情報処理学会論文誌 Vol.54 No.12 2492–2501 (Dec. 2013)

音節N-gramの事前検索結果を利用した音声中の検索語検出の高速化方式

伊藤 慶明[1,a)] 斉藤 裕之[1] 田中 和世[2] 李 時旭[3]

受付日 2013年2月22日, 採録日 2013年9月13日

概要：本論文では，音声中の検索語検出（STD: Spoken Term Detection）において，音節を N 個並べた音節 N-gram の事前検索結果を利用した STD の高速化方式を提案する．提案方式では，すべての N-gram，すなわち N 個の音節のすべての組合せに対し検索対象の音声ドキュメントとあらかじめ照合し，その検索結果を事前検索結果として用意しておく．検索語が与えられると，検索語の音節列を 1 音節ずつずらしながら音節 N-gram に分割し，各音節 N-gram に対し事前検索結果を参照して一次候補区間を抽出する．次に一次候補区間に対し連続 DP によるリスコアリングを行う．すべての音声ドキュメントとの連続 DP を行う従来の全照合方式と比べ，一次候補区間として候補を削減することで検索時間の短縮を図る．2 音節事前検索結果を用いた実験において，検索精度を低下させることなくコア 177 講演で検索時間を 0.922 秒から 0.365 秒に，2,702 講演で検索時間を 16.10 秒から 2.87 秒に高速化することができた．上位候補の評価においては，2,702 講演の場合，上位 1，3，5，10 位いずれについても全照合と同精度で 1 秒未満で検索することができた．

キーワード：音声中の検索語検出，高速化，音節 N-gram，事前検索

A Fast Spoken Term Detection Method by Preparing Pre-retrieval Results for All Syllable N-grams

Yoshiaki Itoh[1,a)] Hiroyuki Saito[1] Kazuyo Tanaka[2] Shi-Wook Lee[3]

Received: February 22, 2013, Accepted: September 13, 2013

Abstract: We propose a method based on pre-retrieval results using syllable N-grams as query terms in advance in the Spoken Term Detection (STD). In the proposed method, all the combinations of syllable N-grams such as syllable bigram and trigram are searched in spoken documents, and the retrieval results are prepared as pre-retrieval results beforehand. When a query is given, the query is divided into sub-queries that are composed by syllable N-grams, shifting an N-syllable window. First candidates are extracted for each syllable N-gram, and re-ranked according to the score obtained by applying Continuous DP. The proposed method could reduce the retrieval time from 0.922 s to 0.365 s with no performance deterioration for Core 177 lectures sets and could reduce the retrieval time from 16.10 s to 2.87 s with no performance deterioration for 2,702 lectures.

Keywords: spoken term detection, speed-up method, syllable N-gram, pre-retrieval

[1] 岩手県立大学
Iwate Prefectural University, Iwate 020–0193, Japan
[2] 筑波大学
University of Tsukuba, Tsukuba, Ibaraki 305–8550, Japan
[3] 産業技術総合研究所
National Institute of Advanced Industrial Science and Technology, Tsukuba, Ibaraki 305–8568, Japan
[a)] y-itoh@iwate-pu.ac.jp

1. はじめに

近年，ハードディスクレコーダやブルーレイディスク等の大容量記録媒体が広く普及し，Web 上の動画等，大量のビデオデータを扱う機会が増加している．このような環境下において，大量のビデオデータ群の中から目的のデータを簡便

に検索する機能に対するニーズは高くなっている．この機能の実現に向け，現在音声中の検索語検出（STD: Spoken Term Detection）の研究がさかんに行われている．米国の米国国立標準技術研究所（NIST）による TREC（Text REtrieval Conference）では評価型ワークショップが行われ [1]，2011 年には国立情報学研究所が NTCIR Workshop 9 を日本で開催し，Spoken Doc Task [2] として STD の評価が行われた．2013 年の NTCIR Workshop 10 においても Spoken Doc Task が継続実施された．

STD とは，検索語が音声ドキュメント中で発話されている位置を特定することであり，STD では未知語の検索が重要となる．検索語が音声認識システムの辞書に登録されている既知語ならば単語認識結果を用い（単語ベース方式），検索語が辞書に登録されていない未知語ならばサブワード認識結果を用いる方式（サブワードベース方式）が一般的となってきている．我々は未知語の検索に頑健な STD システム構築を目指し，様々なサブワード認識結果を用いて未知語の検索精度の改善を行ってきた [3], [4]．

一般的なサブワードベース方式では，音声ドキュメント群をあらかじめサブワードで音声認識しておき，テキストで検索語が与えられると検索語をサブワード系列に変換し，サブワード系列の検索対象の音声ドキュメントと連続 DP（Continuous Dynamic Programming）等で照合を行う．我々の STD システムでは，連続 DP の局所距離にサブワード間の音響距離を用いることで検索精度向上を実現しているため，すべての音声ドキュメントとの照合が必要となり，検索時間は検索対象の音声ドキュメントの長さとほぼ線形に増加する．我々の STD システムでは，NTCIR-9 Spoken Doc の STD コアタスク，すなわち日本語話し言葉コーパス（CSJ: Corpus of Spontaneous Japanese）の 177 講演，約 44 時間に対する 1 検索語あたりの検索時間は約 1 秒，全講演タスク，すなわち 2,702 講演，604 時間に対する 1 検索語あたりの検索時間は約 16 秒であった．実用を考えた場合，1 秒程度で検索が完了するとともに，検索時間の増加が音声ドキュメント量と非線形であることが求められる．本論文では STD システムの高速化を実現するため，音節を N 個並べた音節 N-gram の事前検索結果を利用した方式を提案する．本提案方式では，m 種類の音節（今回は 261 種類の音節を利用：$m = 261$）を N 個並べた音節 N-gram を基本単位とする．音節 N-gram の組合せは m^N 個となる．そのすべての音節 N-gram に対して事前に検索対象の音声ドキュメントと照合を行い，事前検索結果として照合結果を保持しておく．検索語が与えられると，検索語を N 音節単位で 1 音節ずつシフトさせながら分割し，L（検索語の音節数 $-N+1$）個の音節 N-gram を抽出する．まず各音節 N-gram について事前検索結果を参照し，候補区間を一次候補区間として選出する．次に一次候補区間に対してのみ連続 DP 等で詳細な照合を行い，再度スコアリング（リスコアリング）をする．全区間を連続 DP の対象とするのではなく事前検索結果を用いて一次候補区間として絞り込むことで，検索精度を維持しながら検索時間の短縮を図る．

STD システムでは，検索精度，検索速度，インデクスのサイズと構築時間で評価が行われるのが一般的である．STD システムにおける検索語の検出は，検索語とすべての音声ドキュメントとを照合する方式（以下，全照合と呼ぶ）が基本となる．この全照合の検索精度が上限とされ，検索精度と検索速度はトレードオフの関係にあり，一般には高速化を狙うと全照合の検索精度より低下してしまう．一方，検索精度を向上させるためには，複数の認識結果を保持する方式や [5]，N-best で複数の候補を保持する方式等が提案されているが，全照合を行う検索時間をさらに要することになる．本提案手法では全照合の検索精度を低下させずに高速化を図るもので，本提案方式は高精度化を目指したこれらの方式にも応用可能である．検索精度において高精度化方式と競うのではない点に注意されたい．

STD の高速化方式はこれまで様々提案されてきた [6], [7], [8], [9], [10], [11], [12]．文献 [6], [7] に代表されるように，これまでの高速化法 [6], [7], [8], [9], [10], [11], [12] は音声ドキュメントをいったん単語あるいはサブワードで decode し，仮説として生成されたサブワードのラティスあるいはネットワークからインデックスを作り，検索時にはそのインデックスから同一/類似区間を検索するものである．これに対し，本提案方式では認識結果ではなく検索結果をインデックスとしている点に特徴がある．文献 [8], [9] では，2 つの認識システムで音節認識を行い，各々の 5-best 程度の結果と音声認識の結果の音節列に対し，N-gram をインデックスとして高速な検索を実現した．置換誤りに対しては音響距離を割り当て，脱落・挿入をインデックス構築の際に疑似的に作成し検索精度を改善した．文献 [10] では，転置インデックスの代わりに Suffix array を用いた．Suffix array では，Suffix となるインデックスが長くなると検索時間を長く要するため，検索語を 6 程度の音素に分割・固定することで高速なキーワード検索を実現した．認識結果をインデックス化し検索語が与えられた後にインデックスとの照合を行う点で本研究とは異なり，検索語の分割方式や高い Recall 時の検索時間等に課題が残っている．

認識結果をインデックスとして利用する多くの STD システムでは，たとえば検索語が「東京 (t o: ky o:)」で，「東京」と発話された区間の音素 ky が誤認識され，ky が候補にならなかった場合，文献 [11] では音素 N-gram 照合方式（実験で行われた $N = 3$）の段階でこの区間が候補とならない．本提案方式では認識結果ではなく検索結果をインデックスとしているため，認識結果のインデックスでは誤認識により候補となりえない区間でも一次候補として抽出することが可能であり，全照合方式と同レベルの検索精度が期

待できる．また本提案方式ではあらゆる検索語を事前に照合することを想定し，音節 N-gram をすべて検索しておくことで高速化を図るものである．本提案方式は，検索語に含まれる L 個の音節 N-gram について事前検索結果を参照するため，音節認識結果等から音節 N-gram を検索できるシステムであればどのような STD システムにおいても利用できる．高精度化を目指して複数の様々な認識単位を用いて得た複数の認識結果を利用する方式 [3], [4], [5] に対しては，文献 [7], [8], [9] の方法で対応可能であるが，本提案方式は各音節 N-gram に候補区間を持たせるだけのシンプルな構造であるため実装が容易であり，文献 [3], [4], [5] および N-best を扱う [8], [9] 等の STD システムで利用可能である点も本システムの特長と考える．N 種類の認識結果や N-best を用いること，さらにその結果をネットワーク化することにより STD の高精度化が実現されているが，N 個の認識結果に対し L 音節でインデックス化する場合には，最大 N^L 倍にインデックスサイズが大きくなる．NTCIR-9 で最も高い検索精度を出したシステム [5] では 10 種類の認識結果を扱うため，CSJ 全講演を対象とした場合インデックス化は難しくなるが，インデックス化を行わずに事前に検索を行っておく本提案方式は適用が可能である．

本論文では，まず従来の STD 方式について概説し，提案方式を詳述する．次に提案方式の評価を行う．実システムで利用する場合を考慮し検索結果の上位での評価を行い，本提案方式の有効性を示す．

2. 提案方式

2.1 従来の STD システムの概要

本節では，我々が提案するサブワードベースの STD システム [3] について，サブワードとして本論文で用いた triphone・音節を例に概説し，現状の検索時間を示す．

STD システムの概要を**図 1** に示す．検索対象の音声ドキュメントはポーズにより発話ごとにセグメンテーションされ，発話ごとにサブワード認識（音響モデルに triphone を用いた音節認識）を行い，認識結果として出力されるサブワード（triphone）系列を保持しておく．本論文では検索語はテキストで与えられるものとし，検索語は変換規則に則り自動でサブワード（triphone）系列に変換される．このサブワード系列の検索語と保持している認識結果のサブワード系列との間で連続 DP による照合を行う．照合時の局所距離には，サブワード（triphone）間音響距離を用いる．サブワードの各音響モデルは HMM（Hidden Markov Model）で構成されており，サブワード間音響距離は 2 つのサブワード（たとえば，triphone の a-k+i と a-k+e）間の違いを HMM の統計量から事前に以下の 3 つのステップで求めておいたものである [3]．

(1) 同一状態間の任意の分布間の距離を算出：それぞれ 3 状態からなる 2 つのサブワードにおいて同一順番の状態から分布を 1 つずつ取り出し，その 2 つの分布間の距離を算出する．分布間の距離には Bhattacharyya 距離を用いた．
(2) 同一状態間距離を算出：(1) で求めた同一状態間の分布間距離のうち最小の距離を同一状態間距離とする．
(3) サブワード間距離を算出：同一状態間距離の 3 状態の平均をサブワード間距離とする．

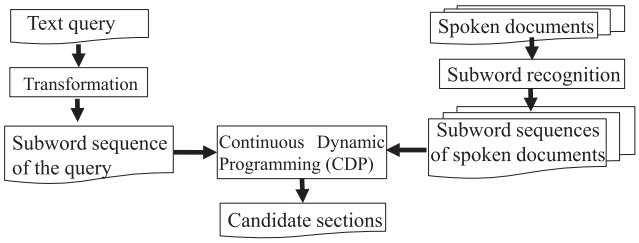

図 1 STD システムの概要
Fig. 1 Outline of our STD system.

サブワード間距離の平均は 8.50，および最大は 30.0，最小は 0.0，標準偏差は 3.50 であった．連続 DP における累積距離が小さい，すなわち類似度が高い順に候補区間をユーザへ提示する．

この STD システムは検索語と音声ドキュメント全体との照合を行う全照合方式であり，以下，実験条件，評価指標とともにベースライン性能となるこの全照合方式の検索精度と検索時間を示す．

(1) 音声認識条件

本論文では音響モデルはモデル数 3,500 に集約した集約 triphone [13] を用い，認識の単位は音節とした．集約 triphone とは，学習データ中の出現頻度が低い triphone を出現頻度の高いモデルへ集約した triphone モデルで，検索時の取りこぼしを削減し検索精度を向上させた [13]．学習データは CSJ 中の検索対象コア 177 講演を除いた本講演と摸擬講演のうちの講演 ID が偶数の講演を用いた[*1]．音響モデル構築には HTK を，言語モデル構築には Palmkit を用いた．言語モデルには音節単位の前向き 2-gram と後ろ向き 3-gram を，音声認識には大語彙連続音声認識エンジン Julius ver. 4.1.5.1 [15] を用いた．音響分析条件を**表 1** に示す．

(2) 評価用データ

検索対象の音声ドキュメントは CSJ コア 177 講演，約 44 時間分ならびに全 2,702 講演，約 604 時間分を用いる．音声ドキュメントは CSJ 付属の xml データで発話単位（IPU: Inter Pausal Unit）にセグメンテーションしてあり，コア 177 講演では全 53,892 発話，全 2,702 講演データでは 880,391 発話となる．システムでは検索語が対応する発話中の部分区間を出力可能であるが，CSJ では単語ごとの時間情報がないため，文献 [2] と同様に発話の単位で検索語

[*1] 本論文ではモデルの作成・認識の時間等の理由により，全 2,702 講演のうち 1,260 件の講演は Closed な認識となっている．

表 1 音響分析条件
Table 1 Acoustic analysis conditions.

標本化周波数	16 kHz
量子化	16 bit
音響特徴量	MFCC(12 dim.) + ΔMFCC(12 dim.) + ΔΔMFCC(12 dim.) + ΔPower + ΔΔPower (Total 38 dim.)
分析窓	ハミング窓
frame 長	25 ms
frame shift	10 ms

が含まれるか否かで正解判定を行った．本論文で用いる候補区間はこの発話単位の区間と同義になる．候補区間は音声ドキュメント中の何番目の発話かを表す発話 ID のみで特定され，発話 ID は全講演の場合 880,391 が最大となり 4 バイトで格納できる．

検索語は NTCIR-9 Spoken Doc Task フォーマルランで用いられたコア 177 講演用検索語と全 2,702 講演用検索語各 50 個を用いる．セグメンテーションされた発話内に検索語が含まれていれば正解とする．コア 177 講演用検索語 50 個の音素数は 6〜27，平均は 10.82 であり，正解数は 2〜23，平均 7.16 である．全 2,702 講演用検索語 50 個では，音素数は 6〜18，平均は 11.02 であり，正解数は 7〜45，平均 19.68 であった．

(3) 性能指標・時間計測

検索精度の指標には，NTCIR-9 で用いられた MAP（mean average precision）[2] を用いる．ある検索語における正解出現時の適合率の平均が AP（average precision）であり，各検索語における AP を全検索語で平均したものが MAP である．検索語 i の AP と MAP の計算式を以下の式 (1)，式 (2) で示す．

$$\mathrm{AP}(i) = \frac{1}{C}\sum_{j=1}^{R} \delta_j \times \mathrm{precision} \quad (i,j) \tag{1}$$

$$\mathrm{MAP} = \frac{1}{Q}\sum_{i=1}^{Q} \mathrm{AP}(i) \tag{2}$$

正解数を C，正解が出現した最低順位を R，検索語の数を Q とする．i 番目の検索語について連続 DP のスコアで順位付けを行い，j 番目の候補区間が正解であれば $\delta_j = 1$，不正解ならば $\delta_j = 0$ とし，式 (1) より i 番目の検索語について正解出力時の適合率（AP）の平均を算出する．式 (2) により各検索語の平均適合率から全検索語の平均（MAP）を算出する．

処理時間の計測には，Intel 社の Core i7 2600，メモリ 8 G の Linux マシンを使用し，C 言語の gettimeofday を用いた．

(4) 従来の検索精度と検索時間

我々が従来用いていたサブワードベースの STD システムの検索精度と検索に要する時間を**表 2** に示す．

表 2 検索精度と検索時間（CORE と ALL に対して）
Table 2 Retrieval performance and retrieval time (CORE and ALL sets).

検索対象の音声ドキュメント	CSJ コア 177 講演	CSJ 全 2,702 講演
発話件数	53,892 発話	880,391 発話
音声ドキュメント内の音素数	1,530,309	24,091,207
音響モデル	Triphone(3,500 models)	
言語モデル	Syllable	
MAP(%)	75.62	66.37
検索時間 (秒／1 検索語)	0.922	16.10

コア 177 講演の約 44 時間分の音声ドキュメントに対して 1 検索語の検索に約 1 秒，全 2,702 講演の約 604 時間分の音声ドキュメントに対して 1 検索語の検索に約 16 秒を要している．従来方式では，検索時の局所距離にサブワード間音響距離を用いているため，連続 DP 等ですべての音声ドキュメントと照合する必要があり，検索時間は音声ドキュメント量と線形に増加する．実際に今回音声ドキュメント内の音素数が約 150 万から 2,400 万と約 16 倍，検索時間が約 17 倍となり，音声ドキュメント量と比例して検索時間が増加した．

以下，音声ドキュメントの増加に対しても実用的な検索時間を目指した本論文の提案方式について述べる．

2.2 提案方式

2.2.1 音節 N-gram の事前検索結果の導入

本項では，提案する音節 N-gram による事前検索結果を用いた STD の高速化方式について述べる．

音声ドキュメント中，検索語が発話されている区間には，検索語を構成する音節列が存在する．たとえば，検索語「イワテ」が発話されている音声ドキュメント内の区間では「イワ」,「ワテ」の 2 つの音節 bigram が両方発話されている．そのため，この 2 つの音節 bigram を検索語として検索した場合の整合度も高いと想定できる．そこで，音節 N-gram のすべての組合せで検索を事前に行っておき，その検索結果（事前検索結果）を保持しておく．検索語が与えられると，検索語の音節列を複数の音節 N-gram に分割し，各音節 N-gram の事前検索結果を参照し一次候補区間として抽出する．これにより，この限定した候補に対してのみ連続 DP による最終的なリスコアリングを行うことで検索の高速化を図る．本提案方式は，**図 2** に示すように以下の 5 ステップからなり，1，2 は事前に処理しておく．

1. 検索対象の音声ドキュメントは，2.1 節と同様に音節認識を行いその認識結果であるサブワード系列を保持し

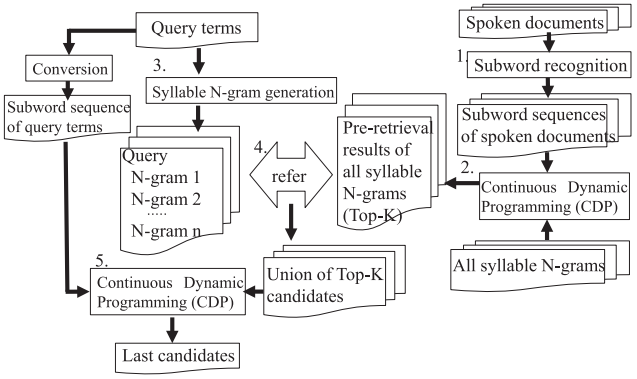

図 2 事前検索結果を用いた STD 高速化法の概要

Fig. 2 Outline of the proposed STD system using pre-retrieved results.

ておく（Subword sequences）．

2. 音節 N-gram のすべての組合せ（N = 2 であればアア，アイ，アウ，…）を検索語として，音声ドキュメントのサブワード系列と照合し，上位 K 件（Top-K と表す）までの候補区間を事前検索結果として保持する．
3. 検索語がテキストで与えられると，検索語を N 音節単位に 1 音節ずつずらしながら L 個（検索語の音節数 −N + 1）の音節 N-gram を抽出する．検索語が「イワテ」で N = 2 ならば，「イワ」と「ワテ」の 2 つの音節 bigram を抽出する．
4. 検索語中の各音節 N-gram について，事前検索結果を参照し類似度が高い上位 K 件（Top-K）の区間を選出する．検索語中の L 個の音節 N-gram に対して各 K 件ずつの候補区間が選出され，それらの和集合をとり一次候補区間とする（Union of Top-K candidates）．検索語が「イワテ」の例であれば，「イワ」と「ワテ」それぞれの事前検索結果について上位 K 件，計 2K 件の候補区間を選出しその和集合を一次候補区間として抽出する．
5. ステップ 4 で求めた一次候補区間に対して，連続 DP で検索語のサブワード系列と照合しリスコアリングを行い，スコア順に最終候補としてユーザへ提示する．

ステップ 2 で作成する事前検索結果は，候補区間をスコア順にソートした後に発話 ID のみを保持する．ステップ 4 の事前検索結果の参照で，一次候補区間の選出は発話 ID をキーとしたハッシュテーブルを構築することで高速に行う．ステップ 4 での一次候補区間の選出では，取りこぼしを抑えるため今回は論理和，すなわち抽出された区間すべてを候補区間としてステップ 5 に渡すこととした．

2.2.2 距離閾値による候補絞り込み

検索時間はステップ 5 の連続 DP の時間が大半を占めるため，その候補区間数を上位 K 件に削減することで検索時間の削減を図った．上位 K 件以内でも，類似度が低い候補は除外しても検索精度への影響は小さいと想定できる．そこで，本項では事前検索結果作成時の上位 K 件以内の候補を連続 DP の距離でさらに絞り込むことで，さらなる検索時間とインデックスサイズの削減を図る．ステップ 2 で，上位 K 件以内の事前検索結果を保持する際に，連続 DP の距離閾値を設け，閾値以上ならば上位 K 件以内であっても事前検索結果から除外する．

2.3 上位候補の高速抽出

STD の評価で一般的に用いられる MAP での評価は再現率 0%〜100%の平均の適合率であり，検索全体（一般に低再現率&高適合率〜高再現率&低適合率）での評価となっている．しかし，実システムとして考えた場合，最上位の候補群に正解が含まれているかが検索時間とともにユーザを満足させるうえでは重要であると考える．音声ドキュメントの検索ではテキスト検索とは異なり，候補群が正解かは見て判断できず，聞いて確認する作業が不可欠である．その確認作業の間に精密な検索を行うことができる．したがって，STD においてはすべての検索を高速に完了する必要は必ずしもなく，上位の候補区間に対してだけ高速に抽出できる方式が有効と考える．

2.2.1 項のステップ 4 での事前検索結果の参照件数は上位 K 件としたが，これを上位 K' 件（K' < K）とし，まず比較的小さい K' を用いることで最上位候補を高速に求め，ユーザがこれらの候補を確認する間に，より大きい K' の値で検索することで網羅的な検出を行う．このように，ステップ 4 の上位 K' 件を制御することにより，待ち時間なく上位候補から順に提示できると考える．

3. 評価実験

2.1 節の実験条件で評価実験を行い，提案方式が全照合と比べ精度を維持しながら検索時間を削減できることを示す．

3.1 音節 N-gram における N の設定

提案方式では，すべての音節 N-gram に対して事前検索結果を作成する必要がある．本論文では 261 種類の音節を用いているため，事前検索結果の作成に要する時間は $261^N \times$ (1 検索語あたりの検索時間) となり，N の値に対して指数関数的に増加する．コアタスク（1 検索語の検索時間：0.922 秒）において N = 3 とすると，約 196 日が必要となる．N = 1 であれば高速に事前検索結果を作成できるが，湧き出し区間が多くなり検索時の精度低下が予想される．N = 2 であれば約 18 時間で作成可能であるため，事前検索結果の精度と作成時間を考慮し，本論文では N = 2，音節 bigram で実験を行う．

Top-K	All	5,000	3,000	1,000	500	300
MAP(%)	75.62	75.62	75.48	75.04	73.70	69.90
Time(s)	0.922	0.404	0.278	0.113	0.062	0.036

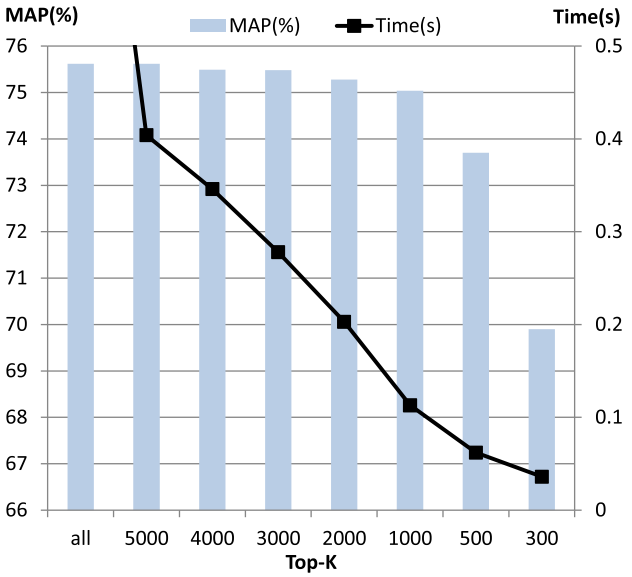

図 3 音節 bigram の事前検索結果を用いた検索精度（コア 177 講演）
Fig. 3 Performance using pre-retrieval results of syllable bigrams (for Core set including 177 lectures).

3.2 音節 bigram 事前検索結果利用時の検索精度と検索時間

音節 bigram の事前検索結果利用時の実験結果を，コア 177 講演について**図 3** に，全 2,702 講演について**図 4** に示す．図中，検索精度は棒グラフ（MAP），1 検索語あたりの検索時間は折れ線グラフ（Time）で表す．

図 3 から，コア 177 講演を対象とした場合，すべての音声ドキュメントを連続 DP で照合した全照合の場合（all）と比べ Top-K における K = 5,000 で検索精度の低下なしに検索時間を 0.922 秒から 0.404 秒と，57.6%削減できた．この結果より，コア 177 講演の場合 K = 5,000 以上の事前検索結果を保持する必要がないと判断できる．また，K = 500 では検索精度が 75.62%から 73.70%に 1.92 ポイント低下したが，検索時間は 0.922 秒から 0.062 秒と，all と比較し 93.59%削減できた．K = 300 では高速な検索ができるが，検索精度が 75.62%から 69.90%に 5.72 ポイント低下した．

K = 3,000 ではほとんど検索精度の低下なしに（低下は 0.14 ポイント），検索時間は 0.278 秒になり 70%以上の検索時間を削減できた．K = 1,000 では，検索精度の低下を 0.58 ポイントに抑えつつ，0.113 秒となり 88.14%の検索時間を削減できた．以上より，本方式は，検索精度を維持しつつ検索時間を大幅に削減可能であると分かる．

図 4 から，全 2,702 講演を対象とした検索を行う場合，全照合の場合（all）と比較し，K = 25,000 で同じ精度で 1 検索語あたりの検索に要する時間は 16.10 秒から 2.87 秒に減少し 5 倍以上の高速化を実現した．コア 177 講演と同様に，

Top-K	all	25,000	15,000	7,500	5,000	1,000
MAP(%)	66.37	66.37	66.12	65.86	64.35	54.16
Time(s)	16.10	2.87	1.88	0.964	0.649	0.156

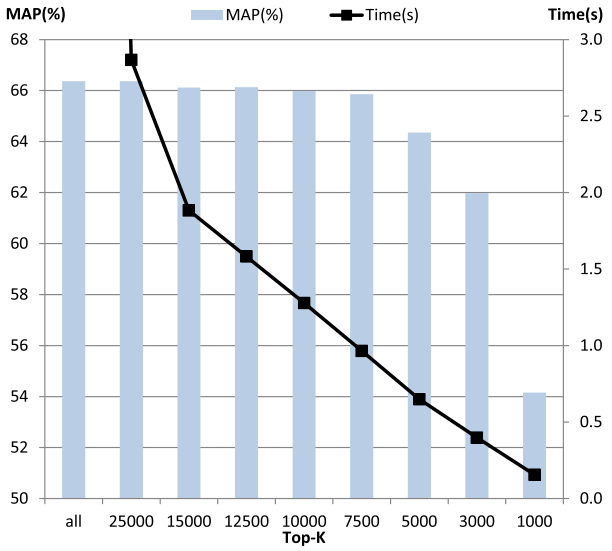

図 4 音節 bigram の事前検索結果を用いた検索精度（全 2,702 講演）
Fig. 4 Performance using pre-retrieval results of syllable bigrams (for All set including 2,702 lectures).

全 2,702 講演の場合は K = 25,000 以上の事前検索結果を保持する必要はないと判断できる．K = 15,000 のとき，検索精度の低下は 0.25 ポイント（66.37 → 66.12）で検索時間を 16.10 秒から 1.88 秒と 88.3%削減できた．K = 7,500 では検索精度の低下を 1 ポイント未満（0.51: 66.37 → 65.86）に抑えたうえで 1 秒以内（0.964 秒）の検索を実現でき，実用的な待ち時間と考える．全 2,702 講演を対象とした実験結果をまとめると以下のとおりとなる．

- K = 25,000 で検索精度低下なし，5 倍以上の高速化
- K = 7,500 で検索精度低下が 1.0 ポイント以下，1 秒以内（0.964 秒）の検索を実現

K ≦ 5,000 では検索精度の低下幅が大きくなっており，過剰に絞り込むことで正解も除外してしまったと考える．

検索精度の低下がない場合はコア 177 講演で K = 5,000，全 2,702 講演で K = 25,000 となった．両者を比べた場合，音声ドキュメントは 15.3 倍だが，K は 5 倍で済んだ．

事前検索結果を参照し一次候補区間を求める処理時間は K が大きい場合でも全処理時間の 10%未満で，連続 DP の照合時間が 9 割以上を占めた．3.5 節で述べるように処理時間は理論上候補数 K に比例する．実際に図 3，図 4 からも K に比例していることが分かる．連続 DP 処理を行う一次候補区間数の削減が検索時間の削減につながる．

3.3 事前検索結果の閾値による絞り込み

2.2.2 項で提案した距離閾値による絞り込みの評価を行う．閾値は 10.0，8.0，6.0，4.0 とした．K はコア 177 講演の場合，K = 1,000，3,000，5,000 について，全 2,702 講演

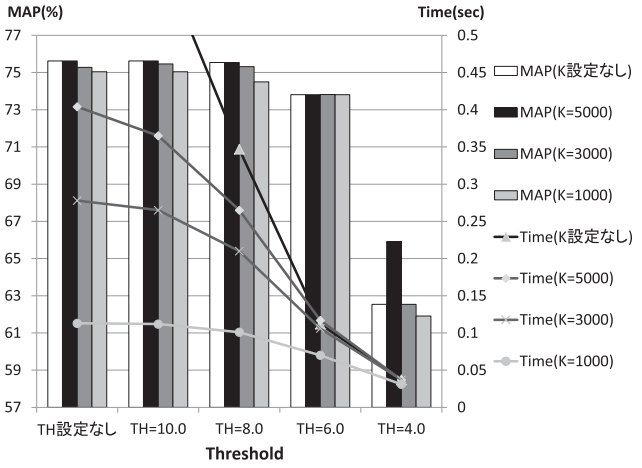

図 5 距離閾値を用いたときの検索性能（コア 177 講演）

Fig. 5 Performance using a threshold for Core set including 177 lectures.

の場合，K = 5,000，7,500，10,000 で実験を行った．コア 177 講演の検索精度と検索時間を**図 5** に示す．横軸は閾値（TH）を表し，検索精度を棒グラフで，検索時間を折れ線グラフで示す．

「K 設定なし」では閾値 TH 以下であればすべての候補を連続 DP の処理対象とし，「TH 設定なし」では閾値処理をせずに K 個の候補をすべて連続 DP の処理対象とする．「K・TH 設定ともになし」は全音声ドキュメントに対して連続 DP を行った場合（図 3，図 4 における all と同じ）である．図 5 で検索時間の折れ線が図から出ている 2 つの部分は，K・TH 設定ともになし（all）の検索時間が 0.922 秒，K 設定なし・TH = 10.0 の検索時間が 0.612 秒であった．

コア 177 講演の場合，距離閾値 10.0，K = 5,000 のときに検索精度低下なしで検索時間を 0.365 秒に削減できた．8.0 以上の閾値では検索精度の低下は小さく，6.0 で検索精度の低下が 2.0 ポイント弱，4.0 にすると検索精度の低下が大きくなった．検索精度の低下を厳しく抑えたうえで高速化を図ることを重視した場合は，閾値は 8.0 以上が望ましい．

K・TH 設定ともになしの all の場合は MAP 75.62%，検索時間 0.922 秒だったが，距離閾値 8.0 を導入した場合，以下のように検索時間の削減ができた．

- K = 5,000 で，75.54%（−0.08 ポイント），検索時間 0.265 秒（72.19%の削減）
- K = 3,000 で，75.32%（−0.30 ポイント），検索時間 0.210 秒（77.96%の削減）
- K = 1,000 で，74.50%（−1.12 ポイント），検索時間 0.101 秒（89.40%の削減）

これらの結果から，距離閾値を導入することで検索精度の低下を抑えながら検索時間の削減が可能であると分かる．

例として，以下の 2 つの結果を比較すると，(1) のように距離閾値単体で用いるより，(2) のように Top-K と併用することで検索精度の低下を抑えたうえで一次候補区間数のさらなる削減ができ，高速化につながったと考える．

(1) K 設定なし，閾値 6.0 で MAP 73.81%，検索時間 0.112 秒

(2) K = 1,000，閾値 10.0 で MAP 75.05%，検索時間 0.112 秒

全 2,702 講演を対象とした場合も，コア 177 講演と同様，閾値 8.0 までは検索精度の低下を抑えたうえで検索時間を削減できた．コア 177 講演を対象とした結果と比較すると，全 2,702 講演を対象とした方が距離閾値を導入した効果は小さかった．これは，全 2,702 講演検索では Top-K のみを用いた段階ですでに 9 割以上の検索時間を削減できており，また類似度が高い区間が増え，閾値で除外できる類似度の低い候補が少なくなったためと考える．

なお，TH = 4.0 のとき，K 設定なしの方が K = 5,000 よりも検索精度が低下した．前述したようにコア 177 講演の場合 K = 5,000 以上の事前検索結果を保持する必要がない．一方，いくつかの音節バイグラムは TH = 4.0 としても事前検索結果が数千以上（2 千以上が 39 種）となっており，これらの音節バイグラムの和をとると一時候補区間数が 5,000 を超える場合がある．5,000 位より低位の一時候補区間が湧き出し誤りを生成し MAP を低下させたと考える．

3.4 上位候補による評価

2.3 節で述べた上位候補の高速抽出方式を評価するため，比較的小さい K' によって求めた候補のうち，ユーザに最初に提示される候補となる上位 T 件での評価を行う．各検索語の上位 T 件に正解が含まれる件数を，50 個の検索語の平均で評価する．T は 1，3，5，10 とした．コア 177 講演検索時の結果を**表 3** に，全 2,702 講演検索時の結果を**表 4** に示す．表中の括弧内の数字は，T × 50 件の候補内の正解数を示している．

コア 177 講演を対象とした場合，図 3 の結果のとおり，K = 5,000 では全照合（all）と同等の精度が得られたが，K = 300 に制限した場合，精度の低下が大きかった．一方，上位候補では表 3 のように K' = 300 としても最上位（T = 1）は all から精度の低下なしに 50 個中 48 個の検索語で正解が得られた．K' = 500，T = 10 において 10 件中 4.84 個と約半数が正解を含んでおり，1 検索語あたりの検索時間は 0.1 秒未満（網掛け部）で非常に高速な検索が実現できた．

全 2,702 講演を対象とした場合，表 4 のとおり，K' = 1,000 のときでも最上位（T = 1）は all から精度低下なしに 50 個中 47 個の検索語で正解が得られた．K' = 3,000，T = 3 および K' = 7,500，T = 5，T = 10 で all と同等の検索精度で上位候補を高速にユーザに提示することができた．

コア 177 講演，全 2,702 講演どちらも T = 1 のとき，all で正解数が 47 個に対し，Top-K 設定時に 48 個となるケー

表 3 コア 177 講演：上位候補 T 件以内の平均正解数と検索時間（50 クエリの正解の総数）
Table 3 Average hit number in Top-T candidates and retrieval time for Core set including 177 lectures (total hit number for 50 queries).

Top-K'	T=1	T=3	T=5	T=10	Time(s)
100	0.88(44)	2.30(115)	3.06(153)	3.88(194)	0.013
200	0.92(46)	2.40(120)	3.34(167)	4.30(215)	0.025
300	0.96(48)	2.58(129)	3.56(178)	4.60(230)	0.036
500	0.96(48)	2.60(130)	3.64(182)	4.84(242)	0.062
5,000	0.94(47)	2.62(131)	3.68(184)	4.96(248)	0.404
all	0.94(47)	2.62(131)	3.68(184)	4.96(248)	0.922

表 4 全 2,702 講演：上位候補 T 件以内の平均正解数と検索時間（50 クエリの正解の総数）
Table 4 Average hit number in Top-T candidates and retrieval time for ALL set including 2,702 lectures (total hit number for 50 queries).

Top-K'	T=1	T=3	T=5	T=10	Time(s)
1,000	0.94(47)	2.54(127)	4.12(206)	6.94(347)	0.156
1,500	0.96(48)	2.58(129)	4.20(210)	7.16(358)	0.214
2,000	0.96(48)	2.64(132)	4.32(216)	7.44(372)	0.278
3,000	0.96(48)	2.70(135)	4.40(220)	7.44(372)	0.398
7,500	0.96(47)	2.70(135)	4.50(225)	7.82(391)	0.964
all	0.94(47)	2.70(135)	4.50(225)	7.78(389)	16.10

ス（下線部）があった．これは all では距離が同一となる区間が出現し，同一順位ながら上位に位置したためである．T = 10 で顕著だが，上位 T 件以内に含まれている件数はコア 177 講演よりも全 2,702 講演の方が多くなっている．2.1 節 (2) に示したように，コア 177 講演の平均正解数が 7.16 件に対し全 2,702 講演では 19.68 件で，音声ドキュメント中の正解数の差が要因と考える．T = 10 のとき，網掛け部の Precision は全 2,702 講演の方が高いが，コア 177 講演の Recall は 69.27%，F 値は 57.81%，2,702 講演の Recall は 39.73%，F 値は 51.49%となり，F 値では 2,702 講演が低くなっている．

上位候補の高速検索として，コア 177 講演の場合，上位 1 位については K = 300 のとき 0.036 秒で all と同精度，2,702 講演の場合も 1 位を 0.156 秒，3 位までを 0.398 秒，10 位までを 0.964 秒で all と同精度で高速に求めることができた．提案方式では，事前検索により K 個の候補が整合度の高い順にすでに求められているため，その順序で候補を抽出すれば上位候補が自動的に抽出される．これにより音声ドキュメントが 177 から 2,702 へ約 16 倍となっても上位候補を高速かつ高精度に検索することができた．

3.5 時間・空間計算量の検討

提案方式では，上位候補区間から提示すれば全 2,702 講演，604 時間の音声ドキュメントに対して 1 秒以内に 10 位までを精度低下せずに提示できることを示した．この方式であれば，音声ドキュメントの量に依存せずに上位候補を提示でき，有効な方式と考える．候補区間を発話 ID の整数に対応させると，1 つの候補は 4B（バイト）で保持できる．空間計算量は事前検索結果が大半を占め，4B × 音節 N-gram 数 × K となり，K（各音節 N-gram に対して保持する候補数）に比例する．検索時の時間計算量は，主に一次候補抽出時間と一次候補区間に対しての連続 DP の時間である．一次候補抽出では，L（検索語の音節数 −N + 1）個の音節 N-gram の配列を参照し，昇順に格納された K 個の発話番号を抽出する．これを発話番号順に各 2 回比較すればよく，時間計算量は 2LK である．検索語の音素数を約 2L とし，発話の実際の平均音素数が 28（正確には 28.36）であったので，連続 DP では 56L（2L × 28）の格子点上での計算が必要となる．一次候補区間数はクエリ中の N-gram に各 K 個の候補区間があるので $L \times K$ 程度となる（各 N-gram の候補区間の和集合であるので LK 個が上限数）．したがって連続 DP の計算量は $56L^2K$（56L × LK）となり，連続 DP の計算時間が一次候補抽出の計算時間の 28L 倍（今回 $3 \leqq L \leqq 10$）となり，時間計算量は連続 DP が主であることが分かる．連続 DP の時間計算量は $56L^2K$ と K に比例するため，本方式の時間計算量はほぼ K に比例する．

全照合と同精度となるとき，今回の事前検索結果に要する空間計算量は全 2,702 講演で K = 25,000 のとき，約 6.8 GB（4B × 音節 bigram 数 × K = 4B × 261 × 261 × 25,000：

音節数 = 261）であった．このほかの空間計算量はたかだか 0.04 GB（音響間距離：23.4 MB，一次候補区間：K = 25,000, L = 30 のとき 3.0 MB, 2,702 講演の triphone：2,4971,598 個，5.0 MB）であった．

前述したようにコア 177 講演 44 時間から全 2,702 講演 604 時間へ音声ドキュメントを 15.3 倍にしても，精度を維持する K は 5,000 から 25,000 へと 5 倍に抑えられた．音声ドキュメント量と精度を維持できる K の値を今回実験により示したが，精度を維持できる K の値は，音声ドキュメント量だけでなく，検索語の長さや候補が分布する順位，音声ドキュメントのサブワード認識精度にも影響されると考えられる．このため，理論的な検証は今後の課題とする．

3.6 今後の課題

現状では事前検索結果作成時に各音節 N-gram 間の時間位置/順序を考慮していない．これらの情報を利用することでさらなる一次候補区間の絞り込みができると考える．本論文では検索語を 1 音節ずつシフトすることで分割した音節 N-gram の検索語を生成している．6 音節検索語を重複しない 3 つの音節 bigram に分割し，ビタビアルゴリズム等で時間軸上の整合性を確保したうえで候補を絞り込む方式等の検討を行いたい．

現在，事前検索結果作成に時間を要しており，これはすべての音節 N-gram の組合せを検索しているためである．実際には使用しえない音節 N-gram（ex. ををを，ををを）があり，今後は大規模テキストコーパスを用いて事前検索に必要な組合せ数等の調査を行いたい．

本論文で評価実験に用いた音声ドキュメントは CSJ のみだが，検索対象の音声ドキュメントの種類や規模によって適切な K や閾値は異なると考えられる．そのため，今後は音声ドキュメントの種類，規模に応じた適切な K や閾値の設定法を考えていきたい．

4. おわりに

本論文ではあらゆる組合せの音節 N-gram で事前に音声ドキュメントを検索しておき，その事前検索結果を利用することで STD の精度を維持しつつ高速な検索を実現する方式を提案した．各音節 N-gram に対して事前検索の候補数を上位 K 件までに制限することおよび連続 DP の閾値を導入することで，事前検索結果の候補を絞り込み検索時間の削減を図った．

音節 bigram による事前検索により，コア 177 講演を検索対象とした場合，検索精度の低下なしで検索時間を 0.922 秒から 0.404 秒に（2.28 倍），検索精度の低下 1.0 ポイント未満では検索時間を 0.922 秒から 0.113 秒（8.16 倍）の高速化を実現した．また，全 2,702 講演検索を対象とした場合，検索精度の低下なしで 16.10 秒から 2.87 秒（5.61 倍）に，検索精度の低下 1 ポイント未満では 16.10 秒から 0.964 秒（16.7 倍）の高速化を実現した．

上位候補の高速検索として，コア 177 講演の場合，上位 1 位については 0.036 秒で全照合と同精度を，2,702 講演の場合も上位 1，3，5，10 位いずれについても全照合と同精度を 1 秒未満で実施することができた．これにより，本方式が検索精度を維持しながら高速化できることを確認した．

本論文では音節 N-gram に対して N = 2 での評価を行った．我々はすでに擬似的な音節 trigram の作成も行っており [14]，音節 trigram での評価や音節 N-gram 間の時間情報を用いた方式の検討を行っていく予定である．

謝辞 本研究の一部は文部科学省学術研究助成基金助成金基盤研究（C）No.24500124 を受けて実施された．

参考文献

[1] National Institute of Standards and Technology: The Spoken Term Detection (STD) 2006 evaluation plan (Sep. 2006).
[2] Akiba, T., Nishizaki, H., Aikawa, K., Kawahara, T. and Matsui, T.: Overview of the IR for Spoken Document Task in NTCIR Workshop, *NTCIR-9 Meeting* (2011).
[3] 岩田耕平，伊藤慶明，小嶋和徳，石亀昌明，田中和世，李 時旭：語彙フリー音声文書検索方式における新しいサブワードモデルとサブワード音響距離の有効性の検証，情報通信学会論文誌，Vol.48, No.5, pp.1990–2000 (2007).
[4] 小野寺悠二，伊藤慶明，小嶋和徳，石亀昌明，田中和世，李 時旭：複数のサブワード・言語モデルを用いた音声中の検索語検出の高精度化，第 4 回音声ドキュメント処理ワークショップ講演論文集 (2010).
[5] 名取 賢，西崎博光，関口芳廣：任意語彙発話音声検索のための複数の認識モデルを利用した音節遷移ネットワークの構築，日本音響学会 2009 年秋季研究発表会講演論文集，1-R-27, pp.205–206 (2009).
[6] Wallace, R., Vogt, R. and Sridharan, S.: Spoken term detection using fast decoding, *ICASSP*, pp.4881–4884 (2009).
[7] Pinto, J., Szoke, I., Prasanna, S.R.M. and Hermansky, H.: Fast Approximate Spoken Term Detection from Sequence of Phonemes, *SIGIR '08 Workshop*, pp.28–33 (2008).
[8] 中川聖一，岩見圭祐，藤井康寿，山本一公：連続音節認識結果の距離つきトライグラムアレイ化による未知語音声の超高速検索，第 4 回音声ドキュメント処理ワークショップ講演論文集 (2010).
[9] 岩見圭祐，山本一公，中川聖一：複数音声認識システムを併用した音節 n-gram 索引による検索性能の改善，第 6 回音声ドキュメント処理ワークショップ，SDPWS2012-10 (2012).
[10] Katsurada, K., Sawada, S., Teshima, S., Iribe, Y. and Nitta, T.: Evaluation of Fast Spoken Term Detection Using a Suffix Array, *INTERSPEECH*, pp.909–912 (2011).
[11] 神田直之，住吉貴志，小窪浩明，佐川浩彦，大淵康成：多段リスコアリングに基づく大規模音声中の任意検索語検出，電子情報通信学会論文誌 D，Vol.J95-D, No.4, pp.969–981 (2012).
[12] Miller, D.R.H., Kleber, M., Kao, C.-L., Kimball, O., Colthurst, T., Lowe, S.A., Schwartz, R.M. and Gish, H.: Rapid and accurate spoken term detection, Interspeech, pp.314–317 (2007).
[13] 中野拓也，伊藤慶明，小嶋和徳，石亀昌明，田中和世，李 時旭ほか：音声中の検索語検出における triphone モ

デル集約方式の検討，第 5 回音声ドキュメント処理ワークショップ，SDPWS2011-08, p.6 (2011).
[14] 斉藤裕之，伊藤慶明，小嶋和徳，石亀昌明，田中和世，李時旭ほか：複数音節の事前検索結果に基づく音声中の検索語検出の高速化，日本音響学会 2012 年春期研究発表会論文集 3-7-10 (2012).
[15] http://julius.sourceforge.jp/

伊藤 慶明 （正会員）

平成元年東京大学大学院工学系研究科航空学専攻修了．同年川崎製鉄（株）に入社．平成 4 年より技術研究組合新情報処理開発機構に出向．音声認識対話システムの研究に従事．平成 7 年川崎製鉄（株）に復帰．平成 12 年岩手県立大学助教授．平成 25 年同大学ソフトウェア情報学部教授，博士（工学）．人工知能学会，日本音響学会，電子情報通信学会，IEEE 各会員．

斉藤 裕之

平成 23 年岩手県立大学ソフトウェア情報学部卒業．平成 23 年同大学ソフトウェア情報学研究科博士前期課程修了．現在，三菱電機スペース・ソフトウェア（株）に勤務．

田中 和世 （正会員）

昭和 45 年横浜国立大学工学部卒業，昭和 46 年通商産業省電子技術総合研究所入所，同研究所音声研究室長，総括主任研究官等を経て，平成 13 年産業技術総合研究所研究グループ長，平成 14 年図書館情報大学教授，平成 14 年 10 月より筑波大学教授．共著『音声工学』（森北出版）等．電子情報通信学会，日本音響学会，人工知能学会，IEEE 各会員．工学博士．

李 時旭

平成 9 年韓国嶺南大学 M.Sc.（音声認識研究）．平成 13 年東京大学大学院工学系研究科情報通信工学専攻博士課程修了（工学博士）．同年産業技術総合研究所入所．現在，同研究所情報技術研究部門研究員．日本音響学会，韓国音響学会各会員．