Spoken Term Detection by N-gram Index with Exact Distance for NTCIR-SpokenDoc2

Nagisa Sakamoto
Toyohashi University of Technology
1-1 Hibarigaoka
Toyohashi-shi
Aichi,440-8580
sakamoto@slp.cs.tut.ac.jp

Seiichi Nakagawa
Toyohashi University of Technology
1-1 Hibarigaoka
Toyohashi-shi
Aichi,440-8580
nakagawa@slp.cs.tut.ac.jp

ABSTRACT

For spoken term detection, it is very important to consider Out-of-Vocabulary (OOV). Therefore, sub-word unit based recognition and retrieval methods have been proposed. This paper describes a very fast Japanese spoken term detection system that is robust for considering OOV words. We used individual syllables as sub-word unit in continuous speech recognition and an n-gram index of syllables in a recognized syllable-based lattice. We propose an n-gram indexing/retrieval method in the syllable lattice for attacking OOV, and high speed retrieval. Specially, in this paper, we applied our method to SDPWSspeech and reported the evaluation results.

Keywords

spoken term retrieval, syllable recognition, n-gram, dummy syllable

1. INTRODUCTION

The wide availability on the web of such multimedia data as audio continues to grow. Information can be found using an existing textual search engine if the target data are comprised of such textual information as transcriptions of broadcast news or newspapers. However, efficient robust spoken document retrieval (SDR) or spoken term detection (STD) methods have not yet to be established, since system designers face specific problems, such as recognition errors and out-of-vocabulary(OOV) terms. The SDR task, which seeks suitable documents or passages based on the query, is usually performed using STD results. The aim of this research is to develop a robust and efficient STD method.

For retrieving speech-based documents, some problems to be solved remain, such as OOV and recognition errors. In German, the retrieval method based on the weighted Levenshtein distance between syllables (words consist of only one syllable in a ratio of half)[1] has been proposed. In Chinese, syllable-unit (440 syllables in total) has often been used as a basic unit of recognition/retrieval[2]. Japanese consists of only about 110 syllables, therefore the syllable unit is suitable for the spoken retrieval of OOV words. In addition, other retrieval methods based on elastic matching between two syllable sequences have been tried for considering recognition errors[3]. Phoneme based n-gram has been proposed for various retrieval methods, usually with bag of words or partial exact matching[4, 5]. For document retrieval, Chen et al[6] used skipped (distant) bigrams such as s_1-s_3, s_2-s_4 for the syllable sequence of $s_1 s_2 s_3 s_4$. Phoneme recognition errors such as substitution errors have not been explicitly considered for OOV term retrieval[7, 8].

Typically, as with the dynamic time warping (DTW) method, a string is used to elastically match candidates for pruning. Katsurada et al. proposed a fast DTW matching method based on suffix array[9]. Kanda et al. [10] proposed a hierarchical DTW matching method between phoneme sequences, where a coarse matching process is followed by fine matching. However, their method still consumes a great deal of computation time and memory storage. Recentry, Saito et al.[11] also proposed a coarse/fast retrieval method based on tri-gram matching results which were calculated in advance, and it is followed by the fine DTW matching. This method consumes huge compatation in advance.

For Fast/Robust STD, we used the n-gram index with distance measure that acounts three kinds of recognition errors in the syllable recognition lattice[12, 13]. First, to handle substitution errors, we use a trigram array that considers the m-best and dummy syllables in the syllable lattice. Second, to tackle insertion errors, we create an n-gram array that permits a one-distant n-gram. Finally, to address deletion errors, we search for edited queries from which one syllable has been deleted.

In this paper, especially, we propose the technique of combination of trigram index, bigram index and unigram index and we will show its availability for SDPWSspeech documents.

The remainder of this paper is organized as follows. In Section 2 and 3, we describe our retrieval system and, in Section 4, present the DTW for baseline. Evaluation results are given in Section 5 and a conclusion in Section 6.

2. SYSTEM OVERVIEW

The n-gram information of syllables is maintained in a data structure called an n-gram array that consists of index and syllable distance information for each n-gram. Figure 1 shows how a trigram array is arranged. First, the appearance positions of the syllables in a recognized syllable lattice for a spoken document are located. Then an n-gram of the syllable is constructed at every appearance position. Next, the n-gram is sorted in lexical order so that it can be searched for quickly using a binary search algorithm. In previons studies, we used only trigram arry[12, 13]. In this paper, we proposed the new method of using trigram, bi-

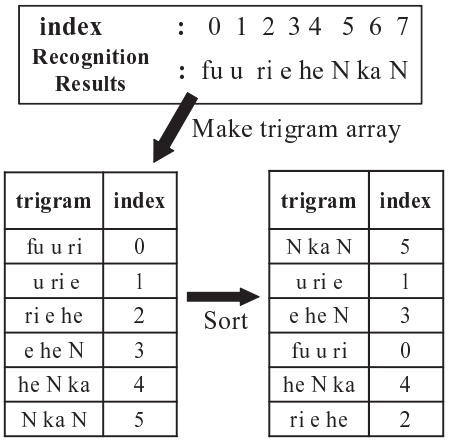

Figure 1: n-gram array indexing procedure (n=3) To simplify, the recognition result is represented by only the first candidate (1-best)

gram and unigram array.

The search process for an n-gram array includes three steps. First, a query is converted into a syllable sequence. Second, an n-gram of the query is constructed. Finally, the n-gram in a query is retrieved from the n-gram array. A query consisting of more than 4 syllables is retrieved using a combination of n-grams. A query consisting of less than 6 syllables but more than 4 syllables is separated into traigram and bigram or unigram for the first and second halves. Thus, the query is retrieved from the trigram array and bigram array or unigram array. The retrieved results are merged by considering whether the position at which the detection result occurred in the first and second halves is the same. Similarly, a query with less than 9 syllables but more than 7 syllables is retrieved by a sequence of syllables by dividing the query into three parts (Fig. 2). For example, when a query consists of six syllables, "i mi ka i se ki" in Fig. 2, the query's syllable sequence is divided into two trigrams; "i mi ka" and "i se ki." If the first term, "i mi ka," is detected at $s_1 \sim t_1$ with a distance less than a thereshold, that is, index position $= s_1$, and the second term, "i se ki," is detected at $t_1 + 1 \sim u_1$ with a distance less than a threshold, that is, index position $= t_1 + 1$, then "i mi ka i se ki" is detected at $s_1 \sim u_1$. For a query consisting of five syllables, "ke i ta i so" in Fig. 2, the query sequence is divided into a trigram and a bigram; "ke i ta" and "i so". If the first term "ke i ta" is detected at $s_2 \sim t_2$ and the second term "i so" is detected at $t_2 + 1 \sim u_2$, then "ke i ta i so" is detected at $s_2 \sim u_2$.

The query term is detected, if the following distance is lower than a pre-determined threshold. Strictly speaking, the threshold depends on the query length.

$$\frac{\alpha \times \sum d_S + \beta \times \sum d_I + \gamma \times \sum d_D}{number\ of\ syllable} \quad (1)$$

,when d_S, d_I and d_D denotes the distans for substitution, insertion, and delition errors, respectively.

3. SOLVING MIS-RECOGNIZED SUB-WORD PROBLEM

Figure 3 illustrates how to construct trigram array indexies for taking into consideration of substitution errors and

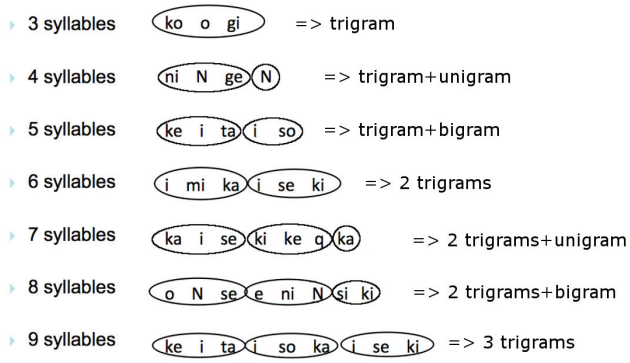

Figure 2: Example of query division into trigram

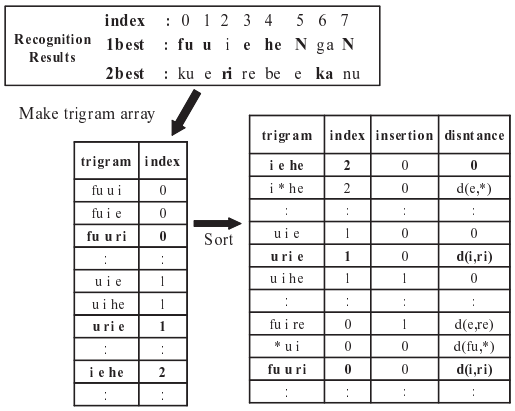

Figure 3: Procedure for making trigram array

insertion errors.

3.1 Substitution error

To handle substitutions errors, we use an n-gram array constructed from the m-best of the syllable lattice[12]. An n-gram array is constructed by using the combination of syllables in the m-best syllable lattice. Thus, for one position in the lattice, there are m^n kinds of n-gram. For example, even if the recognition result of the 1-best is "fu u i e he N ga N" having recognition errors, we can search the query "fu u ri e he N ka N("Fourie Transform" in English)", if a correct syllable is included in the m-best. We used HMM based Bhattacharrya distance [12] as the local distance between the 1st candidate and other candidate. The "fu u ri" distance is calculated as distance between "fu u ri" of target trigram and "fu u i" of the 1-best trigram, where the distance is $d_s(ri, i)$.

Even if we use the syllable lattices, some substitution errors are not contained in the lattice. Therefore, we introduce the dummy syllable symbol or "wild card". A dummy syllable is represented by "*" . The dummy syllable can match with any syllable that is not contained in the m-best recognition results. For example, if the recognition result of the m-best does not include "C", the original method can not search the query "ABCD". At this case, the query using the dummy syllable has n-gram as AB*, A*C and *BC, and we can retrieve the query "ABCD". Therefore, the recall rate is increased. On the other hand, the method has the potential-

ity to decrease the precision rate. This problem is addressed by increasing the distance between "*" and any other syllable, where only one dummy syllable is allowed in a trigram. We should notice that this approach is different from a one distant bigram index method. We used the exact definition of $d_S(e,*)$ as $d_S(syllable\ of\ query, e) + \delta_*$ after finding the index, in other words, instead of a constant value as follows:

$$d_S(*,*) = \lambda \times d_S(\quad syllable\ of\ query, \quad best\ syllable\ for\ the\ dummy\ syllable \quad) + \eta \tag{2}$$

,where λ and η denotes an penalty for using the dummy syllable. For example, if "query" is "i me he", the distance between "me" in the query and "*" in the lattice is defined as $\lambda \times d_s(me, e) + \eta$.

3.2 Insertion error

To address the insertion errors, we make an n-gram array that permits a one-distant n-gram. Considering the gap between appearance positions deals with the error. Even if the recognition result is "fu ku u ri e he N ka N" having an insertion error "ku", we can search for the query "fu u ri e he N ka N", if the n-gram array that considers a one-distant n-gram is allowed. Therefore, it is possible to deal with one insertion error within every n-gram. The trigram of "fu u ri" is constructed as a skipped trigram from "fu ku u ri", when "ku" is regarded as an insertion error.

The insertion distance is defined as follows:

$$d_I(C_2V_2|C_1V_1\_C_3V_3) = \min \left\{ \begin{array}{l} d_S(C_1V_1, C_2V_2) \\ d_S(V_1, C_2V_2) \\ d_S(C_2V_2, C_3V_3) \end{array} \right\} + \delta_I \tag{3}$$

where C_2V_2(C=consonant, V=vowel) denotes the insertion syllable, and C_1V_1 and C_3V_3 denote the left context and right context, respectively. "δ_I" denotes an insertion penalty. "$d_S(V_1, C_2V_2)$" means that "a part of vowel V_1" is mis-separated into the vowel and an inserted syllable.

3.3 Deletion error

To handle the deletion errors, we search the query as above while allowing for the case where one syllable in the query is deleted.

Even if the recognition result is "fu u e he N ka N" having a deletion error, we can search the query "fu u ri e he N ka N", if a syllable ('ri') in the query is deleted.

When a query consisting of syllables more than 2n must consider deletions of two syllables, the errors for a long query can not be corrected simply by deleting one syllable. In such a case, the query is divided two parts, and they are made to drop out by one syllable, and retrieved. For example, for the recognition result of "fu ri e he N ka", it is retrieved by considering one deletion of "fu u ri e" and of "he N ka N" in the case of $n = 3$, respectively.

The deletion distance in a query is defined as follows:

$$d_D(C_2V_2|C_1V_1\_C_3V_3) = \min \left\{ \begin{array}{l} d_S(C_1V_1, C_2V_2) \\ d_S(V_1, C_2V_2) \\ d_S(C_2V_2, C_3V_3) \end{array} \right\} + \delta_D \tag{4}$$

4. DTW -BASE LINE METHOD-

Substitution, insertion and deletion errors occur in subword based automatic speech recognizers. The word detector has to find the candidate positions from the recognized sub-word sequence. We usually call this word spotting. We use a syllable as the sub-word unit, where each Japanese syllable consists of a consonant and a vowel, or a single vowel. Each word is expressed as a concatenation of syllables. For word spotting, the recurrence equation for DTW (between the sub-word sequence $A = a_1 a_2 \cdots a_I$ for a document and a query sub-word sequence $B = b_1 b_2 \cdots b_J$) is given below [12, 13]. Let $g(i,j)$ be the intermediate matching distance, and B(i,j) be the starting frame of the matching, that is, $b_1 b_2 \cdots b_j$ is matched with $a_s a_{s+1} \cdots a_i$, where $s = B(i,j)$.

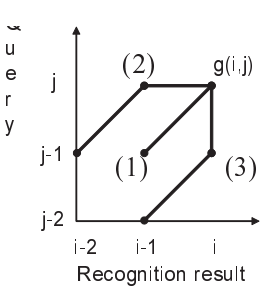

Figure 4: Constrained conditions for DTW (1): substitution (2): insertion (3): deletion

$$g(i,j) = \min \left\{ \begin{array}{l} (1) g(i-1, j-1) + w_1 d(i,j) \\ (2) g(i-2, j-1) + w_{21} d(i-1,j) + w_{22} d(i,j) \\ (3) g(i-1, j-2) + w_{31} d(i, j-1) + w_{32} d(i,j) \end{array} \right. \tag{5}$$

$$B(i,j) = \left\{ \begin{array}{l} (1) B(i-1, j-1) \\ (2) B(i-2, j-1) \\ (3) B(i-1, j-2) \end{array} \right. \tag{6}$$

When the syllable recognition result contains plural candidates, the local distance between i-th syllable in a query and k-th candidate (j_k) for j-th syllable position in a recognition result (lattice) is defined as follows:

$$g(i,j) = \arg \min_k d(i, j_k) + \delta_k \tag{7}$$

In each expression, its label of (1)-(3) corresponds to the restricted path condition for DTW in Fig. 4, that is,

(1) query matches the recognition result or substitution error

(2) insertion error

(3) deletion error

We calculate the distance between a query and the retrieval result using the edit or Bhattacharyya distance between syllables as local distance "d". In the following experiment, we set all weights of {w_1-w_{32}} to 1.0.

Table 1: Recognition results (%)

output	Del	Ins	Subs	Corr	Acc
Syllable (1best)	8.5	3.2	16.2	75.3	72.1
Syllable (3best)	8.5	3.3	8.5	82.9	79.7
Syllable (5best)	8.5	3.2	6.2	85.3	82.1

Table 2: Formal results of retrieval (5 best)

	DTW	3gram	Ngram
Recall	0.302	0.213	0.269
Precision	0.445	0.577	0.431
F-measure	0.360	0.309	0.331
MAP	0.514	0.278	0.380
sec/query	1.70	0.12	0.14

5. EVALUATION

5.1 Experimental setup

We used the SDPWSspeech data for search target document and recognized it by SPOJUS++[14] developed in our laboratory. The context-dependent syllable-based HMMs (928 models in total) were trained on 2707 lectures within the CSJ corpus. We used a left-to-right HMM, consisting of four states with self loops, and has four Gaussians with full covariance matrices per state. We used syllable-based four grams as a language model, which was traind by the CSJ corpus excluding the core data.

We used the query set for the formal run in NTCIR10 (♯unknown query terms = 100, ♯known query terms = 100). The syllable recognition rates are summarized in Table 1. We implemented the proposed method on the following machine specification: Xeon 2.93GHz, 24core CPU, and 74GB memory (we used only a single core). The performance of syllable recognition resluts was worse than those of CSJ lecture corpus, especisslly, there were many deletion errors. In our propsed method, when the syllable recognition performance is poor, the retrieval performancs is strongly affected by errors in compared with the DTW method, because the correct n-gram indexes were not contained in the indexes. We compared the following three methods;

1. baseline based on DTW (NKGW-1 in formal run)
2. trigram index (NKGW-3)
3. Ngram index (trigram, bigram, unigram; NKGW-2)

5.2 Experimental result

Table 2 and Fig.5 show the results (with the maximum of F-measure and MAP) submitted as a formalrun task on NTCIR10, where the "DTW" represents the performance using the our baseline method DTW, and "3gram" represents the performance using the only 3-gram array[15], and "Ngram" represents the performance using the 3-gram, 2-gram and 1-gram array[16]. We used 5 best syllable recognition results for these methods.

From the view point of retrieval performance, "Ngram" result was better than "3gram" result. However our baseline method(DTW) result outperfomed our proposed method (Ngram) result in both of F-measure and MAP. That is due to use an independent threshold on the length of the query. In our previous studies[16], we changed the threshold value by the length of the query for only the n-gram method. Therefore we evaluated retrieval results using dependent thresholds on query length for DTW, 3gram and Ngram methods. In addition, we also tried the retrieval using only 3 best recognition results. Table 3 and Fig.6 show the results of retrieval using the variable thresholds. By introducing various thresholds on the query length, we improved the results, but it was still worse than the baseline (DTW). In the using case of only 3 best candidates in the lattice, the index size was reduced to 1.2GB from 3.6GB in the comparison with the 5 best, while keeping the retrieval performance. In this case, the retrieval time was 30 times faster than the DTW.

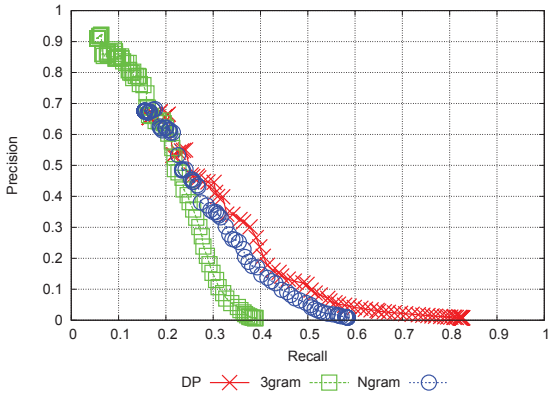

Figure 5: Formal retrieval results

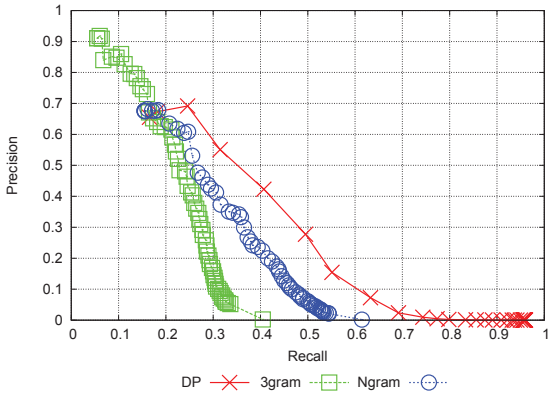

Figure 6: Retrieval results using variable thresholds on query length (5best)

6. CONCLUSION

In this paper, we proposed a spoken term detection method by n-gram (trigram, bigram, unigram) index with exact distance of substitution, insertion and deletion errors. A query term is divided some trigrams and bigram or unigram. This method was applied to NTCIR10-SpokenDoc2;SDPWSspeech.

Table 3: Retrieval results by using dependent thresholds on query length

	DTW(5best)	3gram(5best)	Ngram(5best)	DTW(3best)	3gram(3best)	Ngram(3best)	DTW(1best)
Recall	0.407	0.241	0.247	0.406	0.227	0.242	0.331
Precision	0.423	0.481	0.608	0.422	0.544	0.600	0.487
F-measure	0.415	0.321	0.351	0.414	0.320	0.343	0.394
MAP	0.518	0.281	0.381	0.512	0.249	0.353	0.452
sec/query	1.70	0.12	0.14	1.52	0.03	0.05	1.35
index size[GB]	-	3.1	3.6	-	0.8	1.2	-

7. REFERENCES

[1] M. Larson and S. Eickeler, "Using syllable-based indexing features and language models to improve German spoken document retrieval," EuroSpeech, 2003, pp. 1217–1220.

[2] H. Wang, "Experiments in syllable-based retrieval of broadcast news speech in Mandarin Chinese," Speech Communication, 2000, vol. 32, pp. 49–60.

[3] M. Wechsler, E. Munteanu, and P. Schauble, "New techniques for open-vocabulary spoken document retrieval," SIGIR, 2008, pp. 20–27.

[4] C. Allauzen, M. Mohri, and Saracla M, "General indexation of weighted automata - application to spoken utterance retrieval," Workshop on interdisciplinary approaches to speech indexing and retrieval, 2004, pp. 33–40.

[5] M. Saraclar and R. Sproat, "Lattice-based search for spoken utterance retrieval," HLT/NAACL, 2004, pp. 129–136.

[6] B. Chen, H. Wang, and L. Lee, "Retrieval of broadcast news speech in Mandarin Chinese collected in Taiwan using syllable-level statistical characteristics," ICASSP, 2000, pp. 2985–2988.

[7] C. Ng, R. Wilkinson, and J. Zobel, "Experiments in spoken document retrieval using phoneme n-grams," Speech Communication, 2000, vol. 32, pp. 61–77.

[8] S. Dharanipragada and S. Roukos, "A multistage algorithm for spotting new words in speech," IEEE Transactions on Speech and Audio Processing, 2002, vol. 10, pp. 542–550.

[9] K. Katsurada, S. Teshima, and T. Nitta, "Fast keyword detection using suffix array," Interspeech, 2009, pp. 2147–2150.

[10] N. Kanda, H. Sagawa, T. Sumiyoshi, and Y. Obuchi, "Open-vocabulary keyword detection from super-large scale speech database," MMSP, 2008, pp. 939–944.

[11] H. Saito, Y.Itoh, K.Kojima, and M.Ishigame et al., "Examination of the index in method of the n-syllable sequences in advance," ASJ2013 Spring Meeting, 2013 (in Japanese).

[12] K. Iwami, Y. Fujii, K. Yamamoto, and S. Nakagawa, "Out-of-vocabulary term detection by n-gram array with distance from continuous syllable recognition results," SLT, 2010, pp. 200–205.

[13] K. Iwami, Y. Fujii, K. Yamamoto, and S. Nakagawa, "Efficient out-of-vocabulary term detection by n-gram array indices with distance from a syllable lattice," ICASSP 2011, 2011, pp. 5664–5667.

[14] Y. Fujii, K. Yamamoto, and S. Nakagawa, "Large vocabulary speech recognition system: Spojus++," MUSP, 2011, pp. 110–118.

[15] S. Nakagawa, K. Imami, Y. Fujii, and K. Yamamoto, "A robust/fast spoken term detection method based on a syllable n-gram index with a distance metric," Speech Communication 2012, 2012.

[16] N. Sakamoto, K. Yamamoto, and S. Nakagawa, "Evaluation of spoken term detection method using syllable trigram with distance through voice input," ASJ2013 Spring Meeting, 2013 (in Japanese).