Proceedings of 20th International Congress on Acoustics, ICA 2010

23-27 August 2010, Sydney, Australia

Fast subword-based approach for open vocabulary spoken term detection

Shi-wook Lee (1), Yoshiki Nambu (2), Hiroaki Kojima(1)

Kazuyo Tanaka (1,2) and Yoshiaki Itoh(3)

(1) National Institute of Advanced Industrial Science and Technology (AIST), JAPAN
(2) University of Tsukuba, JAPAN
(3) Iwate Prefectural University, JAPAN

PACS: 43.72.Kb Speech communication systems and dialogue systems, 43.72.Ne Automatic speech recognition systems

ABSTRACT

This paper describes an efficient two-stage approach using sub-phonetic segment N-gram index and shift continuous dynamic programming for open vocabulary spoken term detection. With this two-stage search, we attempt to improve performance in both retrieval accuracy and process time. In the speech recognition process, a more sophisticated subword that is shorter than phonemes is used to minimize the effect of recognition error. Then, in the indexing and search process, N-gram and block addressing techniques are adopted to improve the search speed. In addition, in order to reduce missed errors in indexing, the N-best hypotheses are directly added to the inverted index. We investigate the properties of each method and examine their usefulness for the open vocabulary spoken term detection task.

INTRODUCTION

Information retrieval has become very important in modern society and has grown remarkably over the years. Information retrieval techniques involve the ranking of a collection of documents according to an estimate of their relevance to a query, allowing users to find relevant information easily and quickly. Thus far most research has been based on text databases [1].

With the dramatic expansion of Internet service and the increasing availability of mass storage devices, large amounts of multimedia material, such as images, speech, music, and video, are being accumulated and distributed on the Internet. Currently, these materials can only be retrieved based on user-oriented metadata rather than their actual content. However, as the available material becomes more diverse and varied, the demand for indexing and retrieving information based on speech contents continues to grow at a remarkable rate. Typically, Large Vocabulary Continuous Speech Recognition (LVCSR) is used to transcribe speech, and classical text-based information retrieval algorithms are then applied. The main problem with retrieving information from spoken data is the low accuracy of the automatic transcription. Any word in speech that is not in the vocabulary, i.e., Out-Of-Vocabulary (OOV), will be misrecognized as an alternate that is similar in acoustic feature. Word-based recognition systems are usually based on a fixed vocabulary, resulting in an index with a limited number of words and so do not permit searching for OOV words. Even though such systems can be quickly updated to enroll newly input words, it is generally difficult to obtain sufficient data to train the language models that include OOV words. An alternative method by which to solve this OOV problem is to use subwords, such as phonemes, morphemes, and syllables. In order to search misrecognized words and OOV words, the use of subword units as indexing terms has been employed in numerous systems [2]. From the NIST 2006 STD evaluation reports, systems based on word recognition have an advantage in accuracy over systems based on subword recognition. However, the use of subwords is still necessary in order to search the OOV words.

We have developed a subword speech recognizer and have proposed new subword units, i.e., sub-phonetic segments (SPS) [3,4]. In subword recognition, shorter units are more robust to errors and word variants than longer units, but longer units capture more discriminative information and are less susceptible to false matches during retrieval. Here, we investigate robust methods that take into account the characteristics of the recognition errors and attempt to compensate for the errors in an effort to improve open vocabulary spoken term detection performance.

In the proposed subword-based system, an efficient two-stage approach is based on an inverted index composed by an SPS N-gram and Shift-Continuous Dynamic Programming (SCDP). The present paper deals with the use of N-best hypotheses to minimize the number of missed error in indexing. In addition, block addressing is composed so as to reduce the space size of the index. The properties of each method are discussed based on experimental results, and the use of the proposed subword-based system for open vocabulary spoken term detection task is evaluated.

TWO-STAGE SEARCH

We attempt to perform the retrieval task through a two-stage search. The idea is to search full inverted indexes and perform sequence matching by SCDP sequentially. The index of the first stage does not provide exact time information, but only pointers to an area when the spoken term may be uttered. Next, a sophisticated sequence matching is adopted to locate phonetic sequences on the exact uttered time.

Inverted Index with block addressing

An index is an ultimate data structure built on the text in advance to speed up query processing. Inverted indexes are fundamental indexing structures for large text collections [1]. Inverted indexes are widely used to retrieve information in all practical search systems for large text databases storing natural language documents. The inverted index structure is composed of two elements, namely, all distinct words of the text collection (*vocabulary*) and, for each word in the vocabulary, a list of all text positions in which the word occurs (*occurrences*).

Block addressing is a technique to reduce the space requirements of an inverted index and was first proposed in a system called *Glimpse* [5]. The concept behind block addressing is that the text is logically divided into blocks, and the occurrences do not point to exact word positions but only to the blocks in which the word appears. Block addressing divides the text into blocks of fixed size. Since speech is a time-aligned sequence, the recognized subword transcription contains exact time information. In the present study, we propose time-block addressing to segment the subword transcription into fixed time intervals.

Shift-Continuous Dynamic Programming

When subword sequences are recognized directly with higher error rates than for words, a competent matching approach is necessary. The previously proposed Shift-Continuous Dynamic Programming (SCDP) is an algorithm that identifies similar parts between a reference pattern R_N and the input pattern sequence I_T [6]. The pre-fixed part of the reference pattern, referred to as the unit reference pattern (URP), is shifted from the start point of the reference pattern to the end by a certain number of frames. The matching results for each URP in the reference pattern are then compared and integrated.

$$R_N = \{R_0, \cdots, R_\tau, \cdots, R_{\tau+r}, \cdots, R_{N-1}\} \quad (1)$$

$$I_T = \{I_0, \cdots, I_t, \cdots, I_{t+i}, \cdots, I_{T-1}\} \quad (2)$$

The first URP is taken from R_0 in the reference pattern R_N, and the next URP is then composed of the same number of N_{URP} frames from the $(N_{shift}+1)_{th}$ frame. In the same manner, the k_{th} URP is composed of N_{URP} frames from the $k \times (N_{shift}+1)_{th}$ frame. Thus, the number of URPs becomes $[N / N_{shift}] + 1$, where $[]$ indicates any integer that does not exceed the enclosed value. SCDP is then performed for all URPs in the reference R_N. It is not necessary to normalize each cumulative distance at the end frame of a URP because all URPs are of the same length. Actually, SCDP is a very simple and flat algorithm that performs Continuous Dynamic Programming for each URP and integrates the results. The retrieved spoken terms are presented to the user in decreasing order of Dynamic Programming score, which is given as follows:

$$G(i,r) = argmin \begin{cases} G(i-1, r-1) + D(s_i, s_r) \\ G(i-2, r-1) + D(s_i, s_r) \\ G(i-1, r-2) + 2 \cdot D(s_i, s_r) \end{cases} \quad (3)$$

where $G(i,r)$ denotes the cumulative distance up to reference subword s_r and input subword s_i, and $D(\blacksquare)$ is the local distance, which uses a previously calculated distance matrix. Here, the Bhattacharyya distance D_{12} between two multivariate Gaussian distributions, $N_1(\mu_1,\Sigma_1)$ and $N_2(\mu_2,\Sigma_2)$, is given as follows:

$$D_{12} = \frac{1}{8}(\mu_1 - \mu_2)^T \left[\frac{\Sigma_1+\Sigma_2}{2}\right]^{-1} (\mu_1 - \mu_2) + \frac{1}{2} ln \frac{|(\Sigma_1+\Sigma_2)/2|}{\sqrt{|\Sigma_1||\Sigma_2|}} \quad (4)$$

SUBWORD-BASED APPROACH

Sub-phonetic segments

Here, we consider the feasibility of typical subword units for open vocabulary spoken term detection. In current speech recognition, the typical subword units are phonemes, syllables, or triphones (context-dependent phonemes). Triphones are effective subword units for LVCSR due to the representation of co-articulation effects. The difficulty with triphones is the large number of parameters to be estimated with respect to the limitation of training data. For this reason, we have proposed sub-phonetic segments as new subword units for spoken document retrieval [3,4]. The SPSs are derived from phonemes and refined under the consideration of the acoustic co-articulatory effects. The advantage of training SPS models is that pronunciation variation is trained directly into the acoustic model and does not need to be modeled separately in the vocabulary. In Japanese, 1,610 SPSs can be extracted from the 43 Japanese phonemes. Since some concatenations of phonemes do not exist in real language, 463 SPSs are defined. A graphical description of phonemes and triphones, which are widely used in LVCSR and SPSs, re-estimated from a phoneme sequence consisting of stationary and non-stationary segments, is presented in Figure 1.

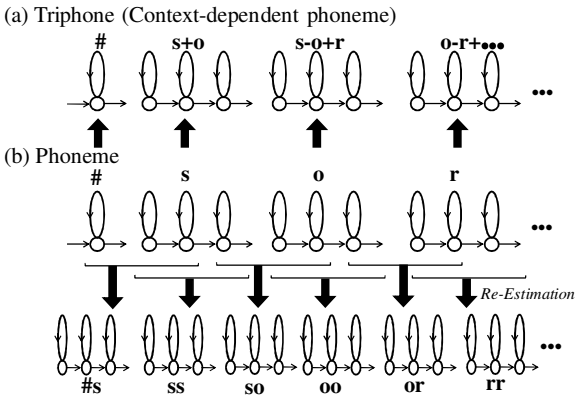

Figure 1: Graphical description of a context dependent phoneme and the proposed SPS

Time-block addressing index of the SPS N-gram

As mentioned previously, an inverted index is a word-oriented mechanism for indexing a text collection in order to speed up the searching task. However, the main drawback in composing an inverted index with subwords is its enormous space size. Furthermore, N-gram indexing, which uses n successive subword concatenation as indexing units, requires much more space. The proposed index structure with time-block addressing is as follows. The speech is recognized as subwords, and the subwords are the concatenated into an N-gram. The index stores all of the different SPS N-grams of the speech as *vocabulary*. For each SPS N-gram, the list of the time-blocks in which the SPS N-gram is uttered is recorded as the *occurrences*. Figure 2 shows the graphical description of composing the inverted index with an SPS N-gram. In speech data, the number of blocks indicates the time in which the spoken term is uttered. Therefore, we refer to this process as time-block addressing.

To search a spoken term in the inverted index, the *vocabulary*, SPS N-gram is scanned to query. When the SPS N-gram of the query is matched, the matching score of each *occurrence* time-block is raised by 1. The matched time-blocks are then listed by score.

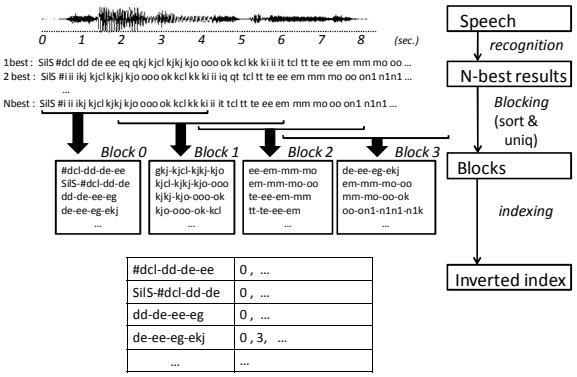

Figure 2: Example of building an inverted index with SPS 4-gram. The recognized SPS is concatenated with 4 and then split into each time-block. The SPS 4-grams in the time-block are sorted alphabetically and unified. Next the inverted index is built with the SPS 4-gram and time-block number. Occurrences in the index denote the time-block number, which is exactly matched with real time.

N-best hypotheses

Typically, speech recognition systems output the most probable match as the recognition output, but maintain multiple hypotheses that are considered during the recognition process. Multiple hypotheses, known as N-best hypotheses, provide additional information about the correct hypothesis. Using these additional hypotheses seems promising for spoken term detection, since it offers the chance of capturing spoken terms that would otherwise be missed by the speech recognizer in documents. This allows spoken terms to match with query terms, increasing document recall [7].

Mangu et al. presented a method by which to align a speech lattice with its 1-best transcription, creating confusion networks [8]. The difficulty in confusion networks is time alignment. Furthermore, time alignment is more difficult with shorter subwords, such as SPS. Rather than the recomposed confusion matrix, we use the N-best hypothesis directly. Since the SPS N-gram is sorted and unified in the time-block, as described in Figure 2, the space can be reduced efficiently. The spoken document representation is expanded to include multiple hypotheses to increase the chance of providing the correct hypothesis.

EXPERIMENTAL RESULTS

In this section, we present experimental results for open-vocabulary spoken term detection. The corpus consists of 10 news paragraphs in Japanese, read 30 times by 13 male and six female speakers. Thus, the corpus is composed of 300 paragraphs. Each story is approximately one minute long. The total length of the corpus is 377 minutes (six hours and 17 minutes). Each time-block is four seconds in duration and shift with two-second overlaps. The total number of blocks of the corpus is 11,332. Each paragraph contains 10 keywords, which are uttered twice. Thus each keyword has 60 relevant locations in the entire corpus. The number of queries is 100 in the experiments. Our task is to retrieve when the spoken term is uttered for a given text query. In Japanese, text can be converted into its phonetic representation using conversion rules, whether the query term is OOV or IV.

For evaluating retrieval performance, we use precision and recall with respect to manual transcription. Let *Correct(q,j)* be the number of times that query q is retrieved correctly in the j-th ranked document. Let *Retrieved(q)* be the number of retrieved documents for query q, and let *Relevant(q)* be the total number of times that q appears in the database.

$$Precision(q,j) = Correct(q,j) \Big/ Retrieved(q) \quad (5)$$

$$Recall(q,j) = Correct(q,j) \Big/ Relevant(q) \quad (6)$$

We compute the precision and recall rates for each query and report the average over all queries. The precision and recall at all retrieved points can be plotted as a curve. In addition to individual precision-recall values, we also compute the F-measure, which is defined as

$$F = 2 \times Precision \times Recall \Big/ (Precision + Recall) \quad (7)$$

The maximum F-measure of each query is presented in order to summarize the information in a precision-recall curve as a single value. We then average the maximum F-measure over all queries. All of the experiments of the present study are carried out on an Intel Core2 Quad CPU Q6700 2.66-GHz machine with 3.2 GB of main memory running the Linux operating system. In the present paper, the process time is calculated and compared with the execute time without file I/O overhead.

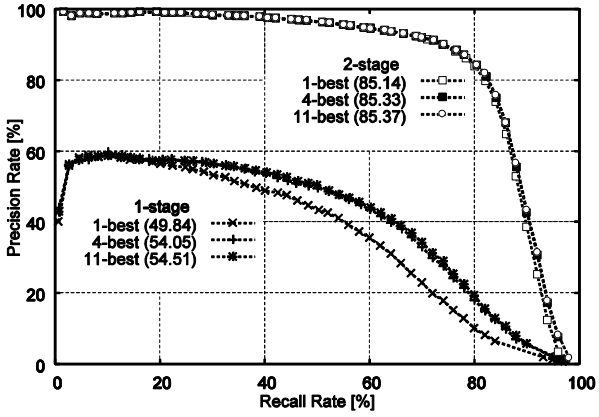

Figure 3: Precision and recall rates for each N-best in the first and second stage. The values in parentheses indicate the average of the maximum F-measures.

Figure 3 shows the Precision and Recall curves, precision and recall rates average over all queries. Since the sequential order of SPS N-gram of input query is not regarded in the 1-stage, the results of the 1-stage are very poor. However, the SCDP is performed on the exact phonetic sequence of SPS in the 2-stage. Thus, the results of the 2-stage are increased significantly. As shown in Figure 4, the performance becomes worse as the number of TOP ranked results from the 1-stage becomes smaller, which is used as candidates for the 2-stage. The lower performance is not avoided by limiting the candidate for speed-up in the 2-stage. However, the TOP300

shows only 3.2% degradation while the search space in the 2-stage is decreased by 97.35% from 11,332 time-blocks to 300.

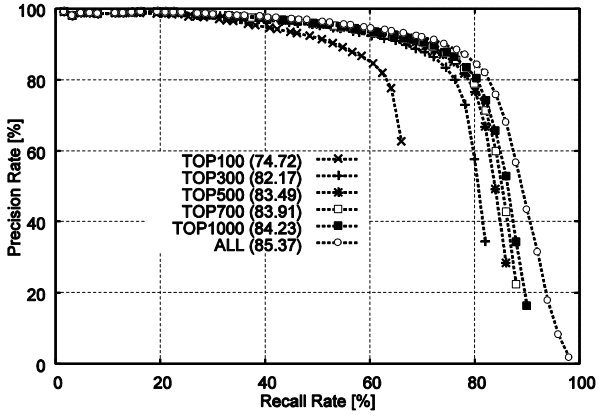

Figure 4: Precision and recall rates for each TOP-N. N is the top-N rank of the first stage, which is the input number for the second stage.

Table 1: Average of the maximum F-measure in each N-gram derived from 1-best and 11-best hypotheses in the first stage search with the inverted index and second stage search with SCDP for the Top-N candidates from the first stage.

N-gram	1st stage with Inverted Index	2nd stage with SCDP			
		Top-all	Top-1000	Top-500	Top-100
1-best					
1	41.54	85.34	82.29	79.50	64.73
2	44.15	85.35	83.30	81.03	67.67
3	49.05	**85.34**	**83.34**	**81.84**	**71.49**
4	**49.84**	85.14	82.98	81.51	71.22
5	49.37	84.30	82.43	80.76	70.32
6	48.57	82.36	81.08	79.46	68.80
7	47.64	79.29	79.09	77.63	67.67
8	46.21	74.49	74.42	73.79	65.52
11-best					
1	43.19	85.41	82.68	79.99	65.07
2	46.31	85.41	83.92	81.71	68.46
3	53.18	**85.41**	**84.53**	83.31	74.18
4	54.51	85.37	84.23	**83.49**	**74.72**
5	**54.67**	85.14	83.99	82.95	74.23
6	54.09	84.25	83.38	82.19	73.25
7	53.50	82.73	82.26	81.40	72.31
8	52.16	79.63	79.46	78.93	70.18

Expanding the documents with the N-best hypotheses can improve performance. As shown in Table 1, while the index is extended from 1-best hypotheses to 11-best hypotheses, the average of maximum F-measures is increased from 49.84% to 54.67%, an increase of 4.43% in the first stage search with the inverted index. Since the second stage search with SCDP is performed on 1-best hypotheses, the results of the second stage in TOP-all show only a slight increase in smaller N-grams. However, when the second stage search with SCDP is performed, the Top-N results of the first stage search with the inverted index indicate that the average of maximum F-measures is increased by only 2-3% at maximum point, and up to 4.66% when using an 8-gram. Based on these results, we can confirm that using the N-best hypotheses in the inverted index is effective for eliminating the effect of the recognition error.

Since the longer concatenation more easily suffers from recognition error, the retrieval accuracy with 1-best is maximized when using the 4-gram and then decrease, as shown in Table 1. However, as shown in Table 2, the process time with the larger N-gram becomes much faster. Even though the process time and memory space are slightly increased, using the N-best hypotheses is effective in order to minimize the effect of the recognition error on the longer concatenation,

Table 2: Process time (*msec*) of the first stage search with the inverted index. (The average number of SPS N-grams in a time-block is shown in parentheses.)

N-gram	1-BEST	4-BEST	11-BEST
1	28.8 (62)	30.4 (75)	29.9 (80)
2	20.6 (81)	23.2 (106)	23.5 (117)
3	14.0 (88)	16.4 (122)	16.7 (138)
4	7.1 (92)	9.0 (134)	9.2 (156)
5	2.9 (93)	3.8 (142)	4.4 (168)
6	1.5 (93)	2.0 (147)	2.5 (178)
7	0.6 (93)	0.8 (151)	1.2 (186)
8	0.4 (92)	0.5 (154)	0.9 (193)

CONCLUSION

In the present paper, we have described the development of an open-vocabulary spoken term detection system based on subword units. The system can perform open vocabulary spoken term detection even if the query consists of OOV words. Several methods were used to improve retrieval accuracy and speed, including the use of SPS N-grams, expanding documents with N-best hypotheses, and time-block addressing. Based on the experimental results, we confirmed that these methods are helpful for open vocabulary spoken term detection. Future research will be conducted in order to evaluate the system for very large collections.

REFERENCES

1 R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval, Addison-Wesley, 1999.
2 K. Ng., "Subword-based approaches for Spoken Document Retrieval", Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, 2000
3 Shi-wook Lee, K. Tanaka and Y. Itoh, "Combining Multiple Subword Representations for Open-Vocabulary Spoken Document Retrieval", Proc. of ICASSP2005, pp.505-508, 2005
4 K. Tanaka, et al., "Speech data retrieval system constructed on a universal phonetic code domain", Proc. of ASRU2001, pp. 1-4, 2001
5 U. Manber and S. Wu. G_{LIMPSE}: A tool to search through entire file systems. Proc. of USENIX Technical Conference, pp.23-32, 1994
6 Y. Itoh, et al., "Automatic Labelling and Digesting for Lecture Speech Utilizing Repeated Speech by Shift CDP", Proc. of EUROSPEECH2001, pp. 1805-1808, 2001
7 Siegler, M.A., Witbrock, M.J., Alattery,S.T., Seymore, K.,Jones,R.E. and Hauptmann, A.G. "Experiments in Spoken Document Retrieval at CMU." In Proceedings of the Seventh Text Retrieval Conference(TREC-7), NIST Special Publication, 1988.
8 L. Mangu, et al., "Finding consensus in speech recognition: word error minimization and other applications of confusion network," Computer Speech and Language, vol.14, no.4, pp.373-400, 2000.