actual phone identity is ϕ_j but misrecognized as ϕ_i. In a similar manner, $PM_{ins}(\phi_i, \phi_j)$, the (i, j)th element of the penalty matrix for insertion, is defined as the penalty for the case when ϕ_i is inserted after a spoken phone ϕ_j. Finally, $PM_{del}(\phi_i, \phi_j)$, the (i, j)th component of the penalty matrix for deletion, indicates the penalty required when the spoken phone ϕ_j is missed after ϕ_i.

There are a number of string matching methods taking into consideration of substitution, insertion and deletion errors [1]–[9]. They usually implement the decoder by applying a dynamic programming algorithm to Markov chains or finite state machines. Since our purpose in this work is to propose and evaluate a new method to estimate the three different kinds of phone mismatch penalty matrices, we apply the conventional dynamic programming method similar to that used in [3].

Let $C_{i,j}^{(n)}$ be the accumulated penalty of the best path upto $(q_i^{(n)}, p_j)$. Then, it is updated as follows:

$$C_{i,j}^{(n)} = \begin{cases} PM_{sub}(q_i^{(n)}, p_j), & i = j = 1 \\ C_{i,j-1}^{(n)} + PM_{del}(p_{j-1}, p_j), & i = 1, j \neq 1 \\ \min\left[PM_{sub}(q_i^{(n)}, p_j), \right. & \\ \quad \left. C_{i-1,j}^{(n)} + PM_{ins}(q_i^{(n)}, p_j)\right], & i \neq 1, j = 1 \\ \min\left[C_{i-1,j-1}^{(n)} + PM_{sub}(q_i^{(n)}, p_j), \right. & \\ \quad C_{i-1,j}^{(n)} + PM_{ins}(q_i^{(n)}, p_j), & \\ \quad \left. C_{i,j-1}^{(n)} + PM_{del}(p_{i-1}, p_j)\right], & \text{otherwise,} \end{cases} \tag{2}$$

in which $1 \le i \le N_{Q^{(n)}}$ and $1 \le j \le N_P$.

Since we do not know the exact starting point of the hypothesized keyword, we use $(\min[PM_{sub}(q_i^{(n)}, p_j), C_{i-1,j}^{(n)} + PM_{ins}(q_i^{(n)}, p_j)])$ when $i \neq 1$ and $j = 1$ instead of $(C_{i-1,j}^{(n)} + PM_{ins}(q_i^{(n)}, p_j))$. This modification enables us to spot the hypothesized keyword regardless of the exact starting time.

After calculating $C_{i,j}^{(n)}$ for all the possible (i, j) grids, $\mathfrak{D}(P, Q^{(n)})$ is obtained as follows:

$$\mathfrak{D}(P, Q^{(n)}) = \min_{1\le i\le N_{Q^{(n)}}} (C_{i,N_P}^{(n)} / l_{i,N_P}^{(n)}) \tag{3}$$

where $l_{i,N_P}^{(n)}$ is the length of the best path upto $(q_i^{(n)}, p_{N_P})$ which is available through the backtracking technique.

3. Phone Mismatch Penalty Matrices

It has been reported that the accuracies of the state-of-the-art HMM-based phone recognizers are around 70% [10], [11]. The performance deteriorates in the presence of background noise. In many cases, the phone sequences of spoken keywords may not be found in the phone recognition results because there exist frequent occurrences of the three type of errors: substitution, insertion and deletion. For that reason, an appropriate penalty to each type of error should be taken into consideration.

A simple way may be to assign the same penalty to all the possible errors. This is called the Levenshtein metric which counts the number of corrections required for converting a sequence to the target sequence. In a number of preliminary experiments, we could observe that some specific error patterns occur more frequently than others. In that case, for a better performance, it is desired to assign different penalty for each error pattern depending on its possibility of occurrence. One of the successful previous approaches is the phone confusion matrix, in which the penalties are estimated based on the recognition error patterns obtained from the training data [6]. However, the amount of recognition errors observed in the training data is usually considered to be insufficient to reflect all the possible phone error patterns. Here we propose a novel technique to determine each penalty matrix from a set of training data.

Let $X = (x_1, x_2, \cdots, x_{N_P})$ be an acoustic feature vector sequence corresponding to a spoken phone sequence $P = (p_1, p_2, \cdots, p_{N_P})$, in which N_P is the number of spoken phones and x_i represents the feature vector segment associated with the i-th phone, p_i. Given P, X can be segmented into each phone region $\{x_i\}$ by applying forced alignment such as the Viterbi decoding approach. Once X is segmented into $(x_1, x_2, \cdots, x_{N_P})$, the log-likelihood for P can be factorized as follows:

$$\log \mathcal{P}(X|P) = \sum_{k=1}^{N_P} \log \mathcal{P}(x_k|p_k) \tag{4}$$

where $\mathcal{P}(\cdot)$ denotes the likelihood computed in the HMM framework.

For a good performance of string match, it is desirable to assign a heavy penalty to the error type that occurs rarely

and light penalties to frequent error patterns. To estimate the relative frequency of each error pattern, we deliberately substitute or delete the spoken phones and insert non-spoken phones so as to create intended phone error patterns. The penalty matrix for substitution is computed as follows:

$$PM_{sub}(\phi_i, \phi_j) = \begin{cases} -\log \Pr\left[\mathcal{P}(x_k|\phi_j) < \mathcal{P}(x_k|\phi_i) \middle| p_k = \phi_j\right] + \alpha_{sub}, \phi_i \neq \phi_j \\ 0, \qquad \phi_i = \phi_j \end{cases} \tag{5}$$

where $\Pr[\cdot]$ denotes the probability of the enclosed event and α_{sub} is a non-negative control parameter which is empirically determined depending on the phone insertion/deletion rates. In (5), it is noted that we make a deliberate substitution of the spoken phone ϕ_j with another phone ϕ_i and if the likelihood of the substituted phone becomes larger, then we treat the case as a possible phone substitution error. Training of $PM_{sub}(\phi_i, \phi_j)$ based on the training data X is achieved by using

$$\Pr\left[\mathcal{P}(x_k|\phi_j) < \mathcal{P}(x_k|\phi_i) \middle| p_k = \phi_j\right] \simeq \frac{\sum_{k=1}^{N_P} \mathbb{I}[p_k = \phi_j, \mathcal{P}(x_k|\phi_j) < \mathcal{P}(x_k|\phi_i)]}{\sum_{k=1}^{N_P} \mathbb{I}[p_k = \phi_j]} \tag{6}$$

with $\mathbb{I}[a]$ denoting the indicator function which equals 1 when the condition a is satisfied and 0 otherwise.

In a similar manner, $PM_{ins}(\phi_i, \phi_j)$ is obtained by

$$PM_{ins}(\phi_i, \phi_j) = -\log \Pr\left[\mathcal{P}(x_k|\phi_j) < \mathcal{P}(x_k|\phi_j, \phi_i) \middle| p_k = \phi_j\right] + \alpha_{ins} \tag{7}$$

with

$$\Pr\left[\mathcal{P}(x_k|\phi_j) < \mathcal{P}(x_k|\phi_j, \phi_i) \middle| p_k = \phi_j\right] \simeq \frac{\sum_{k=1}^{N_P} \mathbb{I}[p_k = \phi_j, \mathcal{P}(x_k|\phi_j) < \mathcal{P}(x_k|\phi_j, \phi_i)]}{\sum_{k=1}^{N_P} \mathbb{I}[p_k = \phi_j]}. \tag{8}$$

In (7) and (8), we replace the original spoken phone ϕ_j by the concatenated phones (ϕ_i, ϕ_j), and α_{ins} is an experimentally determined control parameter. The likelihood $\mathcal{P}(x_k|\phi_j, \phi_i)$ can be easily calculated by constructing an HMM concatenating two phone models for ϕ_j and ϕ_i.

Finally, the penalty matrix for deletion is estimated as follows:

$$PM_{del}(\phi_i, \phi_j) = -\log \Pr\left[\mathcal{P}(x_{k-1}, x_k|\phi_i, \phi_j) < \mathcal{P}(x_{k-1}, x_k|\phi_i) \middle| p_{k-1} = \phi_i, p_k = \phi_j\right] + \alpha_{del} \tag{9}$$

with

$$\Pr\left[\mathcal{P}(x_{k-1}, x_k|\phi_i, \phi_j) < \mathcal{P}(x_{k-1}, x_k|\phi_i) \middle| p_{k-1} = \phi_i, p_k = \phi_j\right] \simeq \frac{\sum_{k=2}^{N_P} \mathbb{I}[p_{k-1} = \phi_i, p_k = \phi_j, \mathcal{P}(x_{k-1}, x_k|\phi_i, \phi_j) < \mathcal{P}(x_{k-1}, x_k|\phi_i)]}{\sum_{k=2}^{N_P} \mathbb{I}[p_{k-1} = \phi_i, p_k = \phi_j]} \tag{10}$$

where (x_{k-1}, x_k) is the concatenation of the two feature vector segments, x_{k-1} and x_k, and α_{del} is an empirically determined control parameter.

The proposed technique is similar to the phone confusion matrix particularly for the substitution error. Phone confusion matrix is derived from the recognition errors observed in the training data [6]. In contrast, the proposed method deliberately creates all the possible error patterns, which will be helpful for robust penalty estimation. Furthermore, more sophisticated treatment of the insertion and deletion errors is achieved compared to the phone confusion matrix technique.

4. Experimental Results

Performance of a keyword spotting algorithm with the proposed penalty matrices was evaluated on the Korean continuous speech Reading Sentence DB collected at Speech Information Technology & Industry Promotion Center (SiTEC) [12]. The SiTEC Reading Sentence DB contains 20,217 sentences consisting of about 30,000 different word tokens. It was collected by recording the speech from 200 male and 200 female speakers. The number of keywords was 1000.

The database was divided into three sets: training, development and testing sets. The training set was used for the estimation of HMM parameters, the development set was used to train the phone mismatch penalty matrices, and the testing set was used for the performance evaluation. A detailed information of each set of the DB is shown in Table 1.

In our keyword spotting system, the HMM-based phone recognizer was applied at the first stage. Context-dependent triphone models were used to construct this phone recognizer. The number of states for each phone was three, and the number of Gaussian mixtures for each HMM state was eight. As the output of the phone recognition, we generated N-best phone sequences or phone lattices where N varied from 10 to 100. The phone recognition accuracy calculated based on the 1-best phone sequences was 69.30%. The phone mismatch penalty matrices were estimated on the development set. To train the penalty matrices, we applied monophone HMMs instead of the triphone models used in the phone recognizer. The control parameters α_{sub}, α_{ins} and α_{del} were set to 0.5, 0.5 and 1.5, respectively, which showed a good performance in our experiments.

As reference systems with which we compared the performance, we also implemented two other keyword spotting systems which were similar to our approach but employed different penalties [6]. The first one employed the

Table 1 Database used in this study.

	Training	Development	Testing	Total
Sentences	30,399	6,904	3,454	40,757
Speakers (Male/Female)	298 (149/149)	68 (34/34)	34 (17/17)	400 (200/200)
Duration (hh:mm:ss)	64:08:36	13:35:08	07:12:47	84:56:31

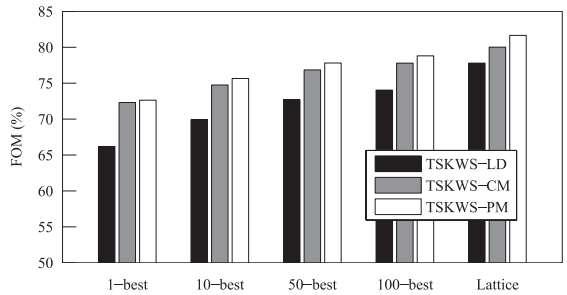

Fig. 2 FOMs for TSKWS-LD, TSKWS-CM and TSKWS-PM.

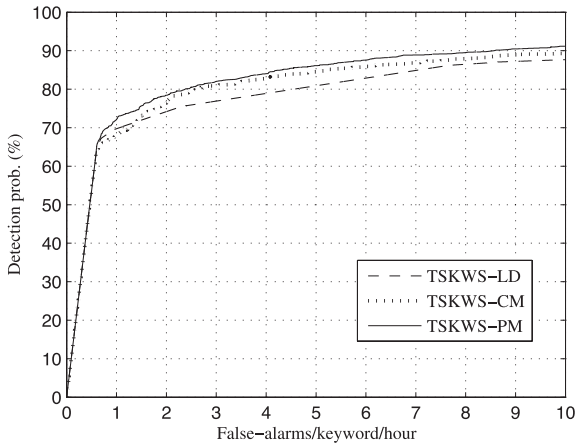

Fig. 3 ROC curves for TSKWS-LD, TSKWS-CM and TSKWS-PM (phone lattice).

Levenshtein distance as the penalties. Let $PM'_{sub}(\phi_i, \phi_j)$, $PM'_{ins}(\phi_i, \phi_j)$ and $PM'_{del}(\phi_i, \phi_j)$ respectively denote the (i, j)th penalty value for substitution, insertion and deletion defined by the Levenshtein distance metric. Then,

$$PM'_{sub}(\phi_i, \phi_j) = \begin{cases} 1, & \phi_i \neq \phi_j \\ 0, & \phi_i = \phi_j \end{cases}$$
$$PM'_{ins}(\phi_i, \phi_j) = PM'_{del}(\phi_i, \phi_j) = 1. \quad (11)$$

In the second system, the substitution penalties were calculated by applying the phone confusion matrix [6] technique with the insertion and deletion penalties being set to 3.5, which showed the best performance in our experiments.

The performances of the proposed and the reference keyword spotting systems were compared in terms of figure of merit (FOM) and receiver operating characteristic (ROC) curves. For convenience, we denote the two-stage approach with Levenshtein metric by TSKWS-LD, with phone confusion matrix by TSKWS-CM, and with the proposed penalty matrices by TSKWS-PM. First, we compared the FOMs of the three approaches. The FOM implies the average detection probability when the number of false alarms per keyword per hour is kept between 0 and 10. Figure 2 shows the resulted FOMs from which we can see that the proposed TSKWS-PM technique significantly outperformed the other approaches. The average relative improvement in FOM of the proposed technique was 18.48% compared to TSKWS-LD and 4.31% compared to TSKWS-CM. Next, the ROC curves for these three systems were obtained as shown in Fig. 3 when the first stage output is given by a phone lattice. The detection probability of the proposed algorithm was higher than that of the other approaches over the whole range of false alarm. From the results, it can be concluded that the proposed approach produced better detection performance compared with the conventional approaches.

5. Conclusions

In this letter, we have presented a new technique to estimate the phone mismatch penalty matrices for two-stage keyword spotting. Proposed penalty matrices are utilized to measure the similarity between recognized phone sequences and phone sequences of the hypothesized keywords. The penalties corresponding to all types of errors are estimated from the training data. When N-best phone sequences or phone lattices are given as the outputs of the first stage, detection of a specific keyword is carried out through dynamic programming based on the penalty matrices. From a number of experiments on the Korean continuous speech recognition task, it has been shown that the proposed approach outperformed the conventional techniques.

Acknowledgments

This work was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2009-0083044) and by the Seoul R&BD Program (10544).

References

[1] S. Dharanipragada and S. Roukos, "A multistage algorithm for spotting new words in speech," IEEE Trans. Speech, Audio Process., vol.10, no.8, pp.542–550, Nov. 2002.
[2] D.A. James and S.J. Young, "A fast lattice-based approach to vocabulary independent wordspotting," Proc. ICASSP, pp.899–902, April 1997.
[3] K.M. Knill and S.J. Young, "Low-cost implementation of open set keyword spotting," Comput. Speech Lang., pp.243–266, July 1999.
[4] K. Thambiratnam and S. Sridharan, "Dynamic match phone lattice searches for very fast and accurate keyword spotting," Proc. ICASSP, pp.465–468, March 2005.
[5] L. ten Bosch, A. Hamalainen, O. Scharenborg, and L. Boves, "Acoustic scores and symbolic mismatch penalties in phone lattices," Proc. ICASSP, pp.437–440, May 2006.
[6] K. Audhkhasi and A. Verma, "Keyword search using modified minimum edit distance measure," Proc. ICASSP, pp.929–932, April 2007.
[7] L.R. Bahl and F. Jelinek, "Decoding for channels with insertions, deletions, and substitutions with applications to speech recognition," IEEE Trans. Inf. Theory, vol.IT-21, no.44, pp.404–411, July 1975.
[8] S. Nakagawa and M.M. Jilan, "Syllable-based connected spoken word recognition by two pass O(n) DP matching and hidden Markov models," Proc. ICASSP, pp.1117–1120, April 1986.
[9] D. Sankoff and J.B. Kruskal, Time Warps, String Edits, and Macromollecules: The Theory and Practice of Sequence Comparison, Addison Wesley, 1983.

[10] Q. Fu, X. He, and L. Deng, "Phone-discriminating minimum classification error (P-MCE) training for phonetic recognition," IEEE Trans. Speech, Audio Process., vol.10, no.8, pp.542–550, Nov. 2002.

[11] F. Sha and L.K. Saul, "Comparison of large margin training to other discriminative methods for phonetic recognition by hidden Markov models," Proc. ICASSP, pp.313–316, May 2007.

[12] SiTEC website. [Online]. Available: http://www.sitec.or.kr/