ROBUST AND FAST LYRIC SEARCH BASED ON PHONETIC CONFUSION MATRIX

Xin Xu, Masaki Naito, Tsuneo Kato
KDDI R&D Laboratories, Inc.
sh-jo, naito, tkato@kddilabs.jp

Hisashi Kawai
National Institute of Information and
Communications Technology
hisashi.kawai@nict.go.jp

ABSTRACT

This paper proposes a robust and fast lyric search method for music information retrieval. Current lyric search systems by normal text retrieval techniques are severely deteriorated in the case that the queries of lyric phrases contain incorrect parts due to mishearing and misremembering. To solve this problem, the authors apply acoustic distance, which is computed based on a confusion matrix of an ASR experiment, into DP-based phonetic string matching. The experimental results show that the search accuracy is increased by more than 40% compared with the normal text retrieval method; and by 2% ∼4% compared with the conventional phonetic string matching method. Considering the high computation complexity of DP matching, the authors propose a novel two-pass search strategy to shorten the processing time. By pre-selecting the probable candidates by a rapid index-based search for the first pass and executing a DP-based search among these candidates during the second pass, the proposed method reduces processing time by 85.8% and keeps search accuracy at the same level as that of a complete search by DP matching with all lyrics.

1. INTRODUCTION

An easy-to-use music information retrieval (MIR) system plays an essential role in realizing satisfactory music distribution services. Current commercial MIR systems accept diverse queries by text, humming, singing, and acoustic music signals. Among these types of queries, text queries of lyric phrases are commonly used (lyric search) [1]. As many MIR systems apply full-text search engines to search lyric, the issue of lyric search has been widely accepted as a solved issue by state-of-art text retrieval techniques. However, the authors' preliminary investigations on real world queries suggested that, users are likely to input incorrect lyric phrases (incorrect queries in this paper) into MIR systems resulting in a failed lyric search. The incorrect lyric phrases are due to unreliable human memory or mishearing, as users remember the lyric phrases when they are impressed by hearing a part of a song without a lyric sheet. The analysis found that incorrect queries which replaces a word with another word of a similar pronunciation reaches 19%. This phenomenon is called "acoustic confusion" here.

In text retrieval field, some fuzzy algorithms, such as Latent Semantic Indexing (LSI) and partial matching, were used by major commercial Web search engines [2] to improve the robustness against incorrect queries. However, Xu's research verified that these algorithms were not helpful for acoustic confusion [3].

To solve this problem peculiar to lyric search, a search method is expected to be able to identify a lyric containing a part that is most similar in acoustic respect to the query. Phonetic string matching, which is used in such applications as name retrieval [4], was considered to be closest to the expected method. It uses edit distance between phoneme strings to search words with similar sound. However, edit distance is the minimum number of operations needed to transform one string into the other [5]. It does not present the degree of acoustic confusability between phonemes. For example, /aki/ is easily misheard as /agi/ as opposed to /aoi/, though the edit distances are identical. This is because the phonemes, "k" and "g", tend to be confused mutually, compared with "k" and "o".

In order to take the degree of acoustic confusability between phonemes into account for string matching, the authors apply a new distance, called acoustic distance, to phonetic string matching. Acoustic distance is obtained by DP matching with the costs derived from phonetic confusion probabilities between the phoneme strings of a query and a lyric. It is motivated by the ideas in Spoken Document Retrieval (SDR) and Spoken Utterance Retrieval (SUR) [6, 7]. Phonetic confusion probabilities are derived from a phonetic confusion matrix that is obtained from a preliminary automatic speech recognition (ASR) experiment.

As it is found by authors' preliminary investigations that queries of lyric phrases are not segmented by word or sentence boundary, edge-free DP matching between two phoneme strings of a query and a lyric is used to calculate acoustic distance. The computation complexity of DP matching $O\{DP\}$ cannot be ignored here because lyrics always contain long phoneme strings. Conventional phonetic string matching applied a complete search by DP matching with all lyrics, so the computation complexity is regarded as $O\{DP\} * I_t$, where I_t is the number of lyrics to search. Since commercial MIR systems usually provide hundreds of thousands of lyrics, the computation complexity is too high to realize a real time search.

Therefore, a lyric search method with a two-pass search strategy is proposed to speed up the search process. In the first pass, the proposed method pre-selects the probable lyric candidates by a rapid approximate search based on the accumulation of pre-computed and indexed partial acoustic distances. Then, a complete search by DP matching with the remaining lyrics is carried out during the second pass, which decreases the value of I_t contributing to the computation complexity.

The experimental results show that the application of phonetic confusion probability improves search accuracy in lyric search. Moreover, the processing time is greatly reduced by using two-pass search strategy.

The remainder of this paper is organized as follows: the analysis of real world queries is described in Section 2. The definition of acoustic distance and the proposed method are introduced in Section 3. The experiments are carried out to evaluate the proposed method in Section 4. The paper is summarized in Section 5.

2. ANALYSIS OF REAL WORLD LYRIC QUERIES

To analyze the queries of lyric phrases for MIR in the real world, the authors investigated some question & answer community web sites, where many questions were found that used lyric phrases to request the names of songs and singers. As 1140 queries of lyric phrases were collected, the authors compared each query with its corresponding lyric to distinguish whether lyric phrases in the query are correct or not (correct query or incorrect query) and how they were mistaken. The lyrics and queries are written in Japanese or English, or a mixture of both.

Figure 1 shows the distribution of incorrect queries in the different types and correct queries within the collected data. The incorrect queries, which occupy around 79%, are classified into the following types:

- Confusion of notations: Chinese characters in the queries are substituted for syllabary characters (Hiragana or Katakana in Japanese), or vice versa.
- Function-word-error: Only the function words, which have little lexical meaning, such as prepositions, pronouns, auxiliary verbs, are mistaken in the queries.
- Content-word-error: The content words such as a noun, verb, or adjective, that have a stable lexical meaning, are mistaken in the queries.

In the current full-text search methods, function-word-error and confusion of notations can be handled using a stop word list to filter out the function words [8], and a hybrid index of words and syllables [9].

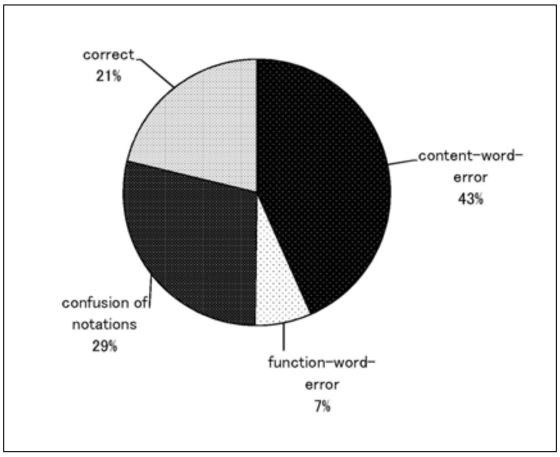

Figure 1. The distribution of incorrect queries in the different types and correct queries within the collected queries

On the other hand, as the content words play more important roles in determining the search intension [8], content-word-error queries were further categorized into three subtypes by the authors, viz., namely "acoustic confusion", "meaning confusion" and "others". The percentages and examples are listed in Table 1. The mistaken parts are marked in bold.

Acoustic confusion is defined as a replacement of a word with that of a similar pronunciation; or a replacement of the unknown-spelling words with syllable strings of a similar pronunciation. For the first example of acoustic confusion queries in Table 1, "/kotoganai/" and "/kotobawanani/" have similar pronunciations while the text strings have no common parts. In the second example, the Japanese syllable string is used as a query whose pronunciation is similar to the English phrase, "You've been out riding fences for so long now" in the target lyric. It was supposed to happen when users were not able to spell the foreign words that they heard in a song.

Meaning confusion is defined as a replacement of a word with its synonym or near-synonym. As shown in Table 1 the first example of meaning confusion queries, "/anata/" is mistaken for "/kimi/". Both of the terms refer to the same meaning "you" in Japanese. For the second example, "/tsuki/" and "/hoshi/", which mean "moon" and "star", are confused.

The type of "others" contains word insertion, word deletion and other errors in the queries. From the analysis of collected examples, it is known that mistakes in "others" type are derived from arbitrary reasons, which include individual experiences or memories, special environments, and other reasons. The analysis did not find a relationship between the mistakes and the lyrics.

As the acoustic confusion queries occupy about 45% of content-word-error queries (19% of the collected queries), it remains an important issue for lyric search. Based on the description above, identifying a lyric containing a part that is most similar in the acoustic aspect of the query is a better solution for acoustic confusion than focusing on the textual or semantic aspects.

Types of queries	Percentage			Examples	
				Correct lyric	Mistaken queries
Acoustic confusion	45%	Ex.1	Text (Japanese):	好きな**事がない**	好きな**言葉は何**
			Pronunciation:	/sukina**kotoganai**/	/sukina**kotobawanani** /
			Meaning:	There is **nothing** I like.	What are you favorite **words**?
		Ex.2	Text:	**You've been out riding fences for so long now**	**ユーベーナウプラウドゥンシーンクスソーセングナウ**
			Pronunciation:	**/yuubiibiiNNautoraidiNNgufeNNshizufoosooroNNgunau /**	**/yuubeenaupurauduNNshiiNNkususooseNNgunau/**
			Meaning:	You've been out riding fences for so long now	* no actual meaning
Meaning confusion	17%	Ex.1	Text (Japanese):	**君**には何でも話せるよと	**あなた**には何でも話せるよと
			Pronunciation:	/**kimi**niwanaNNdemohanaseruyoto/	/**anata**niwanaNNdemohanaseruyoto/
			Meaning:	I can say anything to **you**	I can say anything to **you**
		Ex.2	Text (Japanese):	**月**に願いを	**星**に願いを
			Pronunciation:	/**tsuki**ninegaio/	/**hoshi**ninegaio/
			Meaning:	pray to the **moon**	pray to the **star**
Others	38%	Ex.1	Text (Japanese):	星**から来た**子の見る夢は	星の子**チョビン**の見る夢は
			Pronunciation:	/hoshi**karakita**konomiruyumewa /	/hoshinoko**chobiNN**nomiruyumewa /
			Meaning:	The dream that the child who **came from** the star has	The dream that child **Chobin** of the star has

Table 1. The distribution of mistaken types within content-word-error cases

3. AN EFFICIENT SEARCH METHOD BASED ON ACOUSTIC DISTANCE

3.1 Introduction of Acoustic Distance

The authors introduce acoustic distance to quantify the degree of acoustic confusion. Acoustic distance is calculated by DP matching with cost values derived from phonetic confusion probabilities, instead of the constant cost values used for edit distance.

First, a phonetic confusion matrix is obtained by running a phoneme speech recognizer over training data and by aligning the recognition results of phoneme strings with reference phoneme strings.

For the elements of the confusion matrix, $n(p,q)$ means the number of phoneme q obtained as recognition results by the actual utterances of phoneme p. As "ϕ" represents a null, $n(\phi,p)$ means the number of the misrecognized phoneme p (insertion) and $n(p,\phi)$ means the number of the deleted phoneme p (deletion). M represents the set of phonemes including null.

For each phoneme p, the phonetic confusion probabilities of an insertion $P_{ins}(p)$, deletion $P_{del}(p)$ and substitution for phoneme q $P_{sub}(p,q)$ are calculated on the basis of the confusion matrix elements, by Eq.1∼3.

$$P_{ins}(p) = \frac{n(\phi,p)}{\sum_{k \in M} n(k,p)} \tag{1}$$

$$P_{del}(p) = \frac{n(p,\phi)}{\sum_{k \in M} n(p,k)} \tag{2}$$

$$P_{sub}(p,q) = \frac{n(p,q)}{\sum_{k \in M} n(p,k)} \tag{3}$$

As a large value of $P_{ins}(p)$ represents a high confusability for an insertion of p, it corresponds to a low cost of an insertion operation for p in string matching based on DP. Therefore the value of insertion cost $C_{ins}(p)$, is calculated by Eq.4. In the same way, the value of deletion cost $C_{del}(p)$ and substitution cost $C_{sub}(p,q)$, are calculated from the corresponding phonetic confusion probabilities by Eq.5 and Eq.6.

$$C_{ins}(p) = 1 - P_{ins}(p) \tag{4}$$

$$C_{del}(p) = 1 - P_{del}(p) \tag{5}$$

$$C_{sub}(p,q) = 1 - P_{sub}(p,q) \tag{6}$$

Second, with the calculated cost values, edge-free DP matching between the phoneme strings S_1, S_2 is carried out by Eq.7∼9. Here, $S[x]$ is xth phoneme of phoneme string S and $len(S)$ means the length of S ($S_1, S_2 \in S$). $D(i,j)$ designates the minimum distance from the starting point to the lattice point (i,j). D_{S_1,S_2} is the accumulated cost of DP matching between S_1 and S_2, which is defined as the acoustic distance. It reflects acoustic confusion probability for each phoneme.

1. Initialization:

$$D(0,j) = 0 (0 \leq j \leq len(S_2)); \tag{7}$$

2. Transition:

$$D(i,j) = \min \begin{cases} D(i,j-1) + C_{ins}(S_2[j]) \\ D(i-1,j-1) + C_{sub}(S_1[i],S_2[j]) \\ D(i-1,j-1), (S_1[i] = S_2[j]) \\ D(i-1,j) + C_{del}(S_1[i]) \end{cases} \tag{8}$$

3. Determination

$$D_{S_1,S_2} = min\{D(len(S_1), j)\}(0 < j \leq len(S_2)); \quad (9)$$

3.2 Searching Method based on Acoustic Distance by DP matching

Based on the criterion that a lyric containing a part that has the minimum acoustic distance from the query should be the user's target, a method with a complete search by DP matching with all lyrics is described as follows:

1. The lyrics L_{I_t} are converted into syllable strings using a morphological analysis tool such as Mecab [10]. The syllable strings are converted into phoneme strings by referring to a syllable-to-phoneme translation table. Consequently, a phoneme string $S_{L(k)}$ represents a lyric $L(k)$ $(L(k) \in L_{I_t})$.
2. Once a query Q is provided, it is converted into a phoneme string S_Q in the same way as step 1. By Eq.7∼9, the acoustic distance $D_{S_Q,S_{L(k)}}$ between the query and all lyrics L_{I_t} is calculated.
3. Lyrics L_{I_t} are ranked in the order of the acoustic distance $D_{S_Q,S_{L(k)}}$, and then the lyrics with lower distance values are provided as search results.

3.3 Searching Method based on Acoustic Distance by Two-pass Searching

Considering that the complete search by DP matching with all lyrics requires high computation complexity, a method with two-pass search strategy is proposed and realized with following steps:

- Preliminary indexing: An inverted index construction is preliminarily incorporated for the first pass search. A list of linguistically existing units of N successive syllables (syllable N-gram) $A_1 \cdots A_n$ are collected from the text corpus. The units are organized as index units for fast access, as shown in Table 2. The acoustic distance $D_{S_{A_n},S_{L(k)}}$ between the phoneme strings of A_n and $L(k)$ are pre-computed by Eq.7∼9 and stored in the index matrix. It can be regarded as an index of acoustic confusion.
- First pass search: By accessing the index described above, a fast search is realized by using the four steps below, and the flowchart is illustrated in Figure 2:
  1. The input query Q is converted into a syllable string v by Macab.
  2. By Eq.10 the syllable string is converted into syllable N-gram sets, $V_1, \cdots, V_m, \cdots, V_M$. Here, $v[m]$ is the mth syllable of v.

  $$V_m = \{v[m], v[m+1], \cdots, v[m+N-1]\}; \quad (10)$$

  3. $V_1, \ldots, V_m, \ldots, V_M$ are matched with the index units $A_1, \ldots, A_n, \ldots$. By accumulating the pre-computed and indexed distance values $D_{S_{A_n},S_{L(k)}}$, the approximate acoustic distance $R(k)$ is calculated by Eq.11.

  $$R(k) = \sum_{m=1,\cdots,M} D_{S_{A_n},S_{L(k)}}, (V_m = A_n) \quad (11)$$

  4. To narrow the search space of lyrics, $L(k)$ with higher $R(k)$ is pruned off, and a lyric set L_{I_c} containing I_c $(I_c < I_t)$ best lyric candidates is preserved for the second pass.
- Second pass search: A complete search by DP matching with the lyrics in L_{I_c} is carried out.

Based on the processes above, the computing complexity of the proposed method is reduced to $O\{FPS\}+O\{DP\}* I_c$. As $O\{FPS\}$ is the computing complexity of the first pass search which is much less than $O\{DP\}$ and I_c is much less than I_t, it provides a faster response for a real-time MIR system.

4. EVALUATION OF SEARCH ACCURACY AND PROCESSING TIME

4.1 Experimental Set Up and Test Set

To evaluate the search performance of the proposed search method, experiments were carried out. A database of 10000 lyrics was collected containing both Japanese and English lyrics. The test set consisted of 220 incorrect queries that were mistaken in acoustic confusion. They were from the collected queries mentioned in Section 2. The lyrics corresponding to the queries were included in the database. The results of the experiment were obtained using a personal computer (Intel Core2Duo CPU 3.0GHz, 4G RAM). Four methods described as follows, were compared.

- "Baseline": A normal partial matching method using full-text retrieval engine "Lucene", which is based on inverted index construction [11]. In this method, as the query of lyric phrases is divided into N successive character substrings, the lyric containing more substrings is regarded as a more suitable candidate.
- "Method based on Edit Distance of Phoneme (EDP)": The search method based on the edit distance of phoneme strings, which is described in [4].
- "Method based on Acoustic Distance by DP matching (ADDP)": The search method using a complete search described in Section 3.2. The phonetic confusion matrix for calculating acoustic distance is obtained using the same speech recognition experiment as in [12]. Although the confusion matrix should be based on singing voice, a huge amount of singing data is not available. In this research, telephone speech data of Japanese phonetically balanced sentences were

Lyric No. syllable 3-gram	`L(1)`	···	`L(k)`	···
A_1 `[a-i-u]`	0.34	···	0.23	···
···	···	···	···	···
A_n `[na-ko-to]`	0.88	···	$D_{S_{A_n}, S_{L(k)}}$	···

Table 2. A sample of index item by syllable 3-gram

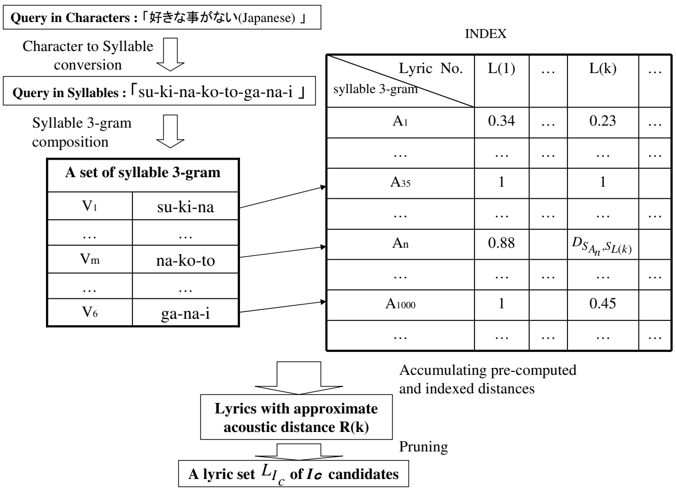
Query in Characters : 「好きな事がない(Japanese) 」
Character to Syllable conversion
Query in Syllables : 「su-ki-na-ko-to-ga-na-i 」
Syllable 3-gram composition
A set of syllable 3-gram
V1 su-ki-na
…
Vm na-ko-to
…
V6 ga-na-i
INDEX
Lyric No.
syllable 3-gram
L(1)
L(k)
A1 0.34 0.23
A35 1 1
An 0.88 $D_{S_{A_n}, S_{L(k)}}$
A1000 1 0.45
Accumulating pre-computed and indexed distances
Lyrics with approximate acoustic distance R(k)
Pruning
A lyric set L_{I_c} of I_c candidates

Figure 2. Flowchart of the first pass search

used as the training data for the acoustic models of ASR.

- "Method based on Acoustic Distance by Two-pass Searching (ADTS)": The proposed method using the two-pass search strategy as described in Section 3.3. Considering the balance of index size and search accuracy, here N of syllable N-gram index is set to 3. The syllable 3-grams are collected from the lyric and newspaper corpus. 100,000 entries of syllable 3-grams, which cover 90% of all syllable 3-grams in the collected text corpus, are prepared in the index. As all the syllable 3-grams which exist in the queries are prepared, no search errors come from out-of-vocabulary syllable 3-grams in the experiments.

4.2 Improvements of Search Accuracy by Applying Acoustic Distance

The comparison of the results between "EDP" and "ADDP" are shown in Figure 3. The vertical axis means the hit rate, while the horizontal axis shows the top T candidates of the ranked lyrics, which is called T-best. Here, the hit rate of T-best is defined as the rate of the total number of hits within top T candidates to the total number of search accesses. "ADDP" improves the hit rates by 2% $\sim$ 4%, as the T of T-best is ranged from 1 to 100. It indicates that the proposed acoustic distance gives better effects than edit distance.

4.3 Evaluation of Search Accuracy and Time Complexity

The search accuracy and time complexity of four methods are shown in Figure 4 and Table 3 respectively.

Note that, the value of I_c for "ADTS" is determined by a preliminary experiment based on the same test set. It only uses the first pass search of "ADTS" for lyric search to investigate the relationship between hit rates and T-best to chose the best threshold value for I_c. Because the hit rates almost saturated when T is larger than 800, I_c is set to 800 in this paper.

With a well-designed data structure, "Baseline" achieved the fastest response among four methods. However, since the normal text retrieval techniques cannot solve the acoustic confusion problem in lyric search, other three methods based on phonetic string matching achieved higher search accuracy than "Baseline" by more than 40%. On the other hand, though "ADDP" and "EDP" provide high performances of search accuracy, the processing times for one query are over 9 seconds, which are not practical in real world search. By applying a fast search in the first pass to narrow the search space, "ADTS" shortens the processing time into 1.85 seconds, which is 14.2% of "ADDP", with only 0.5% $\sim$ 5% deterioration of search accuracy due to the loss happened in the index-based pruning of "ADTS".

Attributing to the application of acoustic distance, "ADTS" keep almost the same hit rates as "EDP" and achieves 2% improvement when T is larger than 20, by using only 19% time of "EDP".

5. CONCLUSION

This paper proposed a robust and fast lyric search method based on the introduced acoustic distance and a two-pass search strategy using an index-based approximate preselection for the first pass and a DP-based string matching in the second pass. In the case of incorrect queries caused by acoustic confusion, the proposed method achieved significantly higher search accuracy than the normal text retrieval method by more than 40%. An improvement by 2% ∼4% is also achieved compared with the conventional phonetic string matching method. Furthermore, the proposed method realized a real time operation by reducing 85.8% processing time with a slight loss in search accuracy compared with a complete search by DP matching with all lyrics. It is proved to be the most practical solution for acoustic confusion queries on the balance of high search accuracy and light computation complexity.

6. REFERENCES

[1] Downie and Cunningham: “Toward a theory of music information retrieval queries: System design implications,” *Proceedings of the Third International Conference on Music Information Retrieval (ISMIR 2002)*, pp. 299–300, 2002.

[2] Denys Poshyvanyk et al.: “Combining Probabilistic Ranking and Latent Semantic Indexing for Feature Identification,” *the 14th IEEE International Conference on Program Comprehension*, pp. 137–148, 2006.

[3] Xin Xu, Masaki Naito, Tsuneo Kato, Hisashi Kawai: “An Introduction of a Fuzzy Text Retrieval System For Music Information Retrieval,” *Information Processing Society of Japan SIG Notes*, No.127, pp. 41–46, 2008.

[4] J. Zobel and P. Dart: “Phonetic string matching: Lessons from information retrieval,” *Proceedings of the 19th International Conference on Research and Development in Information Retrieval*, pp. 166–172, 1996.

[5] E. Ristad and P. Yianilos: “Learning string edit distance,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20, pp. 522–532, 1998.

[6] Ville T. Turunen, Mikko Kurimo: “Indexing confusion networks for morph-based spoken document retrieval,” *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 631–638, 2007.

[7] Takaaki Hori et.al: “Open-Vocabulary Spoken Utterance Retrieval Using Confusion Networks,” *Proceedings of the 2007 International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 73–76, 2007.

[8] Christopher Fox: “A stop list for general text,” *ACM SIGIR Forum*, Vol. 24, No. 1–2, pp. 19-21, Fall 1989/Winter 1990.

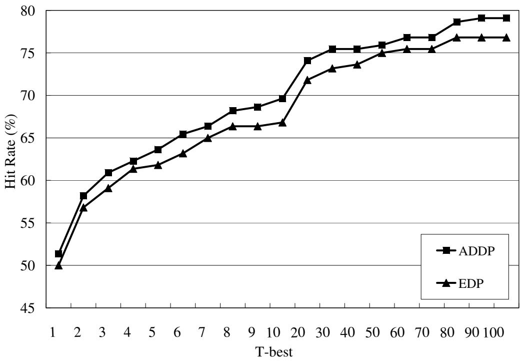

Figure 3. The improvement of search accuracy by acoustic distance

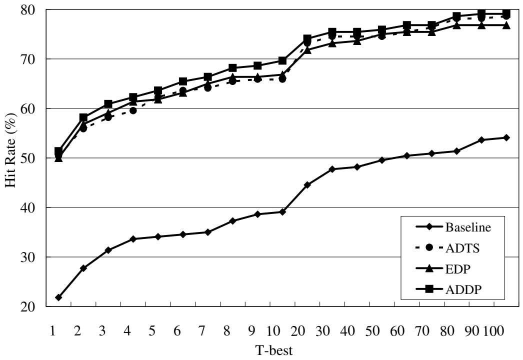

Figure 4. Search accuracy of four search methods

Methods	Baseline	ADTS	EDP	ADDP
Time (second)	0.006	1.85	9.72	13.01

Table 3. Average processing times of one query

[9] Nina Kummer, Christa Womser-Hacker and Noriko Kando: “MIMOR@NTCIR 5: A Fusion-based Approach to Japanese Information Retrieval,” *Proceedings of NTCIR-5 Workshop Meeting, Tokyo, Japan*, 2005.

[10] http://mecab.sourceforge.net

[11] Erik Hatcher, Otis Gospodnetic: *Lucene In Action*, Manning Publications Co., 2004.

[12] Makoto Yamada, Tsuneo Kato, Masaki Naito and Hisashi Kawai: “Improvement of Rejection Performance of Keyword Spotting Using Anti-Keywords Derived from Large Vocabulary Considering Acoustical Similarity to Keywords,” *Proceedings of INTERSPEECH*, pp. 1445–1448, 2005.