山梨大学大学院医工農学総合教育部情報機能システム工学専攻

平成 30 年度博士論文

高精度な音声認識結果の推定技術とその応用に関する研究

氏名　　澤田 直輝
指導教員　鈴木 良弥 教授
　　西崎 博光 准教授
修了年月　2019 年 3 月

高精度な音声認識結果の推定技術とその応用に関する研究

山梨大学大学院
医工農学総合教育部
博士課程学位論文

修了年月 2019年3月

氏名　　澤田 直輝

2018 年度 山梨大学大学院医工農学総合教育部情報機能システム工学専攻
博士論文公聴会及び最終審査にて発表済み
公聴会開催日：2019 年 2 月 6 日
開催場所： 山梨大学医工農学総合教育部情報機能システム工学専攻内
主催： 山梨大学

高精度な音声認識結果の推定技術とその応用に関する研究
論文要旨

近年，動画コンテンツに代表されるマルチメディアコンテンツが充実してきた．その理由に，Graphics Processing Unit（GPU）の発達による個人のマルチメディアデータの生成・編集の普及，Hard Disk Drive（HDD）や Solid State Drive（SSD）などのストレージの大容量化，YouTube や Twitch などに代表されるマルチメディアコンテンツの配信サイトの増加などが挙げられる．また，ビジネスにおいても，会議や講演や，病院のカルテの作成などに，映像の録画や音声の録音を用いることも増えてきている．このようなコンテンツが増加してきている背景には，ネットワークインフラの充実に加えて，タブレット PC，スマートフォンの普及により，容易にマルチメディアコンテンツにアクセスすることができるようになったことが挙げられる．

また近年では，深層学習技術が発展してきており，画像処理の分野や音声処理の分野で様々な手法が提案されている．このような深層学習技術の発展には，ハードウェアの発展と対応したコンテンツが充実してきたことが挙げられる．

音声処理の分野では，大量の音データを効率良く扱うために音声の内容から人間が求めているコンテンツを検索する技術や，人間が話している内容を理解して会話を行う音声対話システムなどの技術が提案されている．これらの技術の基盤技術として音声認識技術が存在する．音声認識技術は，音声データに対して発話内容を文字列として書き起こす技術である．音声認識技術を活用することで，発話内容を理解することや特定の内容を探すことが容易になる．

このような音声認識技術を活用する場合に，正しく音声認識できていない認識誤り箇所が様々な悪影響を及ぼす．このような認識誤りに対して，様々な認識誤り対策が考えられている．例えば音声検索では，単語より小さい音の単位であるサブワード単位で検索する方法や，認識結果の出現確率を用いて検索する方法などが用いられている．

また，音声認識技術においても，様々な新しい手法が提案されており，音声認識性能の改善が図られている．例えば音声認識システムにおいて，深層学習を用いて高精度な音素識別を導入することで認識性能の改善が試みられている．

しかし，音声認識システムは通常単語を認識する必要があるため，音声認識システムで学習されていない単語（未知語）を正しく認識することは困難である．また，認識誤りした結果は音声認識を用いた技術において性能低下の要因となってしまう．そこで，後処理で認識誤りを含む認識結果を正しいサブワード系列に変換することが有用であることが明らかとなっている．

さらに，先行研究では，文字列の表現を変更した複数の音声認識システムを用いることが，検索技術において認識誤りに頑健であることが示された．そこで，複数の認識システムを用いることで認識誤りを修正することができる高精度な正解音識別が可能であると考えられる．

本研究での目的は，音声からの検索などの応用技術に利用することが可能であり，様々な音声認識技術にも適用することが可能な方法で音声認識性能を改善することである．具

体的には，音声認識結果のサブワード系列を入力情報として，正しいサブワード系列を推定することによりそのまま応用技術に入力することが可能な出力を獲得することが実現する．本研究では，このような音声認識結果から正しいサブワード系列を獲得する高精度な認識結果推定器を提案する．さらに，この認識結果推定器が応用技術に有用であるか検証するために，検索実験と単語変換実験を行い性能の調査を行った．

本研究ではサブワード単位として音声の最小単位である音素を用いて高精度な認識結果を推定する．認識結果を推定するために深層学習技術を利用して正しい音素列の推定を行う．具体的には，複数の音声認識結果を音素列に変換し，各音素列を時間情報に基づいてアライメントを行う．このアライメントされた結果に対して深層学習技術を適用して各アライメント区間に対して正しい音素列の推定を行う．このような正解音素推定器を用いて，正しい音素列を推定することにより，元の音声認識列や複数の認識結果を多数決により統合した方法よりも，高い性能が得られた．このような結果から，正解音素推定器により高精度な認識結果を生成できる可能性が示された．

しかし，音声は，時系列を持ったデータであり，音と音が時間の軸で組み合わさることで意味が付いてくる．また，時間方向において制約も存在する．例えば，日本語において子音の後に母音が現れることは絶対であり，子音の後に子音が現れることはない．しかし，時間情報を用いなければこのような制約を考慮することができない．そのため，単純な正解音素推定器では，このような間違いが出現する可能性がある．このことから，正解音素推定器に対して時間情報を付与することは高精度な推定には必要である．

そこで，時間情報を考慮することができる深層学習技術を利用する．時間情報を考慮した正解音素推定器は，考慮していない正解音素推定器と比較して高い性能が示された．このことから，正解音素推定には時間情報が有用であることが示された．

正解音素推定器は，認識結果の誤りを減少させた高精度な認識結果を生成することができるため，音声認識を用いた様々なアプリケーションに適用可能である．そこで，正解音素推定の性能が音声認識を用いたアプリケーションにおいてどのような影響を与えるか調査した．

まず，音声の検索技術である音声中の検索語検出の技術に適用した．音声中の検索語検出は，音声中に存在する目的の単語を探し出す技術である．この技術は一般的に音声認識結果を利用するため，認識精度が検索精度に影響されやすい．そこで，正解音素推定結果から検索することで，高精度な音素推定性能が応用技術に好影響を与えるか調査した．結果として，推定性能が高い結果から検索するほど検索性能も改善の傾向が見られた．このことから，正解音素推定器が応用技術に適用することで，応用技術の性能を改善できる技術であることが示された．

しかし，実際に応用技術に音声認識結果を利用する場合，音声認識結果として単語列を入力することが一般的である．そのため，正解音素推定した結果を単語列に変換して認識結果の単語列の精度を改善することで，どのような応用技術にも適用可能にすることができる．そこで，認識結果の単語列の誤っている単語を，正解音素推定器で変換した単語で

置換することで誤りが少ない単語列を生成する．これは，認識結果において誤り単語箇所を検出し，誤り単語箇所に正解音素推定結果から変換して得られた単語を置換することで性能を改善させる．認識結果に対して適応することで，単語の認識結果が改善することが示された．このことから，正解音素推定器を様々な応用技術に適用することが可能であると考えられる．

今後の研究として，本研究の正解音素推定器は，アライメントにより時系列区間を固定した系列から正解の認識結果を推定している．そのため，アライメント精度による性能の低下が考えられる．そこで，複数の認識システムの時系列アライメントを深層学習技術により正解音素推定器に導入することでさらなる性能の改善が得られると考えられる．

本論文は以下の内容で構成されている．

第 1 章では，音声認識を改善させる先行研究を紹介し，本研究の概要について述べる．

第 2 章では，音声認識システムについて述べる．

第 3 章では，深層学習について述べる．

第 4 章では，深層学習を利用した正解音素推定器について述べる．

第 5 章では，時系列を考慮した正解音素推定器について述べる．

第 6 章と第 7 章では，正解音素推定結果を利用した応用技術について述べる．第 6 章では，音声中の検索語検出に対して正解音素推定器を適用できるか述べる．また，第 7 章では，推定結果に対して単語変換を行い，認識結果の単語列を正しい単語列に変換方法について述べる．

最後に，第 8 章で本研究をまとめる．

Study on High-Accurate Speech Recognition Result Estimation and Its Application

Abstract

Recently, multimedia contents typified by movies has been enriched. There are many reasons for this. First of all, the development and editing of personal multimedia data by the development of Graphics Processing Unit (GPU) can be mentioned as one. Next, the capacity of storage such as Hard Disk Drive (HDD) and Solid State Drive (SSD) is increased. The increment of multimedia contents distribution website such as YouTube and Twitch, and so on.

In recent years, deep learning techniques have been developed, and various methods have been proposed in the field of image processing and the field of speech processing. The development of such a deep learning technique is that contents corresponding to the development of hardware are enriched.

In the field of speech processing, systems based on speech recognition technology have been proposed. For example, in order to efficiently handle a large amount of sound data, a technique of searching for contents that a human is requesting from the contents of a voice has been proposed. Other techniques have been proposed for spoken dialog systems that understand the content that people are talking about and conduct conversations. Speech recognition is a fundamental technology for transcribing speech content as a character string on speech data. It is easy to understand the content of utterance by speech recognition technology and to search for specific contents.

When considering such a technique, a portion of recognition error which correctly recognizes speech has various adverse effects. Therefore, there are various method has been proposed to tackle the problem of recognition error. For example, in the field of speech retrieval, one is processing in subword units that are units of sounds smaller than words. The other is the method of using the occurrence probability of the recognition result.

To improve speech recognition performance, many new methods are also proposed for speech recognition. For example, recognition performance has been improved by introducing deep learning based speech recognition system.

However, since speech recognition generally recognizes as words, it is difficult for the recognition system to recognize words that are not learned. In addition, recognition errors are the fact of low performance in speech recognition technology.

Furthermore, in the previous research, it was shown that using multiple speech recognition systems is robust against recognition errors in search technology. Therefore, it is possible to discriminate correct phones with high performance that can perform the correction of recognition errors by using multiple recognition systems.

The purpose of this research is to improve speech recognition performance. In this study, we propose a high accuracy estimator that estimates the correct subword sequence

from the speech recognition result. This research can be used for applied technology such as speech retrieval and applied to various speech recognition technologies. We estimate the correct subword sequence by using the subword sequence of multiple speech recognition results as input information. It is possible to get an output that can be used as input in the applied technology using speech recognition. Furthermore, in order to verify the estimator is useful for an application using speech recognition, I experiment to measure the performance of speech retrieval and word conversion.

In this research, we will estimate the recognition result with high precision by using phoneme as a subword. A phoneme is a smaller unit of speech. In order to estimate the new recognition result, I use the deep learning technique to estimate the correct phoneme sequence. Especially, the results of multiple speech recognition are converted into a phoneme sequence. Next, each phoneme sequence is aligned based on time information. We estimate the correct phoneme sequence for each alignment interval using deep learning technique.

By correcting the phoneme sequence using the correct phoneme estimator, I obtain higher performance than the original speech recognition sequence and the method combining multiple recognition results by majority decision. As a result, it was shown that it is possible to generate a highly accurate recognition result by the correct phoneme estimator.

However, speech is a time series of data. Therefore, there is a relationship by combining sound and sound on the time axis. There is also a constraint in the time direction. For example, it is true that vowels appear after consonants in Japanese, no consonants appear after consonants. However, the estimator may output a consonant after a consonant without using time-series information. Therefore, with a simple correct phoneme estimator, there is a possibility of making an error. In order to reflect constraints, it is necessary to estimate high performance to give time-series information to the correct phoneme estimator.

Therefore, we use a deep learning technique which can make use of the time-series information. The correct phoneme estimator makes use of the time-series information has shown higher performance than the correct phoneme estimator without using time-series information. In this result, it was shown that time-series information is useful for correct phoneme estimation.

The correct phoneme estimator can be applied to various applications using speech recognition. This is because it is possible to generate a highly accurate recognition result that reduces errors in recognition results. Therefore, we experiment whether the performance of technology using speech recognition can be improved by using correct phoneme estimator.

First of all, we experiment spoken term detection using the correct phoneme estimator. Spoken term detection is a technique of finding a target word existing in a speech. A

spoken term detection method generally uses speech recognition results. Therefore, the search accuracy is easily affected by recognition accuracy. Then, I experiment whether highly accurate phoneme estimation performance has a positive influence on the application using speech recognition. As a result, an improvement of speech retrieval achieved with higher estimation performance. Therefore, it showed improving the performance of the application using speech recognition by using the correct phoneme estimator.

However, in fact of the application using recognition result, speech recognition results are generally represent as word sequences. For this reason, the result of correct phonemes estimation is converted into word sequence. As a result, the accuracy of the word sequence of the recognition result is improved, and it can be applied to any application. Therefore, by replacing the recognition error word of recognition result with the word converted by the correct phoneme estimator, a word sequence with less error is generated. An error word of a recognition result is detected, and then the error word is replaced with the converted word from the correct phoneme estimation result. As a result, it was shown that the recognition result improves by using for word recognition result. From this result, it is considered that it is possible to apply the correct phoneme estimator to a various application using speech recognition.

For the future study, we will improve the performance of correct phoneme estimator. The correct phoneme estimator estimates the new recognition result of the corrected sequence with the fixed time sequence alignments. Therefore, it is considering that the estimation performance is affected by the alignment performance. Therefore, a significant improvement in the speech recognition result can be expected by introducing the deep learning based correct phoneme estimator with multiple recognition alignment tasks.

The remainder of this paper is organized as follows.

In Chapter 1, I will introduce the previous research to improve speech recognition and describe the outline of the study.

In Chapter 2, I describe the speech recognition system.

In Chapter 3, I describe deep learning.

In Chapter 4, I describe correct phoneme estimator using deep learning.

In Chapter 5, I describe correct phoneme estimator considering time sequence.

In Chapter 6 and Chapter 7, I describe the application using the correct phoneme estimation result. In chapter 6, I describe whether a correct phoneme estimator can be applied to speech retrieval. In chapter 7, I describe converting the word into the estimation result and how to convert the word sequence of the recognition result into the correct word sequence using correct phoneme estimator.

目 次

図 目 次

表 目 次

第1章　緒言

1.1　研究の背景と目的

近年，動画コンテンツに代表されるマルチメディアコンテンツが充実してきた．その理由に，Graphics Processing Unit（GPU）の発達により個人によるマルチメディアデータの生成・編集の普及，Hard Disk Drive（HDD）や Solid State Drive（SSD）などのストレージの大容量化，YouTube[1]や Twitch[2]などに代表されるマルチメディアコンテンツの配信サイトの増加などが挙げられる．また，ビジネスにおいても，会議や講演や，病院のカルテの作成などに，映像の録画や音声の録音を用いることも増えてきている．このようなコンテンツが増加してきている背景には，ネットワークインフラの充実，タブレット PC，スマートフォンの普及により，容易にマルチメディアコンテンツにアクセスすることができるようになったことが挙げられる．

また近年では，深層学習技術が発展してきており，画像処理分野や音声処理分野で様々な手法が提案されている．深層学習は生物の脳を参考にしており，その発想自体は古くから提案されている．しかし，学習の計算コストが高く，実現するのが困難だと考えられていたが，近年の GPGPU に代表される計算機技術の発展により学習が可能となってきた．また，深層学習には大量の学習するためのデータが必要であり，マルチメディアコンテンツが充実してきたことも深層学習の技術の発展の要因の一つである．

音声処理の分野では，大量の音データを効率良く扱うために音声の内容から人間が求めているコンテンツを検索する技術や，人間が話している内容を理解して会話を行う音声対話システムなどの技術が提案されている．これらの技術の基盤技術として音声認識技術が存在する．音声認識技術は，音声データに対して発話内容を文字列として書き起こす技術である．音声認識技術を活用することで，発話内容を理解することや特定の内容を探すことが容易になる．

このような音声認識技術を活用する場合に，正しく音声認識できない認識誤り箇所が課題となる．例えば，検索技術においては認識誤り箇所が間違って検出されたり，音声対話システムが人間の話している内容を誤解してしまうといった課題が考えられる．このような認識誤りに対して，様々な認識誤り対策が考えられている．例えば，単語より小さい単位であるサブワード単位で処理 [1] を行ったり，認識結果の出現確率を用いたりする方法 [2]，複数の音声認識システムを用いる方法 [3] などが用いられている．

また，音声認識技術においても，様々な新しい手法が提案 [4, 5, 6, 7, 8, 9] されており，

[1]https://www.youtube.com/
[2]https://www.twitch.tv

音声認識性能の改善が図られている．例えば音声認識システムを，深層学習を用いて高精度な音識別を導入することで認識性能の改善がされている．

しかし，音声認識システムは単語を認識する必要があるため，音声認識システムで学習されていない単語（未知語）を正しく認識することは困難である．また，認識誤りした結果は音声認識を用いた技術において性能低下の要因となってしまう．そこで，後処理で認識誤りを含む認識結果を正しいサブワード系列に変換することが有用である．サブワードとは，単語より小さい単位である．例えば，平仮名に対応する音節や，音声の最小単位である音素などが挙げられる．音素とは，音声の最小単位のことであり，山梨という単語の音素列は/y a m a n a sh i/となり，スペースで区切られた文字列が音素に対応する．この音素の表現を用いることで，辞書に登録されていない単語でも表現することができる．

さらに，先行研究 [3] では，文字列の表現を変更した複数の音声認識システムを用いることが，検索技術において認識誤りに頑健であることが示された．これは，実際には音素という表現に変換している．さらに，複数の音声認識システムを用いることで各音声認識システムがそれぞれの音声認識誤りをカバーして正しい音素列も認識することができる．そこで，複数の音声認識システムの認識結果を音素列に変換しておくことで，認識誤りを修正することができる高精度な正解音素識別が可能であると考えられる．

本研究での目的は，音声からの検索などの応用技術に利用することが可能であり，様々な音声認識システムにも適用することが可能な方法で音声認識性能を改善することである．具体的には，音声認識結果のサブワード系列を入力情報として，正しいサブワード系列を推定することによりそのまま応用技術に入力することが可能な出力を獲得することができる．本研究では，このような音声認識結果から正しいサブワード系列を獲得する高精度な認識結果推定器を提案する．

1.2 関連研究

音声認識は様々な要素で構成されており，それぞれの性能を改善させるために様々な研究が行われている．まず，音声の特徴量を話者ごとに変換することで性能を改善させる手法 [10] が提案されいてる．また，音声認識において音情報を識別する音響モデルと単語の繋がり情報を持つ言語モデルが存在する．音響モデルだけにおいても音情報の識別性能を上げるために手法 [11, 12, 13] が提案されている．Povey ら [11] は，話者情報を音響モデルに付加することにより話者に特化した音響モデルを作成し性能を改善した．また，Hinton ら [12] は，音響モデルに深層学習を用いて性能を改善させた．言語モデルにおいても深層学習を用いた手法 [14, 15, 16] が提案されている．さらに，Srivastava ら [17] や Sailor ら [18] は，少資源言語と呼ばれる学習データが多く用意することができない音声に対して音声認識性能を改善する方法を提案している．少資源言語において学習データが少ないのは，正解のテキストを作成するためには労力がかかるため，マイナーな言語は正解テキストを作成することができず学習データを多く用意することができないからである．そのため，学習データ量が少ないままでもこのようなマイナー言語の認識性能を改善させ

る手法が提案されている．具体的な手法としては，このような低資源言語の音声認識システムを作成する場合に，深層学習を活用する場合に最適な構造などが提案されている．他にも，Hadian ら [13] は，音響モデルと言語モデルの結果を統合する際に音の持続時間を使用することで性能の改善を行った．また，近年では音声認識システムを深層学習を用いた End-to-End と呼ばれる構造 [19, 20, 21, 22, 23] により高精度な認識性能を実現している．End-to-End は音響モデルと言語モデルを深層学習を用いて単一のモデルで表現した構造のことを表している．本研究では，複数の音声認識システムの結果から正しいサブワード系列を獲得する研究であるため，このような音声入力から高精度な認識結果を獲得する手法とはアプローチが異なる．そのため，本研究ではこれらの研究成果を入力することで活用することでき，さらに高精度なサブワード系列を獲得できる手法であると考えている．

音声認識の後処理で認識精度を改善する効率的かつ代表的な手法として ROVER（Recognizer Output Voting Error Reduction）法 [24] がある．ROVER 法では，複数の音声認識システムの結果を，多数決を利用して統合することで認識性能を改善した．しかし，単純な多数決では音声認識システムごとの情報は活用していない．実際に音声認識結果は，音声認識システムごとに間違える箇所が異なることが多い．このことから音声認識システムごとに得意な音情報や言語情報が存在することが考えられる．そこで，多数決ではなく深層学習を用いて，それぞれの音声認識システムの情報を用いた統合を行うことでさらなる性能の改善できると考えられる．

また，深層学習を用いて音素分類を行う研究 [25] も存在する．この研究では，音声の特徴量であるフィルタバンク出力から音素を直接分類する研究である．しかし，音声認識分野において言語情報を用いたほうが，一般的に認識性能が高いことが一般的である．その点，本研究では，音声認識システムの結果を活用することで高い認識精度をより高精度な認識結果に変換することができる．

音声認識結果を利用した応用技術として，音声中の検索語検出や，対話システムなどが存在する．

例えば，音声中の検索語検出では検索性能を改善させるために様々な方法が提案されている．音声中の検索語検出は一般的に音声認識システムを用いるため，入力する音声認識結果の認識精度を向上させることで検索性能を改善する研究が存在する．特に複数の音声認識システムを用いて検索性能を改善する手法 [24][26][27] が多く提案されている．例えば，Fiscus ら [24] は，複数の音声認識システムを統合することで検索性能が改善させた．この研究では，複数の音声認識システムの結果を多数決を行うことで統合し，高い認識精度を実現している．これは，認識した音声認識システムの数が多いほどその認識結果が信頼できることができるという考えからきている．しかし，音声認識システムの特徴により，音情報や言語情報の得意不得意などが存在することが考えられる．そこで，本研究では，音声認識システムの組み合わせ方法に深層学習を導入することで高精度なサブワード認識結果の獲得を行う．実際に音声中の検索語検出の分野では，音素や音節のようなサブワードラティス [1] や，コンフュージョンネットワーク（Confusion Network）を用いた手法 [28] が提案されている．

また，対話システムにおいて性能を改善する様々な手法も提案されている．例えば，対

話システムにおいても誤認識に対策した手法 [29, 30] が提案されている．しかし，このような手法は以前の発話内容を保持しておき，発話のテーマにそぐわない内容を修正することで認識誤り対策を行っている．そのため，このような応用技術にも本研究を適用が可能である．

1.3 研究の概要

先行研究 [3] では，複数の音声認識システムの出力をサブワード単位で用いることで，音声認識システムの誤り認識や，音声認識システムに登録されていない未知語に対して，頑健な音声検索システムを提案している．この先行研究では，音声認識システムとして複数の音声認識システムを使用している．この複数の音声認識システムはサブワード単位に変換することで，音声認識システムの誤り認識結果や，音声認識システムに登録されていない必ず認識誤りする未知語に対して頑健な手法となっている．

そこで，本研究では複数の音声認識結果のサブワード単位である音素列から高精度な正しい音素列を推定する正解音素推定器を提案する．

本研究の目的は，音声認識誤りを含む音声認識結果を，音声認識システムを用いた応用技術に適用可能な方法で高精度な音声認識結果に変換することである．そこで本研究では，まず複数の音声認識システムの結果を音素列に変換する．この複数の音声認識システムの音素列を入力とし，深層学習を用いて正解音素列を推定することで高精度な音素列に変換させる．実際に，複数の音声認識結果を用いて正解音素列推定器の性能を評価した．音素の識別性能が，入力した音声認識システムの結果の 83.5%から 85.9%に改善することが分かった．また，音声認識システムを用いた応用技術として音声中の検索語検出を正解音素推定器の結果から行った．検索実験では，他手法を用いた推定結果からの検索と比較して精度が高い検索を行うことができた．また，正解音素推定器の推定結果から単語に変換する変換器で単語列に変換した．入力した音声認識システムの単語認識結果よりこの単語変換器により変換した単語変換器で変換した結果のほうが高い精度となることが分かった．これらの実験結果から，複数の音声認識結果を深層学習を用いた正解音素推定器により応用技術に利用可能な高精度な認識結果を作成できた．

1.4 本論文の構成

本論文は 8 章から構成されている．

本章に続く第 2 章では，本研究で用いる複数の音声認識システムについて述べる．

第 3 章では，深層学習について述べる．

第 4 章では，高精度な音素系列を推定するための深層学習技術を用いた正解音素推定技術について述べる．

第 5 章では，高精度にするために時系列を考慮した正解音素推定器について述べる．

第6章では，正解音素推定技術の応用方法として，正解音素推定結果からの音声中の検索語検索技術について述べる．

第7章では，異なる応用方法として，正解音素推定結果からの単語変換技術について述べる．

第8章では，本研究のまとめと今後の研究課題について述べる．

第2章　複数の音声認識システム

本章では，高精度な正解音素系列を推定するために用いた複数の音声認識システムについて述べる．

音声認識システムは，音声認識エンジンには同一のものを用い，後述する言語モデルと音響モデルの 2 種類のモデルを変更することによって，複数の音声認識システムを用意した．

言語モデルは形態の違いにより 5 種類，音響モデルは 2 種類，すなわち 2 つのモデルを組み合わせて 10 種類の音声認識システムとした．10 種類の音声認識システムのうち，6 つは平仮名認識システムである．これは，かな漢字表記では表記の違いで認識誤りになってしまうことも考えられるため，平仮名の認識システムにすることで表記の違いを考慮せず認識することができるためである．

先行研究 [3] では，音声中の検索語検出（Spoken Term Detection：STD）において 10 種類の音声認識システムにより単一の音声認識システムを用いた場合と比べて，音声認識誤りや未知語に対して頑健な検索を行えることが示されている．この知見から，音声認識システムが誤ってしまった場合にも正しい音素列を推定することが期待できる．

2.1　音声認識システム

音声認識システム [31] は，一般的には音声波形から声の特徴を抽出する音響分析部，音響モデルや言語モデル，単語辞書を参照しながらその特徴量を単語列に変換する音声認識デコーダから成る．

近年では，音声認識システムの構成要素を一つのモデルに集約した End-to-End 音声認識システム [19] が提案され，高い性能が示されている．本研究では，音声認識システムの認識結果の文字列に対して適用するため，どのような音声認識システムに対しても適用することができる．音声認識システム単体で高精度な認識結果に対して本研究を適用することで高精度な正解音素推定が実現できると考えられる．しかし，音声認識性能が低い認識システムに対しても本研究を適用することで認識性能の改善が期待することができる．そこで，本研究では音声認識性能が高くない認識システムに対して認識性能の改善ができるか確認するために，一世代前の認識システムを用いて性能の評価を行った．

本研究では，音声認識エンジンとして Julius rev. 4.1.3 を用いる（現時点での rev. は 4.5）．Julius とは，IPA「日本語ディクテーション基本ソフトウェアの開発」プロジェクト [32] から提供された大語彙連続音声認識エンジンである．

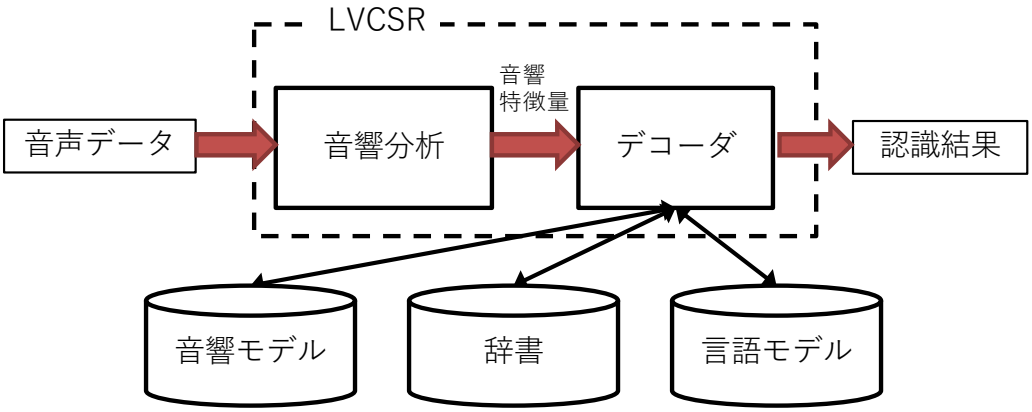

図 2.1: 音声認識システムの概要

2.2 音響モデル

音響モデル（Acoustic Model : AM）とは，音素などのサブワード（本研究では音素もしくは音節）の周波数パターンを保持しておき，統計的にどのサブワードに最も近いかを調査するために使用されるモデルである．この周波数パターンは，一般的に前後の音素を考慮した音素単位（これをトライフォンと呼ぶ）で保持しておく方法が取られる．このため，音声認識において音素列が音声を構成する最小単位となりえる．

音響モデルは，HMM（Hidden Markov Model）[33] でモデル化されるのが主流である．HMM は，時系列信号の確率モデルであり，複数の定常信号源の間を遷移することで，非定常な時系列信号をモデル化したものである．

HMM は，観測信号以外に状態を導入しており，観測信号は状態が出力した確率分布としたものである．HMM の状態は有限個となっており，状態を飛ばすことができない left-to-light 型の HMM が用いられる．また，HMM の各状態が出力する確率分布は一般的に混合ガウス分布が使用される．

このような，混合ガウス分布を確率分布として持つ HMM を用いた音節モデルを，GMM-HMM（Gaussian Mixture Model-Hidden Markov Model）モデル [34] と呼ぶ．

本研究では 2 種類の音響モデルを使用した．まず 1 つは，各音を日本語の平仮名 1 音に対応させてモデル化した音響モデル [35] である．そしてもう 1 つが，連続する 3 音素をモデル化したトライフォンモデルを使用した．

2.3 言語モデル

言語モデル (Language Model : LM) とは，ある 1 単語の後ろに統計的にどの単語が繋がる可能性が高いかを調査するために使用されるモデルである．統計的言語モデルとしては N-gram モデルが有名であり，本研究で使用する音声認識システムもこれを用いている．

以下では，本研究で用いる 5 種類の言語モデルの違いによる認識結果の差異について説明する．

形態素ベース言語モデル：Word-Base Characters (WBC)

形態素ベースの trigram モデル．形態素は，漢字と英数字，平仮名，片仮名で構成されている．学習に用いた形態素数は 約 27,000 語である．形態素は一般的な音声認識システムと同じ構成であり，一番言語的な繋がりを考慮することができる．

例：今回 / の / 実験 / の / 目的

平仮名形態素ベース言語モデル：Word-Base Hiragana (WBH)

単語ベースの trigram モデル．単語はすべて平仮名で構成され，元の単語に漢字や英数字，片仮名が含まれている場合には，すべて平仮名系列に変換される．すべての単語を平仮名に変換してあるため，同音異義語のような間違いが起きることがなく，言語的な繋がりも考慮することができる．

例：こんかい / の / じっけん / の / もくてき

文字ベース言語モデル：Character Base (CB)

文字ベースの trigram モデル．文字はすべて平仮名によって構成されている．平仮名の繋がりを考慮しているため，話し言葉の繋がりを考慮することができる．

例：こ / ん / か / い / の / じ / っ / け / ん / の / も / く / て / き

文字系列ベース言語モデル：Bi-Mora (BM)

文字系列ベースの trigram モデル．文字系列は 2 文字の平仮名によって構成されている．CB 同様に話し言葉の繋がりを考慮しているが，CB よりも言語的な繋がりが考慮することができる．

例：こん / かい / のじ / っけ / んの / もく / てき

疑似連続音節認識用言語モデル：Non

全てのモーラの出現確率を等しくした言語モデル．全てのモーラの出現確率が等しいことで，擬似的に連続音節認識を行うことが可能となる．言語的な制約が一切なく，最も音響的な系列を獲得することができる．

2.4 認識用単語辞書

認識用単語辞書とは，音響モデルと言語モデルのそれぞれに対して整合をとるために用いられる．

認識用単語辞書は語彙のエントリの表記と音素記号列からなる．例として，「山梨」という言葉を表すには，音素一つずつの表記であるモノフォンの場合は，/y a m a n a sh i/ と音素で表記するが，母音と子音をまとめた表記である音節の場合は/ya ma na shi/のように表記する．

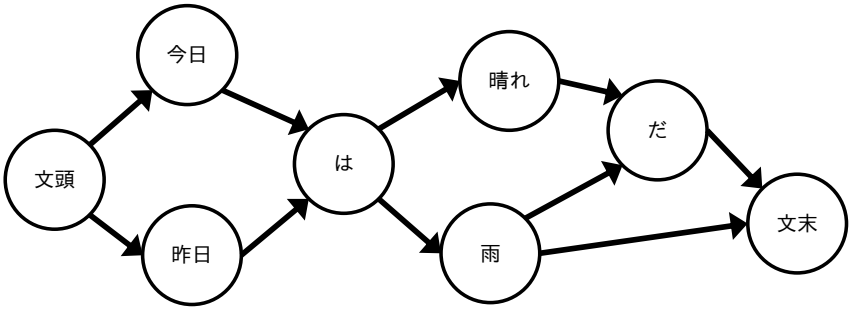

図 2.2: ラティスの例

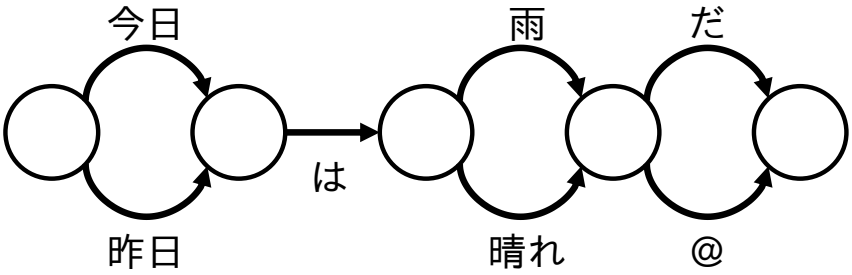

図 2.3: コンフュージョンネットワークの例

2.5 単一の認識結果の出力形式

音声認識システムを用いて音声認識を行うことで音声をテキスト情報に変換することができる．音声認識システムは最も確率が高い認識結果を出力するが，最も確率が高い認識結果が正しいとは限らない．

そこで，音声認識システムは複数の音声認識結果を出力することができる．この複数の結果を N-best 認識結果と呼ぶ．N は認識候補の数を表している．例えば，3-best の認識結果であれば，確率が高い上位 3 個の認識結果の文章が出力される．「今日は雨である．」という音声があった場合，認識候補として “今日は雨だ”，“今日は晴れだ”，“昨日は雨” の 3 文章が出力されることになる．

これに対して，複数の認識結果の候補を単語グラフ（ラティス）と呼ばれる形式で表すことができる．ラティスは，各単語がそれぞれどのくらいの重みで接続されるか表したグラフ形式の表現となっている．先ほどの 3-best でのラティスの例を図 2.2 に示す．また，各単語の接続の重みをネットワーク形式で表した表現をコンフュージョンネットワーク（Confusion Network）と呼ぶ（図 2.3）．これらの表現により，各単語間の重みを表し，どのような単語が候補として存在するか確認することができる．こちらも先ほどの 3-best での例を図 2.3 に示す．

2.6 複数の認識結果の出力形式

前節で説明した N-best 認識結果は，複数の認識結果を確率付きで並べることにより，作成することが可能である．

表 2.1: 認識用単語辞書の語彙数

認識用単語辞書種	奇数モデル	偶数モデル
認識用単語辞書 WBC	26,693	26,693
認識用単語辞書 WBH	19,953	19,953
認識用単語辞書 CB	262	262
認識用単語辞書 BM	12,120	12,407
認識用単語辞書 CSB	15,010	15,361
認識用単語辞書 Non	146	146

そのため，例えば ROVER 法 [24] は，複数の認識結果に対して認識された文字列で多数決を行うことで，多数認識された文字列を信頼できるとして確率を大きくする手法である．このように，多数の認識システムを用いることにより 1-best の認識性能が改善させることができることが示されている．

2.7 各モデルの学習条件

日本語話し言葉コーパス (Corpus of Spontaneous Japanese : CSJ)[36][37] は，学会講演 987 講演，模擬講演 1,715 講演の合計 3,302 講演で構成されている．これ以外に，「コア」と称する 177 講演（学会講演 70，模擬講演 107）約 39 時間のコア講演が存在する．

本研究に用いる音響モデルは，CSJ のコア講演以外の講演音声を用いて学習を行っている．

また，言語モデルの Non 以外のすべてのモデルは，CSJ のコア講演以外の講演音声を書き起こしたテキストから学習している．

なお，応用実験における STD の性能評価をオープンなデータで行うために，2010 年 5 月に公開された CSJ の日本語 STD 用テストコレクション [38] の音声認識条件に基づき学習，認識を行った．ただし，音声認識システムの学習に対して，全講演の認識環境をオープンにするために講演 ID が奇数と偶数で分けた．ここで，言語モデルの BM は認識用単語辞書が奇数モデルと偶数モデルで異なっている．BM 以外の言語モデルでは，作成した言語モデルの性質上，奇数モデルと偶数モデルの各認識用単語辞書の語彙数は同一となる．各言語モデルにおける語彙数は表 2.1 のようになっている．

また，CSJ のコア講演音声に対する音節ごとの認識率を表 2.2 に示す．このように，10 種類の認識システムのなかで言語モデルが “WBC”，音響モデルが “Tri” の組合せが最も認識性能が高いことが分かる．本研究において “WBC/Tri” の性能がベースラインとなる．

2.8 まとめ

本章では，音声認識システムと，音声認識システムの構成要素である音響モデルや言語モデル，単語辞書について述べた．

表 2.2: CSJ コア講演音声の音節認識率 [%]

LM / AM	Corr.
WBC/Tri	**86.46**
WBH/Tri	86.27
CB/Tri	81.83
BM/Tri	83.60
CSB/Tri	85.66
Non/Tri	71.00
WBC/Syl	79.11
WBH/Syl	79.32
CB/Syl	73.84
BM/Syl	77.89
CSB/Syl	78.58
Non/Syl	63.68

第 3 章では，深層学習について述べる．

第3章　深層学習

本章では，深層学習について述べる．

第2章では，本研究で使用する複数の音声認識システムについて述べた．

本章では，まず深層学習がどのような技術なのか述べる．次に，単純な構造な深層学習がどのように実現しているか述べる．そして，音声のような時系列データに対してどのような構造が必要なのかについて述べる．

3.1　深層学習とは

深層学習（Deep Learning）とは，脳神経を模したニューラルネットワーク [39, 40, 41] を重ねて多層にしたものである．ニューラルネットワークの隠れ層は入力データの特徴表現を持つことが知られている．これを多層化した Deep Neural Network（DNN）[42, 43] は，この特徴表現の幅がより広がり，その結果，入力データに対して表現豊かな（より識別能力の高い）特徴表現を持つことが可能となる．

この DNN の考え方は以前から存在した [44, 45, 46, 47]．しかし，DNN を学習するためには，莫大なデータ量が必要であり，当時の計算機の処理能力では，現実的なものではなかった．しかし，近年 GPU の性能の向上などにより，様々な研究分野において注目されている．

3.2　深層順伝播型ネットワーク

深層順伝播型ネットワーク（deep feedforward networks）は典型的な深層学習モデルである．順伝播型ネットワークの目的はある関数 f' を近似することにある．例えば分類では，$y = f'(x)$ は入力 x をカテゴリ y へ写像する．順伝播型ネットワークは写像 $y = f(x;\theta)$ を定義し，最もよい関数近似となるようなパラメータ θ の値を学習する．

このモデルは入力 x から f を定める中間的な計算を経て最終的な出力 y へと順に関数が評価されるため順伝播と呼ばれる．

順伝播型ネットワークは, 多くの異なる関数を組み合わせて表現される．順伝播型ネットワークのモデルは, 例として3つの関数 $f^{(1)}$, $f^{(2)}$, $f^{(3)}$ が繋がった $f(x) = f^{(3)}(f^{(2)}(f^{(1)}(x)))$ と構成される．このような複数の関数が繋がった構造がニューラルネットワークにおいて一般的な構造である．この例において，$f^{(1)}$ を1層目，$f^{(2)}$ が2層目となる．これらの層のことを中間層と呼ぶ．また，最後の関数である $f^{(3)}$ が出力層と呼ばれる．目的関数であ

る $f'(x)$ にニューラルネットワークの $f(x)$ を近づけるように訓練させるのが深層学習である．各事例にはラベル $y \approx f'(x)$ のように，入力データ x に対して出力層が何を出力すべきかを指定する必要がある．つまり，出力層は y に近い値を出力しなければならない．ここで，学習させる際に出力層以外の層が何を出力するかは指定しない．そのため，学習アルゴリズムが f' において y に近似した値を得るために出力層以外の層をどのように変化させるかを決定させる必要がある．

3.3 ネットワークのモデル化

深層順伝播型ネットワークを用いてタスクを解く場合，タスクを解くことができる関数 $y = f(x;\theta)$ を持つモデルを定義する必要がある．このとき，θ はパラメータであり，目標関数である $y = f'(x)$ に近づくようにパラメータを変更させる．

モデル $f(x;\theta)$ を線形モデルで表す場合，パラメータ θ を $\mathbf{W}$ と $\mathbf{b}$ であるとすると，以下の式のように表すことができる．

$$\mathbf{y} = \mathbf{W}^{\mathrm{T}}\mathbf{x} + \mathbf{b} \tag{3.1}$$

この $\mathbf{W}$ が重みパラメータであり，$\mathbf{b}$ がバイアス項と呼ばれるパラメータである．

ここで，ネットワークが 2 層存在するモデルを考えると以下の式になる．

$$\mathbf{y} = f^{(2)}(f^{(1)}(\mathbf{x})) \tag{3.2}$$

このモデルには，式 (3.1) より $\mathbf{h} = f^{(1)}(\mathbf{x};\mathbf{w};\mathbf{b})$ と $\mathbf{y} = f^{(2)}(\mathbf{h};\mathbf{W};\mathbf{c})$ が存在している．ここで，バイアス項をいったん無視して $f^{(1)}(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x}$ および $f^{(2)}(\mathbf{h}) = \mathbf{h}^{\mathrm{T}}\mathbf{W}$ とする．こうすることで，$f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{W}^{\mathrm{T}}\mathbf{x}$ となる．この関数は $\mathbf{w}' = \mathbf{W}\mathbf{w}$ とすると $f(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{w}'$ と表現することができる．つまり線形モデルを多層に重ねても一つの線形モデルで表すことができてしまう．

そのため，ネットワークのモデルには非線形関数を使用する必要がある．現在のほとんどのニューラルネットワークは学習したパラメータで制御されるアフィン変換を使用し，それに続いて活性化関数と呼ばれる固定された非線形関数を適用することで特徴量を表現している．ここで非線形関数としたモデルを，式で表すと以下のようになる．

$$\mathbf{y} = g(\mathbf{W}^{\mathrm{T}}\mathbf{x} + \mathbf{b}) \tag{3.3}$$

活性化関数 g はタスクにより様々な非線形関数が用いられる．

3.3.1 活性化関数

隠れ層は基本的に入力としてベクトル x を受け取り，アフィン変換 $z = W^{\mathrm{T}}x + b$ を行う．そして，要素ごとに活性化関数である非線形関数 $g(z)$ を適用する．

Rectified Linear Unit

Rectified Linear Unit（ReLU）は以下の式で表される式で定義 [48, 49, 50] される．

$$g(z) = \max(0, z) \tag{3.4}$$

ReLU は線形関数と非常によく似ているため最適化しやすい．線形関数と ReLU の違いは ReLU は定義域の半分で 0 で出力する点である．

ReLU の派生として $z_i < 0$ となる場合に 0 にせず傾き α で表現する活性化関数が存在する．Leakly ReLU[51] では，α を 0.01 という小さい値で固定した以下の式で表される．

$$g(z) = \begin{cases} z & (z >= 0) \\ \alpha z & (z < 0) \end{cases} \tag{3.5}$$

また，パラメトリック ReLU では，α を学習可能なパラメータとして扱う活性化関数である．

シグモイドとハイパボリックタンジェント

ReLU が提案されるまで多くのネットワークモデルはシグモイド関数で表現されていた．シグモイド関数の式は以下のように定義されている．

$$g(z) = \frac{1}{1 + e^{-z}} = \frac{\tanh(z/2) + 1}{2} \tag{3.6}$$

また，ハイパボリックタンジェント関数も活性化関数で使用されていた．

$$g(z) = \tanh(z) \tag{3.7}$$

シグモイド関数はハイパボリックタンジェントの式で表すことができ，この 2 つの活性化関数は非常に近い関係がある．

3.4 ネットワークの学習

ネットワークのモデルを学習するためには，損失関数とネットワークの出力表現を選択する必要がある．

3.4.1 損失関数

深層学習を行う場合に大事な要素として損失関数の選択が存在する．損失関数は，一般的には学習データとネットワークモデルの間を交差エントロピーを損失関数として用いる．

深層学習において最尤法を用いて訓練した場合，損失関数は単純に負の対数尤度になり，モデルの出力分布と学習データの分布の間の交差エントロピーである．この損失関数は以下の式で表すことができる．

$$J(\theta) = -\mathbb{E}_{x,y\sim\hat{p}_{data}} log(p_{model}(y|x)) \tag{3.8}$$

損失関数の具体的な形は $log(p_{model})$ の形式に応じてネットワークモデルごとに異なる．

この最尤推定から損失関数を導出する手法の利点はネットワークモデルごとにコスト関数を設計する必要がなくなることである．これは，モデル $p(y|x)$ を決めることで自動的にコスト関数が決定することができるからである．

3.5 誤差逆伝播法

入力 x，出力 y である順伝播型ネットワークの場合，入力された情報はネットワークを順方向に伝播されていく．これは，入力 x が最初の情報として各層にある隠れ層に流れていき，最終的に予測結果である $\hat{y}$ が出力される．この流れを順伝播と呼ぶ．学習している場合は，損失値である $J(\theta)$ が得られるまで順伝播を続ける．誤差逆伝播法 [52] は勾配を計算するために損失値からの情報をネットワークの逆向きに伝播させる手法である．

3.5.1 一般的な誤差逆伝播法

スカラー値として z の勾配をグラフ上でその先祖ノードの 1 つにあたる x に関して計算する場合を考える．最初に z に関する勾配を計算する．これは線形変換（ReLU）の場合は $\frac{dz}{dz} = 1$ となる．さらに，グラフ中の z の各親ノードに関する勾配は現在の勾配に z を生成した演算のヤコビ行列を掛けることで計算することができる．つまり，z の勾配を x に関して計算するには，現在の勾配に対して x に到達するまでヤコビ行列の掛け算することで算出することができる．また，逆方向に探索していく際に経路が 2 つ存在する場合には，複数経路の勾配を単純に足し合わせることで計算することができる．

3.6 時系列を考慮したニューラルネットワーク

単純な深層順伝播型ニューラルネットワーク構造では，ある時間のデータは独立しており他の時間における情報を用いることができない．

そこで，Recurrent Neural Network（RNN）は，時系列情報を持つデータを処理することができるネットワークである．本節では，最初に単純に時間情報を追加した RNN について説明する．そして，単純な RNN より長距離の情報を保持できるようにした Long Short-Term Memory（LSTM）[53] と，Gated Recurrent Unit（GRU）[54] について説明する．さらに，畳込み演算を用いた畳込みニューラルネットワークについて説明する．

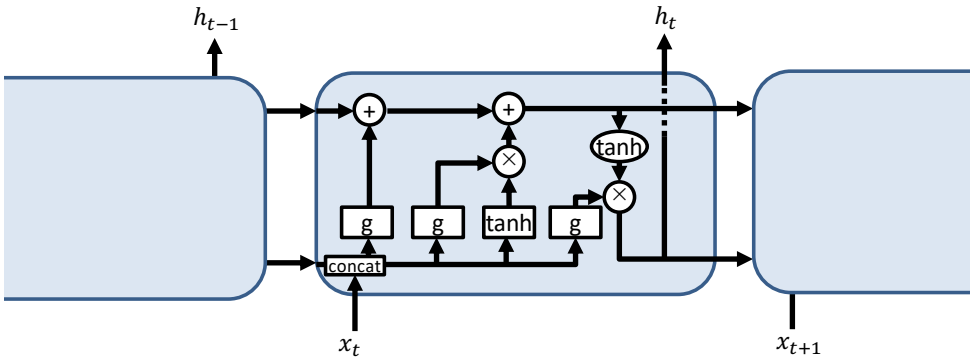

図 3.1: LSTM の構造図

3.6.1 単純な Recurrent Neural Network

単純な RNN は，中間層の入力として前の時間情報を追加することで実現できる．一般的な単純な RNN における隠れ層の式は以下のように定義される．

$$h_t = g(b + W^{\mathrm{T}} x_t + W_r^{\mathrm{T}} h_{t-1}) \tag{3.9}$$

前の時系列データを隠れ層の入力とすることにより，過去の履歴情報を用いることができるため時系列情報を扱うことができるようになる．

しかし，この構造では一つ前の時系列情報を入力しているだけであるため，長期間の時系列情報を扱うことができない．そこで，長い履歴情報を保持するために，LSTM や GRU などが提案されている．

3.6.2 Long Short-Term Memory

LSTM の構造は，入力と出力に加えて，前の出力を次の時系列に伝播する隠れ特徴量と，過去の時系列の特徴量を未来の時系列に伝播するセル特徴量が存在する．この隠れ特徴量とセル特徴量により長距離の時系列情報に考慮した深層学習を行うことができる．

LSTM の構造図を図 3.1 に示す．ここで，四角で示される図は活性化関数付きのニューラルネットワークであり，丸で示される図は関数である．また，LSTM の構造を表す式は以下のようになる．

$$\begin{aligned}
f_t &= g(W_f \cdot [h_{t-1}, x_t] + b_f) && (3.10)\\
i_t &= g(W_i \cdot [h_{t-1}, x_t] + b_i) && (3.11)\\
\tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) && (3.12)\\
C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t && (3.13)\\
o_t &= g(W_o[h_{t-1}, x_t] + b_o) && (3.14)\\
h_t &= o_t \odot \tanh(C_t) && (3.15)
\end{aligned}$$

ここで，x_t はある t 番目の系列的な入力ベクトル，o_t は出力ベクトル，C_t は時系列情報を保持した内部情報，h_t は時系列を考慮した出力である．また，g は活性化関数であり，

W_f，W_i，W_C，W_o は学習可能な変換行列であり，b_f，b_i，b_c，$_o$ は学習可能なバイアス項である．

3.6.3 Gated Recurrent Unit

GRU では，セル特徴量を除いた 3 つの構造を有しており，前の出力を伝播させるのみで時系列を考慮している．この方法により，学習が簡易化され，性能の改善される可能性が存在する．

GRU の構造は以下の式で表される．

$$r_t = g(W_r x_t + U_r h_{t-1} + b_r) \tag{3.16}$$

$$z_t = g(W_z x_t + U_z h_{t-1} + b_z) \tag{3.17}$$

$$\bar{h}_t = \tanh(W_x + U(r_t \odot h_{t-1}) + b_h) \tag{3.18}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \bar{h} \tag{3.19}$$

ここで，x_t はある t 番目の系列的な入力ベクトル，h_t は t 番目の出力ベクトルである．また，W_r，W_z，W_x，U_r，U_z は学習可能な変換行列であり，b_r，b_z，b_h は学習可能なバイアス項である．

3.6.4 双方向時系列の考慮

LSTM や，GRU といった履歴情報は，過去の情報しか用いられていない．しかし，時系列データを扱う場合に未来の情報を扱うことも有効であることが考えられる．例えば音声において，音声ファイルを録音した環境では未来に話している内容からも，今の音声の内容を推定に扱うことができるため有効であることが考えられる．

実際に双方向の時系列を扱うためには，2 つの時系列を扱うことができる構造を使用することで実現できる．式で表すと以下のように定義される．

$$h_t^f = \mathrm{ReccurentUnit}(x, h_{t-1}^f) \tag{3.20}$$

$$h_t^b = \mathrm{ReccurentUnit}(x, h_{t+1}^b) \tag{3.21}$$

$$h_t = [h_t^f, h_t^b] \tag{3.22}$$

式における $\mathrm{ReccurentUnit}(x, h)$ は，上記で説明した単純な RNN，LSTM，GRU などの時系列を考慮することができる構造のことである．このように，それぞれの時系列を扱う構造を別々に用意することにより双方向の時系列を扱うことができる．

3.6.5 畳込みニューラルネットワーク

畳込みニューラルネットワーク（Convolutional Neural Network : CNN）は，一般的に画像処理の分野で性能が示されたネットワーク構造である．しかし，この CNN をテキストといった時系列情報に適用 [55] することでも高い性能が得られることが示された．

CNN は，畳込み演算を持ったネットワーク構造のことを表す．CNN のパラメータとして，入力チャンネル数，出力チャンネル数，カーネルサイズ，ストライド，パディングが挙げられる．また，CNN から出力される 1 チャンネルあたりを特徴マップと表す．入力チャンネル数は，入力されるデータの系列の数を表す．画像処理における CNN を例とすると，一般的に RGB である 3 色の画像を扱うため最初の入力チャンネル数は 3 チャンネルとなる．出力チャンネル数は，特徴マップを何チャンネル出力するかを表す数値である．CNN におけるチャンネル数は，他のネットワーク構造における特徴量空間の大きさ（次元数）と対応していると言える．畳込み演算には，入力データと掛け合わせるカーネルと呼ばれる重み行列が存在する．このカーネルの大きさにより学習に影響する領域の広さが変わってくる．CNN においてカーネルを複数用意することにより多様な入力データに対応させている．また，一般的に入力データと比較してカーネルの大きさが小さいため，カーネルをどのくらいの間隔（ストライド）で適用させていくかを決める必要がある．パディングは，入力データに対する 0 埋め処理のことを表す．畳込み演算を行うと出力は一般的に入力と比較して特徴マップは小さいサイズとなってしまう．そこでパディングを行うことにより，入力マップを擬似的にサイズを大きくし出力する特徴マップのサイズを変わらない大きさにすることができる．

3.7 汎化性能改善のための技術

深層学習は，入力データに対するラベルを推定する技術である．このために，事前に大量の学習データを用意する必要がある．しかし，実際に使用する場合に，学習データと同じ環境で使用されるとは限らない．例えば音声の場合，学習データは静かな環境で収集したが，実際に使用する場面では騒がしい郊外で使用するということが存在する．このような場合に，ネットワークモデルは静かな環境でしか性能を発揮することができず，騒がしい環境では著しく低い性能となってしまう．

このため，深層学習において汎化性能は大事な性能の一つである．そこで，汎化性能を改善する手法として Dropout[56] が提案された．Dropout を含んだ構造の式は以下のように定義される．

$$r \sim Bernoulli(p) \tag{3.23}$$

$$\tilde{x} = r \times x \tag{3.24}$$

$$y = g(W^{\mathrm{T}}\tilde{x} + b) \tag{3.25}$$

$Bernoulli(p)$ はベルヌーイ分布のことを表し，p は 1 になる確率である．r は入力データ x と同じ大きさの 0，1 で構成されたベクトルである．ここで，演算子 $\times$ はベクトルの要素ごとの積を表す．こうすることにより，入力データ x が部分的に 0 に変換されることが分かる．このように，Dropout は接続するノードを削除する手法である．Dropout は入力を部分的にのみ扱うことでネットワークモデルの汎化性能を改善させることができる．

3.8　まとめ

本章では，深層学習について述べた．

具体的には，深層学習とはどのような技術なのかを述べた．次に，深層順伝播型ネットワークがどのように実現しているかを述べた．また，深層学習において時系列データ扱うためにどのような構造が存在するかを述べた．

第4章　Deep Neural Networkを用いた正解音素推定器

本章では，複数の音声認識システムを利用した深層学習を用いた高精度な正解音素推定器について述べる．

第3章では，深層学習について述べた．

本章では，深層学習を用いた正解音素推定について述べる．次に，深層学習を行うために音声認識結果をどのように扱うかについて述べる．さらに，実際に正解音素推定を行うためにどのような構造を用いるか説明する．そして，評価実験として正解音素推定器を用いて推定された音素列の精度を調査した結果について述べる．最後に実験結果として，正解音素推定器を用いた方が音素列の精度が改善したことを述べる．

4.1　正解音素推定

正解音素推定は，複数の音声認識結果から正しい音素列を推定する．この正解音素推定により，高精度な音声認識結果を生成する．高精度な音声認識結果を生成することにより，音声認識を利用した応用技術に有効な入力を作ることができる．

深層学習を用いた音素推定器の概要図を図4.1に示す．まず，各音声認識システムの認識結果の音素列を時間情報に基づいてアライメントをとる．アライメントには，DPマッチングを用いている．このとき，音声認識システムごとに大きく認識された文字列の長さが大きく異なる場合に，本来とは異なる場所の音素と一致してしまう可能性が存在する．そこで，音声認識結果の時間情報から大きく離れないようにアライメントを行う．具体的には，DPマッチングの距離コストには単純な編集距離を用いている．音素ごとのコストが同じコスト同士では，音声認識システムの認識した時間から一番近い時間列に配置している．このアライメント結果に対して正しい音素を出力するような正解音素推定器を，深層学習を用いて学習する．アライメントを行った認識結果に対して正解音素推定を行うことで，音声認識システムが誤ってしまっている場合でも，実際に話されている正しい音素を，DNNを用いて推定することができると考えた．

正解音素推定器は，10種類の音声認識システムの認識結果から学習する．アライメントした認識結果に対して，アライメントの時間情報をそのまま用いて正解音素推定器に入力する．そして，各アライメントに対応する正しい音素を出力するようにする．用いる音素の数は日本語の音素35種類を使用して分類を行う．実際の日本語の音素列は40種類程度あるが，この音素列には外来語にしか存在しない音素列もある．そのため，音声認識結

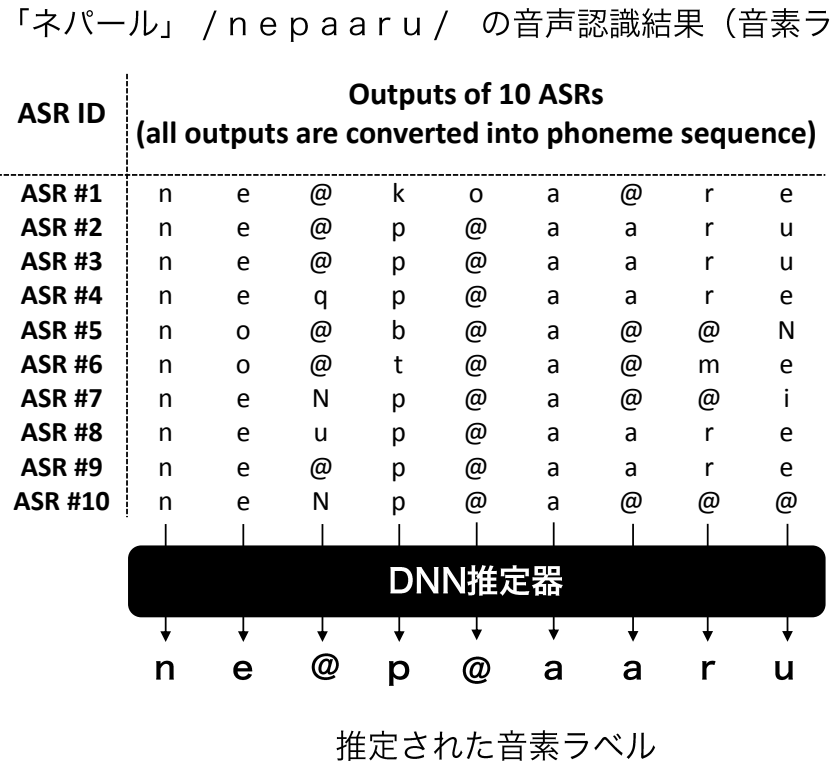

図 4.1: 正解音素推定器の概要図

果にはあまり現れない音素列が存在する．そこで，音声コーパスによく出現する 35 音素を使用することとした．

4.2　単純な正解音素推定器

深層学習を用いた正解音素推定器が，実際に正しい音素列を推定することができるのか調査した．

調査する正解音素推定器は，一般的には単純な構造を有する DNN 構造で性能の調査する．今回，一般的な DNN 構造として，中間層に全結合層のみを有した DNN を用いる．この単純な構造を図 4.2 に示す．

具体的には，まず，入力層が複数の認識システムの音素列であるため，文字列を入力しなければならない．そこで，各音素列を文字列に変換することを行う．この方法には，言語処理で用いられる Word Embedding[57, 58] の技術を用いて音素の文字列をベクトル列に変換する．Word Embedding では，各文字列（例えば，単語）が有する意味表現をベクトルに変換する技術のことである．この技術を用いることにより，各音素列がどの音素と類似しており間違えやすい音素列の情報をベクトルで表現することができる．本研究では，音素列に対して Word Embedding の技術を適用しているため，このベクトル変換を音素 Embedding と呼ぶ．また，本研究では複数の音声認識システムを用いるため，音声認識システムの数と同じ数の音素 Embedding を用いることですべての音素列をベクトル系列に変換する．

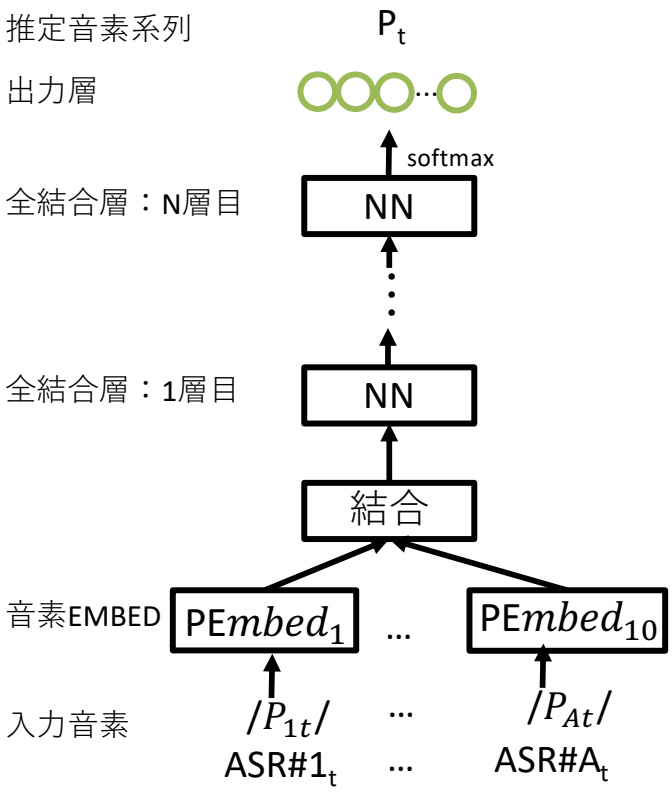

図 4.2: 単純な正解音素推定機の構造

音素 Embedding を複数使用することで，音声認識システムと同じ数のベクトル特徴量を獲得することができる．このベクトル表現を DNN に更に渡さなければならない．そこで，各ベクトル特徴量の結合処理を行う．この結合処理には，単純な方法で行われている．例えば，各ベクトル表現が 5 次元のベクトル特徴量であり，認識システム数が 10 種類であった場合に，50 次元のベクトル特徴量となる．

認識システム a に対応する音素 Embedding を $PEmbed_a$ としたとき，認識システムの結果からベクトル特徴量に変換する式は以下のようになる．

$$h^0_{at} = PEmbed_a(P_{at}; \theta_{ea}) \tag{4.1}$$

$$h^0_t = [h^0_{1t}, \cdots, h^0_{At}] \tag{4.2}$$

ここで，P_{at} は認識システム a のアライメント区間 t の音素を表す．また，h^0_{at} は認識システム a，アライメント区間 t の中間層のベクトル特徴量である．

音素 Embedding から得られたベクトル系列を，全結合層で処理を行っていく．全結合層は，深層学習の中で一般的な層の一つである．この全結合層は，入力されたベクトルに対して全ての次元を用いて新しいベクトル特徴量を生成する．この生成されたベクトル系列は，入力されたベクトル特徴量とは異なる空間の特徴量を持っている．そのため，全結合層の処理を繰り返していくことで分類や推定に有用な特徴量に変換することが可能である．

全結合層を通して中間層のベクトル特徴量を獲得する式を以下に示す．

$$h_t^l = ReLU(FC(h_t^{l-1} : \theta_{fl})) \tag{4.3}$$

また，h_t^l は l 層目の全結合層，アライメント区間 t の中間層のベクトル特徴量である．

そして，最後の全結合層から得られたベクトル特徴量を音素推定を行うために出力層で処理を行う．出力層の式を以下に示す．

$$O_t = Softmax(FC(h_t^L; \theta_o)) \tag{4.4}$$
$$\hat{P}_t = argmax(O_t) \tag{4.5}$$

ここで，O_t はアライメント区間 t の推定音素の出現確率であり，$\hat{P}_t$ はアライメント区間 t において推定された最も正解の可能性が高い音素を表す．

以上のような正解音素推定器を用いて，正しい音素列を推定することができるか調査する．

4.3 評価実験

4.3.1 正解音素推定器のハイパーパラメータ

正解音素推定機の中間層の全結合層の数は 7 とし，各層の活性化関数には ReLU を用いた．音素 Embedding は各 5 次元で合計 50 次元，中間層の各層は 512 次元で実験を行った．各層のパラメータの初期化方法には −0.1 から 0.1 の一様分布の乱数で初期化を行った．最適化手法には，確率的勾配降下法（Stochastic Gradient Descent : SGD）を用いた．また，Dropout を 20%で各層に行っている．

4.3.2 ベースライン

正解音素推定器が高い精度で音素を推定することができているか比較をするため，ベースライン手法として ROVER 法（“ROVER”）を用いて比較を行う．

ROVER 法は，複数の音声認識システムを統合させる手法である．ROVER 法の例を図 4.3 に示す．この例では，単語単位で複数の音声認識システムを統合させる例である．音声認識結果は，7 つの音声認識システムの認識結果が「今日は晴れです」，2 つの音声認識結果が「今日は晴れ模様です」，1 つの認識結果が「昨日は雨かも」という結果のラティス表現である．このとき，それぞれ最大のパスを通っていくので，「今日は晴れです」が ROVER 法では認識結果として選択される．このように複数の認識システムの結果において，認識システム数が多い認識結果を採用する手法が ROVER 法である．

本研究では，音素単位での高精度化を目指すため，アライメントを行った複数の音声認識結果に対して，多数決を行うことで正しい音素を選択させる．

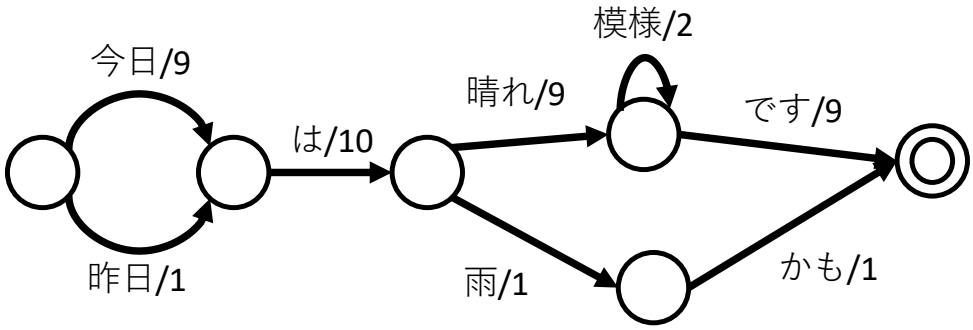

図 4.3: ROVER 法の例

4.3.3 データセット

正解音素推定器の学習データは，CSJ のコア講演を除く全講演の音声認識結果から学習している．音声認識システムには，2 章で説明した 10 種類の音声認識システムを用いて音声認識を行っている．

評価データには，2 つの音声コーパスを用意した．まず 1 つが CSJ（“CSJ set”）のコア講演音声 (39 時間の音声) であり，これは学習データには含まれてないがコーパスが等しいコーパスがクローズドな評価音声データとなる．もう一つが，音声ドキュメント処理ワークショップ（“SDPWS set”）の講演音声（28.6 時間）である．これは，コーパスもオープンな学会講演音声である．学会講演音声は，音声を収録するために話しているわけではないため，一般的に音声認識が困難となり認識は低めとなる．また，コーパスがオープンであることで未知の収録環境においても，正しく推定することができるか調査することができる．

4.3.4 正解音素推定の評価尺度

音素推定の評価には，音素正解率を用いる．音素正解率は，正解音素列をどれだけ検出することができたかを示している．つまり，音声認識の評価で利用される正解率（correct rate）とほぼ同等の評価指標となる．そのため，この尺度では発話単位での正解音素の検出性能の評価となる．音素正解率は，最大で 5-best までの出力を用いて評価する．DNN の正解音素推定器での 5-best の場合には，DNN の出力から得られる音素 posteriorgram[59] の出現確率が上位 5 位までの中に，正解音素が存在するかどうかを調査する．音素 posteriorgram とは，正解音素推定器の出力の事後確率系列（posteriogram）である．正解音素推定器はそのアライメント区間の音素の確率を出力する．これが音素 posteriogram である．

4.3.5 実験結果

まず，コア講演音声での認識性能の調査結果を表 4.1 に示す．参考として複数の音声認識の音素列の選択が理想的に行われた場合の音素正解率は 97.3% である．

表 4.1: コア講演音声の音素推定精度 [%]

N-best	1-best	2-best	3-best	4-best	5-best
単語音声認識	91.4	92.4	92.9	93.2	93.3
ROVER	90.3	96.0	96.8	97.0	97.0
DNN	91.5	96.1	97.0	97.6	98.0

表 4.2: SDPWS 音声の音素推定精度 [%]

N-best	1-best	2-best	3-best	4-best	5-best
単語音声認識	83.5	84.6	85.0	85.3	85.5
ROVER	82.1	90.9	92.1	92.4	92.4
DNN	84.8	91.1	94.0	95.5	96.1

単語音声認識結果と ROVER 法を比較すると，1-best は単語音声認識の方が高い性能が得られ，他の N-best では ROVER 法が高い結果となった．ROVER 法で 1-best の性能が低くなってしまった理由として，アライメントによって性能の低下を招いてしまったと考えられる．複数の音声認識を統合する場合にアライメントをすることで区間を決定している．そのため，アライメントがずれてしまうとその時点で性能が低下してしまうことが考えられる．つまり，アライメントを行わない単語音声認識よりもアライメントの性能低下によって ROVER 法が 1-best で性能が下回ったと考えられる．

各ベースラインと比較して，正解音素推定器の精度は各 N-best において高い性能となった．このことから，深層学習を用いて正解音素推定器を用いることで正しい音素系列を生成することができる．

また，単語音声認識と比較して正解音素推定器を用いた結果の性能が改善したか検定を多なった．これには，音声認識システムの性能検定と同様の方法を用いて，2 項分布の差の検定を用いて検定を行った．単語音声認識の 1-best と正解音素推定器の 1-best を検定したところ，有意水準 5 %で認識結果に有意差がある結果が得られた．

次に，SDPWS 講演音声での認識性能の調査結果を表 4.2 に示す．

SDPWS 講演音声においても，深層学習を用いた正解音素推定器が最も高い性能となった．このことから，コーパスがオープンな環境においても正解音素推定器を用いることにより，高精度な音素系列を生成することが分かった．

4.4 まとめ

本章では，DNN を用いた正解音素推定器について述べた．

具体的には，正解音素推定器がどのようなものか説明した．次に，複数の音声認識システムの結果に対してアライメントを行い，正解音素推定器の学習に用いることについて説明した．そして単純な構造の正解音素推定器がどのような構造なのか説明した．最後に，評価実験を行い，音声認識結果の音素列を正解音素推定器により性能の改善を行うことが

できることを示した.

第5章　時系列情報を考慮した正解音素推定器

本章では，時系列を考慮した正解音素推定器について述べる．

第3章では，深層学習を用いた正解音素推定器について述べた．

本章では，前節で説明した正解音素推定器をさらに高精度にするために，時系列を考慮した正解音素推定器について述べる．

まず最初に，正解音素列推定における時系列情報について述べる．次に，時系列情報を考慮できる正解音素推定器の構造について述べる．そして，時系列を考慮した正解音素推定器の性能調査の実験について述べる．さらに，実験結果として時系列を考慮することで正解音素推定の性能が改善することについて述べる．

5.1　正解音素推定における時系列情報

音声は，時系列を持ったデータであり，音と音が時間方向で組み合わさることで意味が付いてくる．また，時間方向において制約も存在する．例えば，日本語において子音の後に母音が現れることは絶対であり，子音の後に子音が現れることはない．しかし，時間情報を用いなければこのような制約を考慮することができない．そのため，単純な正解音素推定器では，このような間違いが出現する可能性がある．このことから，正解音素推定器に対して時間情報を付与することは高精度な推定には必要である．

5.2　時系列を考慮した正解音素推定器

時系列を考慮した正解音素推定器が，実際に正しい音素列を推定することができるのか調査した．

調査する正解音素推定器は，時系列を考慮した構造で性能の調査する．今回，時系列を考慮した構造として，LSTM か GRU を有した中間層を持った構造である．例えば，GRU を用いた場合の時系列を考慮した構造を図 5.1 に示す．

具体的には，音素列をベクトル系列に変換する方法としては，4.2 節に示した音素 Embedding を用いて変換した．そして，音素 Embedding の結果を，LSTM や GRU で処理を行う．ここで，1 つの LSTM や GRU は単方向の時系列情報しか扱うことができない．しかし，正解音素推定は認識結果の文字列に対して行うため，未来の時間情報も扱うことができる．そこで，2 つの LSTM や GRU を用いて過去の情報と未来の情報の両方向の時系

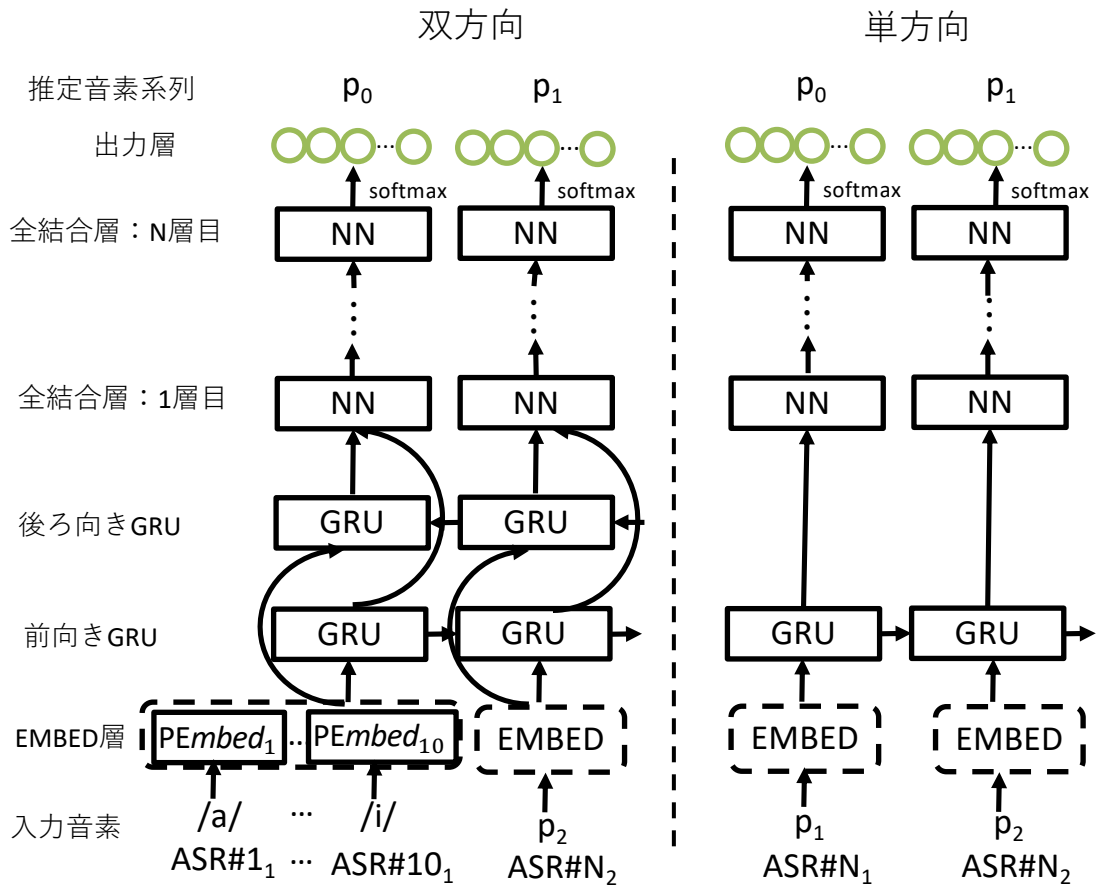

図 5.1: 時系列を考慮した正解音素推定機の構造

列を利用する．そして，その結果を多層の全結合層に通して，出力層に伝播させていく構造となっている．

5.3 Attention 機構を導入した正解音素推定器

Attention 機構 [60, 61] とは，複数存在する情報（例えば，時系列データ）に対してどの情報に対して注目するかを選択する機構である．

本研究では時系列の選択に Attention 機構を導入する．時系列情報を過去と未来の情報を利用することで多様な時系列パターンに対応することができる．しかし，情報量が増加することは学習が困難になりやすくなり性能が必ずしも改善するとは限らない．そこで，過去と未来の時系列のどちらの時系列が有効なのかを選択する Attention 機構を用いる．

本研究で用いる Attention 機構を導入した正解音素推定器の構造を図 5.2 に示す．

基本構造は GRU を有した両方向の時系列を考慮した正解音素推定器と同様である．

ここで，“AU” は Attention 用の全結合層のことを表している．“AU” を用いることで時系列情報として有用なのかを判断することが可能となる．“AU” の式を以下に示す．

$$\mathrm{AU}(x) = \tanh(FC(x;\theta)) \tag{5.1}$$

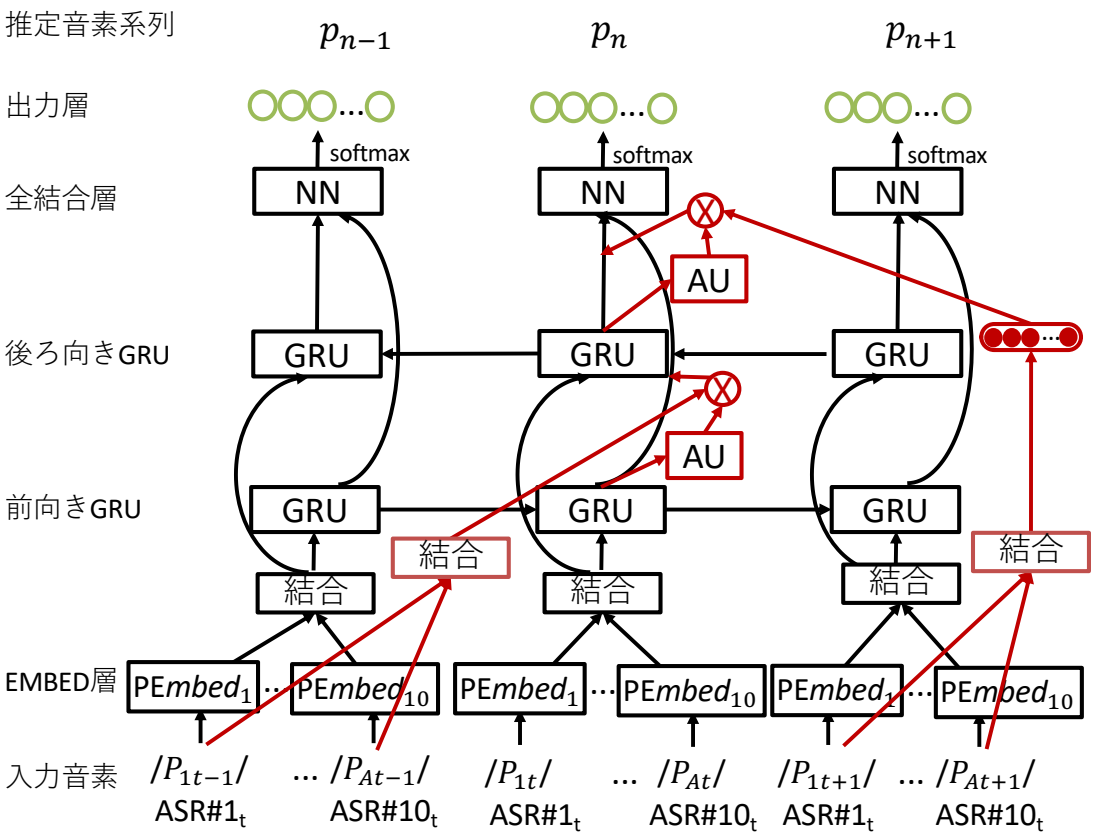

図 5.2: Attention 機構を導入した正解音素推定器の構造

“AU” の出力と前後の音素 Embedding 結果を掛け合わせることで重要度の重みを計算する．この得られた重要度を両方の時系列にそれぞれ掛け合わせることで，最終的に選択された時系列情報だけを獲得することができる．

5.4 評価実験

5.4.1 正解音素推定器のハイパーパラメータ

時系列を考慮した正解音素推定機は，時系列を考慮するために LSTM と GRU を利用する．ここで，単方向と両方向の違いを検証するためにそれぞれの構造で実験を行う．その他のハイパーパラメータは単純な正解音素推定器と同様で，正解音素推定機の中間層の全結合層の数は 7 とし，各層の活性化関数には ReLU を用いた．音素 Embedding は各 5 次元で合計 50 次元，中間層の各層は 512 次元で実験を行った．各層のパラメータの初期化方法には −0.1 から 0.1 の一様分布の乱数で初期化を行った．最適化手法には，確率的勾配降下法（Stochastic Gradient Descent：SGD）を用いた．また，Dropout を 20%で各層に行っている．

5.4.2 ベースライン

ベースラインとして，前章と同様に，複数の音声認識システムのなかで最も性能が高い単語音声認識システムの音素列，ROVER 法を用いて評価を行う．また，時系列を考慮することで性能が上がるか調査するために，単純な正解音素推定器とも性能の比較を行う．

5.4.3 データセット

正解音素推定器の学習データは，前章で述べたデータセットと同様で CSJ のコア講演を除く全講演の音声認識結果から学習している．音声認識システムには，2 章で説明した 10 種類の音声認識システムを用いて音声認識を行っている．

評価データに，前章と同様な CSJ（“CSJ set”）と音声ドキュメント処理ワークショップ（“SDPWS set”）の 2 つの音声コーパスを用意した．

5.4.4 正解音素推定の評価尺度

音素推定の評価には，音素正解率を用いる．音素正解率は，正解音素列をどれだけ検出することができたかを示している．つまり，音声認識の評価で利用される正解率（correct rate）とほぼ同等の評価指標となる．そのため，この尺度では発話単位での正解音素の検出性能の評価となる．音素正解率は，最大で 5-best までの出力を用いて評価する．DNN の正解音素推定器での 5-best の場合には，DNN の出力から得られる音素 posteriorgram の出現確率が上位 5 位までの中に，正解音素が存在するかどうかを調査する．

5.4.5 実験結果

まず，コア講演音声での認識性能の調査結果を表 5.1 に示す．参考として複数の音声認識の音素列の選択を理想的にできた場合の音素正解率は 97.3% となっている．

単純な深層学習と比較して，時系列を考慮した正解音素推定器の方が高い性能となった．また，LSTM と GRU の結果を比較すると性能に大きな違いが存在しないことが分かる．さらに，単方向と双方向では，双方向を用いた方の性能が高くなっていることが分かる．このことから，正解音素推定器に時系列情報を用いることが有効であることが分かる．

また，Attention 機構を導入した正解音素推定器は，“B-GRU” と比較して性能の違いがあまりないことが分かる．これは，時系列を考慮する “LSTM” や “GRU” の構造が時系列の選択も内部で行われている可能性が存在することが考えられる．そのため，Attention 機構などの外部に選択する機構はあまり必要がないと考えられる．

また，SDPWS 講演音声での認識性能の調査結果を表 5.2 に示す．

SDPWS 講演音声においても，1-best の結果などの全体の傾向としては CSJ コア講演と同様で，時系列情報は用いることで性能が改善し，双方向の時系列を用いた構造の性能が

表 5.1: コア講演音声の音素推定精度 [%]

N-best	1-best	2-best	3-best	4-best	5-best
単語音声認識	91.4	92.4	92.9	93.2	93.3
ROVER	90.3	96.0	96.8	97.0	97.0
DNN	91.5	96.1	97.0	97.6	98.0
U-LSTM	91.5	96.2	97.2	97.7	98.1
U-GRU	91.9	96.2	97.1	97.7	98.0
B-LSTM	92.1	96.5	97.6	98.1	98.4
B-GRU	92.2	96.4	97.5	98.0	98.4
Attention	92.1	96.4	97.5	98.0	98.4

表 5.2: SDPWS 音声の音素推定精度 [%]

N-best	1-best	2-best	3-best	4-best	5-best
単語音声認識	83.5	84.6	85.0	85.3	85.5
ROVER	82.1	90.9	92.1	92.4	92.4
DNN	84.8	91.1	94.0	95.5	96.1
U-LSTM	85.1	92.0	94.3	95.5	96.2
U-BGRU	85.7	91.2	93.5	95.4	96.1
B-LSTM	85.9	91.5	94.0	95.6	96.5
B-BGRU	85.9	91.8	94.3	95.7	96.4
Attention	85.9	91.8	94.4	95.6	96.3

高いことが分かる．このことから，コーパスがオープンな環境においても時系列情報を考慮することにより，正解音素推定器の性能を改善させることが分かる．

5.5 時系列を考慮した深層学習の構造

本節で説明した構造以外にも，時系列を考慮した構造はほかにも考えられる．そこで，予備実験として付録 A に構造を比較した実験結果をまとめる．

5.6 まとめ

本章では，時系列情報を利用した正解音素推定器について述べた．

具体的には，正解音素推定器に対して時系列を考慮する必要性について述べた．次に，正解音素推定器において時系列を考慮するための構造について述べた．そして，評価実験から時系列を考慮することにより正解音素推定の精度を改善できることを述べた．

第6章　正解音素推結果からの音声中の検索語検出

本章では，音声中の検索語検出について述べる．

第 4 章では，時系列を考慮した正解音素推定器について述べた．

本章では，正解音素推定器の結果を用いた応用技術として，音声中の検索語検出について述べる．

まず，音声中の検索語検出について述べる．次に，正解音素推定器を用いてどのように音声中の検索語検索を行うか述べる．さらに，比較手法として条件付き確率場を用いた三つ組音素検出器と検索エンジンについて述べる．そして評価実験で，比較手法よりも正解音素推定器の検索エンジンのほうが高い性能が得られたことを述べる．

6.1　音声中の検索語検出とは

音声ドキュメント検索の一分野である STD の目的は，検索語 (1 個以上の単語からなる言葉) が話されている箇所を音声ドキュメント中から特定することである（図 6.1）．一般的に，STD は音声認識システムの認識結果を利用し，検索語の検索を行う．しかし，音声認識システムは認識辞書に登録されていない語 (未知語) を正しく認識することができない．そのため，正しく音声認識できていない音声に対しても，正確に検索語を検出しなければならない．

本研究で提案した正解音素推定器を用いることにより，認識誤りが少なくなり，正解音素が増えることを第 3 章と第 5 章で示した．そこで，正解音素推定器を用いた音素列が STD に対して有用なのか調査を行った．

6.2　正解音素推定器を用いた検索エンジン

各アライメントに対して正解音素推定器が出力した音素事後確率系列（phoneme posteriorgram）に対して DP マッチングでクエリの検出を行う．この流れを図 6.2 に示す．

まず，検索対象の音声を複数の音声認識システムで音声認識を行い文字列に変換を行う．この複数の認識結果に対して，正解音素推定器を用いて音素事後確率系列に変換する．音素事後確率系列は，各アライメント区間における各音素の確率を示している．そこで，クエリの音素列で音素事後確率系列に対して各音素の確率を用いて探索を行うことで，クエリがどこの区間に存在するか探索することができる．具体的には，音素事後確

図 6.1: STD タスクの具体例

率系列に対して以下の式 (6.1) で表される DP マッチングを用いてクエリの音素列を探索する．

$$D(i,j) = max \begin{cases} D(i,j-1) \times 0.0 \\ D(i-1,j) \times prob(Query(j),i) \\ D(i-1,j-1) \times prob(Query(j),i) \end{cases} \tag{6.1}$$

ここで $prob(Query(j),i)$ は，i 番目のノードにおける $Query(j)$ の音素の推定確率を示している．脱落誤りの遷移コストは “0.0” とした．すなわち，脱落誤りは一切許していない．

この式 (6.1) から求めた結果を，クエリのその発話における検出確率とする．

6.3 条件付き確率場を用いた 3 つ組音素検出器と検索エンジン

正解音素推定器を用いることにより高精度な検索が可能であるか比較検証するために，条件付き確率場（Conditional Random Field : CRF）を用いた検索エンジンと比較を行った．

6.3.1 条件付き確率場

CRF は，自然言語処理の分野，例えば固有表現抽出や文節チャンキング等の研究で広く使われており，その有効性が示されている．音声言語処理の分野でも句読点挿入 [62] や

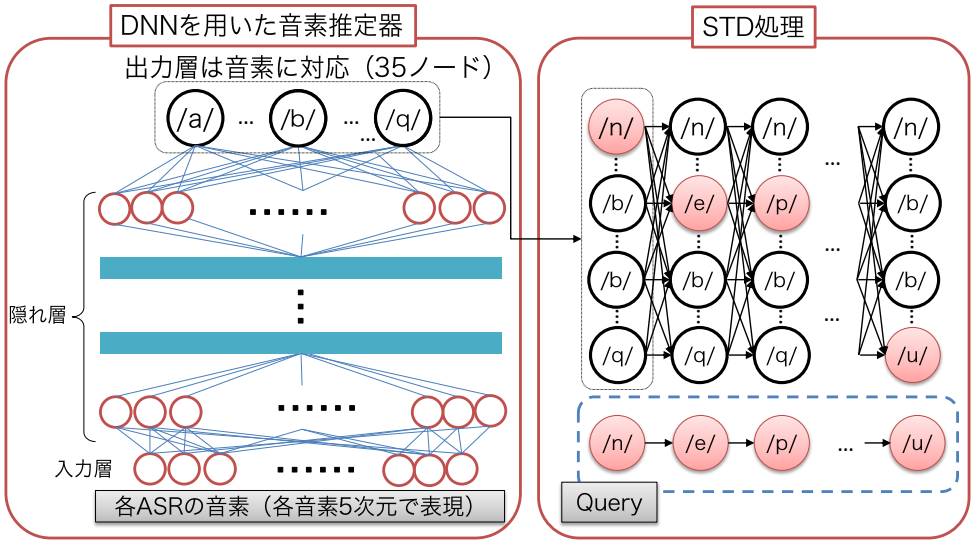

図 6.2: DNN を用いた音素推定器と phoneme posteriorgram に基づく STD 処理

未知語検出 [63]，誤り訂正 [64] 等で利用されている．

CRF では，ある入力記号系列 x が与えられたときの出力ラベル系列 y が得られる条件付き確率は次の式で計算できる．

$$P(y|x) = \frac{1}{Z(x)} exp(\sum_{k} \lambda_k F_k(y, x)) \tag{6.2}$$

ここで，$F_k(y, x)$ は素性関数，λ_k は素性関数に対する重みである．$Z(x)$ は正規化項で，以下の式で表現される．

$$Z(x) = \sum_{y} exp(\sum_{k} \lambda_k F_k(y, x)) \tag{6.3}$$

6.3.2 CRF を利用した音声中の検索語検出

CRF を利用した STD の流れを図 6.3 に示す．

CRF で単語の検出を行うことは，全ての単語を学習しなければならず，困難である．そこで，単語は音素表記に変換することができることから，CRF で triphone 単位の検出を行う．

学習には，10 種類の音声認識システムから得られる音素系列の書き起こしを利用して，音素の 3 つ組である triphone を検出する CRF モデルを学習する．検索語は，一度音素列に変換され，さらに triphone に分解し，各 triphone に対応する CRF モデルを用いて，ある発話にその triphone が含まれている確率を計算する．検索語を構成する全ての triphone の条件付き生起確率の総乗を，この検索語の検出スコアとする．

音素間誤りパターンを学習するために，10 種類の音声認識システムから得られる音素系列の書き起こしから得られる素性を利用する．具体的な例を図 6.4 に示す．

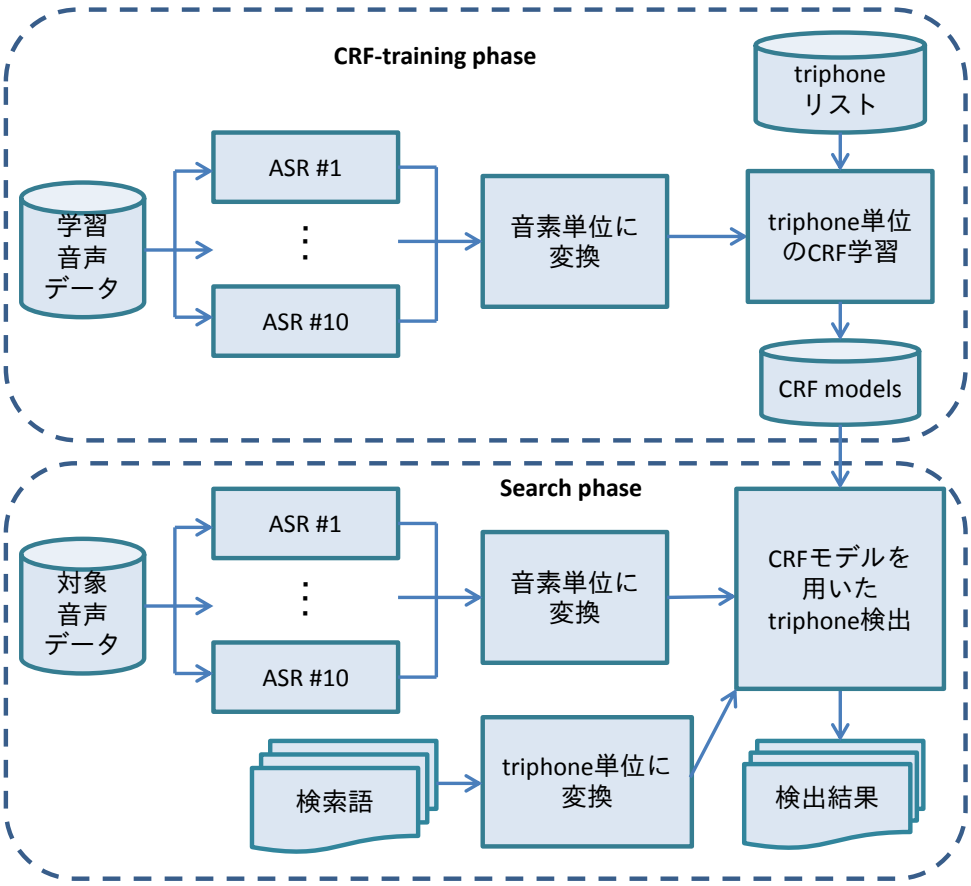

図 6.3: CRF を利用した STD の流れ

図 6.4 の学習データは，10 種類の音声認識システムから得られる音素を ASR#1 を基準に，DP マッチングを用いてアライメントを行っている．アライメントをとった学習データに対して，書き起こしデータから学習対象である triphone の出現位置を特定し，BIO（beginning/inside/outside）ラベルを付与し作成する．BIO ラベルを用いた CRF はテキストを小さいグループに分割する問題を解決するために用いられている．“B” ラベルは，実際に学習対象の triphone の最初の音素が存在する箇所につける．“I” ラベルは，“B” の後に続く triphone の音素が存在する箇所につける．そして，“O” ラベルは対象の triphone が存在しない箇所につけられる．この BIO ラベルを各アライメントに対して付与する．また，評価データも同様の手順を用いて作成する．

CRF による triphone 検出モデルの学習データは，CSJ のコア講演を除く全講演の音声認識結果から学習している．学習には CRF++toolkit [1] を用いた．

学習に用いる素性は，複数の認識システムの出力から得られる unigram，bigram，trigram を用いる．表 6.1 に全ての素性と，各素性の数を示す．h_p^i は i 番目の認識システムの p 番目に存在する音素を示している．r_p は BIO タグのことを示している．図 6.4 にも示すように bigram については，current token を中心として前後 1 つのコンテキストを考慮する．また，trigram も，２つ前までの履歴，前後 1 つずつ，後ろ 2 つまでのコンテキストを考慮している．さらに，認識システム間の bigram も素性に利用する．これによって，音素

[1] CRF++: Yet Another CRF toolkit,https://code.google.com/p/crfpp/

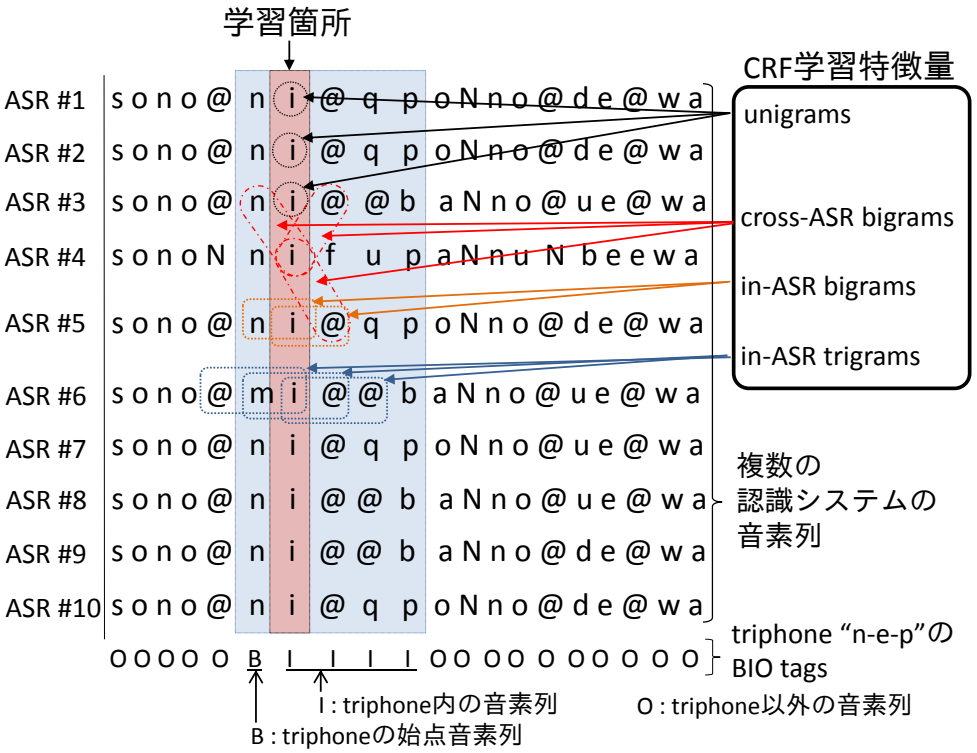

図 6.4: CRF の学習例

表 6.1: CRF の学習素性

学習素性	定義	素性数
unigram	$(r_p,\ h^i_t)$	10
in-ASR bigram	$(r_p,\ h^i_{p-1},\ h^i_p)$, $(r_p,\ h^i_p,\ h^i_{p+1})$	20
cross-ASR bigram	$(r_p,\ h^i_{p-1},\ h^j_p \mid i \neq j)$, $(r_p,\ h^i_p,\ h^j_{p+1} \mid i \neq j)$	180
in-ASR trigram	$(r_p,\ h^i_{p-2},\ h^i_{p-1},\ h^i_p)$, $(r_p,\ h^i_{p-1},\ h^i_p,\ h^i_{p+1})$, $(r_p,\ h^i_p,\ h^i_{p+1},\ h^i_{p+2})$	30

間の多様な音声認識誤りのパターンを CRF で学習させ，認識誤りを含んだ音素系列の書き起こしに対して頑強な triphone 検出を試みる．

ある発話 i に含まれる N 個の triphone から構成される検索語 T の検出確率 $P(T|x_i)$ は，以下の式で計算できる．

$$P(T|x_i) = \left\{ \prod_{j=1}^{N} P_{t_j}(y|x_i) \right\}^{\frac{1}{N}},\ (l_{t_1} < l_{t_j} < l_{t_N}) \tag{6.4}$$

ここで，t_j は検索語 T の j 番目の triphone，x_i は発話 i における入力記号系列，y は出力ラベル系列 (ただし発話全体ではなく一部分の区間のみ) である．l_{t_j} は N 個の triphone から構成されており，各 triphone の出現位置に矛盾が生じていないことが前提となっている．各 triphone の検出確率 $P_{t_j}(y|x_i)$ は，発話全体の出力ラベル列を用いて計算するの

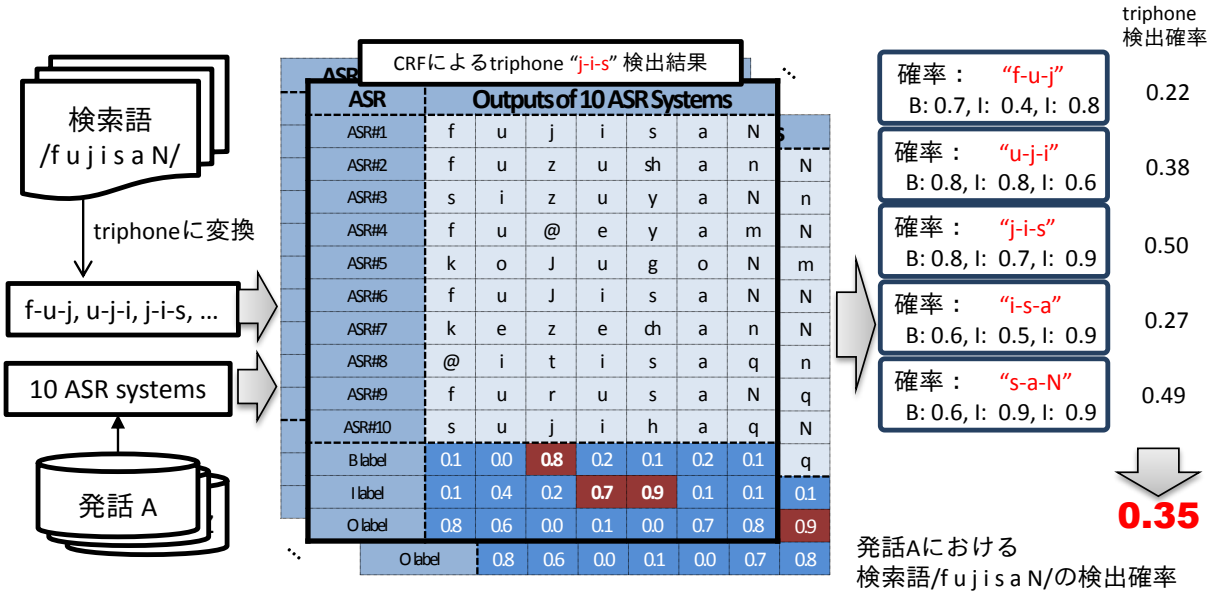

図 6.5: CRF 手法による検出例

ではなく，B と I ラベルが出力されたときのラベル系列のみしか考慮しない．すなわち，O ラベルは無視する．これは，最大エントロピー法とほとんど同じであるが，CRF では，入力系列全体に対して最適な出力ラベル系列を付与することができる．そのため，BI ラベル系列を検出する本タスクでは，CRF モデルのほうが高い精度で triphone を検出することができる．

最終的に，triphone t_j の検出確率は以下の式で計算する．

$$P_{t_j}(y|x_i) = \prod_{L=B}^{I_{tail}} P_{t_j}(L|x_i) \tag{6.5}$$

ここで，B と I_{tail} は，それぞれ triphone t_j の先頭と最後尾の検出確率である．すなわち，triphone t_j の生起確率は，先頭ラベルから最後尾ラベルまで，それぞれのラベルの条件付き生起確率の総乗を取ることで計算する．もし，$P_{t_j}(y|x_i)$ がある設定確率 ϕ を下回った場合，$P_{t_j}(y|x_i)$ は ϕ に置き換える．これは，検索語 T において，1 つの triphone でも検出されなかった場合に T の検出確率が極端に低くなることを防ぐためである．経験的に，ϕ を 0.01 とした．$P(T|x_i)$ が設定閾値 θ_C を変化させることで，Recall-Precision カーブを描くことができる．

図 6.5 に，CRF を用いて検索語（富士山：/f u j i s a N/）を検出する具体例を示す．triphone “j-i-s” は音声発話 A において，“B” と “I” のそれぞれのラベルを乗算した値となり，0.5（$0.8 \times 0.7 \times 0.9$）という検出確率となっている．そして，各 triphone の検出確率の総乗を計算することで，発話 A における富士山の検出確率は 0.35 となる．

6.4 評価実験

6.4.1 STD タスク

STD タスクは音声中から検索語が正しい箇所を検索するタスクである．そして，正しくない箇所で検出された場合，その箇所を誤検出されたと扱う．そのため，誤検出が少なく，正解箇所の検出を行うことが大事である．

検出単位は，発話単位とし，ある発話に単語が含まれている場合に正解発話として扱う．

検索を行うことで，各発話に対してそれぞれスコアが付与されることになる．このスコアを閾値で分けて評価していく．

6.4.2 実験条件

STD の評価は，CSJ のコア講演音声（39 時間の音声）と，SDPWS の講演音声（28.6 時間）を対象とする STD タスクで評価する．情報処理学会のワーキンググループによって制定された STD のためのテストコレクション [38] の 1 つであるコア講演未知語セットと，NTCIR-11 SpokenDoc-2 タスクの moderate-size サブタスクを用いた．コア講演用未知語セットは 50 種の検索語，総出現箇所 233 のテストセットとなっている．moderate-size サブタスクは，100 種類（未知語 53，既知語 47）の検索語で，総出現箇所 962 のテストセットである．

評価実験は，3 章と 5 章でそれぞれ示した正解音素推定器を用いた STD と，CRF を用いた STD の比較を行う．

6.4.3 評価尺度

検索性能の評価には，F 値，MAP(Mean Average Precision) を用いた．以下に，評価式を示す．

$$Recall(t) = \frac{N_{corr}(t)}{N_{true}} \tag{6.6}$$

$$Precision(t) = \frac{N_{corr}(t)}{N_{corr}(t) + N_{spurious}(t)} \tag{6.7}$$

$$F\text{ 値 }(t) = \frac{2 \times Recall(t) \times Precision(t)}{Recall(t) + Precision(t)} \tag{6.8}$$

$$MAP = \frac{1}{Q} \sum_{q=1}^{Q} AveragePrecision(q) \tag{6.9}$$

$$AveragePrecision(q) = \frac{1}{N_{true}(q)} \sum_{k=1}^{R} \delta_k \times Precision_{rank}(k) \tag{6.10}$$

表 6.2: コア講演未知語セットにおける F 値と MAP

システム名	最大 F 値 [%]	MAP
CRF	61.1	0.795
DNN	68.4	0.812
LSTM	71.7	0.840
GRU	72.0	0.826
BLSTM	73.6	**0.853**
BGRU	**75.2**	0.842
Attention	74.9	0.849

$$Precision_{rank}(k) = \frac{\text{第 } k \text{ 位までに得られた正解数}}{k} \tag{6.11}$$

q は検索語を表しており，検索語ごとに算出されることを示している．また，Q はテストセットの検索語数を表す．

N_{corr} は検出された適合検索語の出現数を表し，$N_{spurious}$ は誤検出された検索語の出現数を表す．N_{true} は音声データ中に本来存在する検索語の出現総数を表す．

F 値は全検索語の合計検索結果から算出したものを用いている．

式 (6.10) の R は最後に正解が表れた順位を表し，δ_k は k 位の区間が正解であれば 1，不正解であれば 0 となる．式 (6.11) は第 k 位の候補における適合率を示す．MAP は Average Precision(AP) を全検索語で平均したものであり，AP は正解出現時の適合率を平均したものである．

6.4.4 実験結果

コア講演未知語セットの STD 実験の結果を表 6.2 に示す．

CRF を用いた検索結果と比較して，正解音素推定器を用いた検索結果のほうが最大の F 値と MAP ともに高い性能となった．このことから，CRF よりも高精度な検索インデックスを作成することができていることが分かる．次に正解音素推定器同士を比較すると，全体的に音素正解率が高い結果が基本的に検索性能が高いということが分かる．

また，moderate-size サブタスクの実験結果を表 6.3 に示す．コア講演未知語セットと同様に，CRF の検索結果よりも，正解音素推定器の検索結果のほうが高い検索性能となった．また，単純な深層学習構造よりも，時系列を考慮した正解音素推定器のほうが高い検索性能となっている．

このことから，正解音素推定器を用いて高精度な検索インデックスを作成することで，STD の検索性能に応用することができ，正解音素推定器を用いて高精度な検索インデックスを作成することが検索性能の改善につながることが分かった．

表 6.3: moderate-size task における F 値と MAP

システム名	最大 F 値 [%]	MAP
CRF	28.6	0.460
DNN	44.0	0.557
LSTM	46.1	0.543
GRU	45.8	0.552
BLSTM	46.5	0.564
BGRU	**47.0**	**0.565**
Attention	46.2	0.556

6.5 まとめ

本章では，正解音素推定器の応用技術として，正解音素推定結果からの STD について述べた．

具体的には，音声中の検索語検出とはどのような技術なのか述べた．次に正解音素推定器の結果から STD を行う検索エンジンについて述べた．そして，比較手法として条件付き確率場を用いた 3 つ組音素検出器と検索エンジンについて述べた．最後に評価実験では，高精度な正解音素推定器を用いて検索インデックスを作成することで検索性能を改善することができることを述べた．

第7章　正解音素推結果からの単語変換器

本章では，正解音素推定器からの単語変換について述べる．

第5章では，正解音素推定器の応用技術として正解音素推定器の結果音声中の検索語検出について述べた．

本章でも，正解音素推定器の結果を用いた応用技術として，正解音素推定器の推定結果からの単語変換器について述べる．

7.1　正解音素推定からの誤り単語修正

正解音素推定器は，音声認識誤りや未知語の音声認識に対して頑健な処理を行うために精度の高い音素列を推定する．しかし，実際に音声認識アプリケーションで音声認識結果を利用する場合，音声認識結果として単語列を入力することが一般的である．そのため，正解音素推定した結果を単語列に変換して認識結果の単語列の精度を改善することで，どのようなアプリケーションにも適用可能にすることができると考えた．

そこで，本章では，認識結果の単語列の誤っている単語を，正解音素推定器で変換した単語で置換することで誤りが少ない単語列を生成する方法について述べる．

7.2　重みつき有限状態トランスデューサ

7.2.1　重みつき有限状態トランスデューサとは

重みつき有限状態トランスデューサ（Weighted Finite State Transducer：WFST）は，状態遷移機械のモデルとして知られる有限オートマトンの一種である．有限オートマトンの最も基本的なモデルは，有限状態アクセプタ（Finite-State Acceptor：FSA）と呼ばれ，ある特定の記号列を受理するか否かを表す．FSA は状態と状態遷移の有限の集合によって記述され，各状態遷移に受理できる記号を持つ．入力記号列が与えられたとき，それを受理するような初期状態から終了状態に至る状態遷移過程が存在するときその記号列は受理される．

FSA の拡張として，有限状態トランスデューサ（Finite-State Transducer：FST）がある．FST は記号列を受理し，それと同時に別の記号列を出力する記号列変換モデルである．さらに，状態遷移に対して重みを付与することでコストや確率といった概念の導入を可能にしたモデルが WFST である．

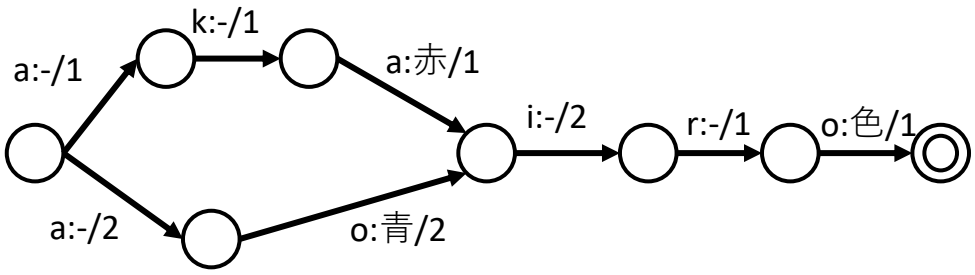

図 7.1: WFST の言語モデル

7.2.2 重みつき有限状態トランスデューサによるモデル表現

音声認識システムの各モデルを WFST[65, 66] で表すことができる．たとえば，言語モデルは図 7.1 のように表すことができる．入力記号は音素の文字列を表す記号（たとえば音素 “a”）．出力記号はその音素の文字列が表す単語単位（たとえば単語 “赤”）である．WFST の重みは言語繋がりの状態遷移における重みとなる．この WFST を式で表すと以下の式で表すことができる．

$$\hat{W} \sim \mathrm{argmax} \prod_{t=1}^{T} P(o_t, s_t | s_{t-1}) \tag{7.1}$$

ここで，T は入力される系列の長さを表しており，s_t は t 番目に入力される音素系列，o_t は t 番目に出力される単語系列を表している．

7.3 正解音素推定結果からの単語変換処理

正解音素推定結果を単語に変換する処理の概要を図 7.2 に示す．ここで，“FD” が誤り箇所検出，$\hat{p}_i$ が i 番目の区間の正解音素推定結果を表す．また，$\hat{w}_t$ は t 番目の単語の推定結果を表している．

正解音素推定結果を単語に変換する処理の流れは，まず，単語認識結果で誤っている区間を特定する，誤り箇所検出を行う．そして，誤っている区間は正解音素推定結果を用いて，正しい区間は単語認識結果の音素列を使用して単語変換処理を行う．この単語変換処理には WFST を使用する．誤っている区間の単語列を WFST から得られた単語列に入れ替える．こうすることにより，誤っている単語を正しい単語に置き換えることができ，正しい単語列を獲得することができる．

7.3.1 単語認識結果の誤り箇所検出

単語認識結果が誤っているかの判定には，単語認識結果の音素列と 10 種類の音声認識結果で比較して異なる箇所を誤っていると仮定した．具体的には，10 種類の音声認識結果に対して，多数決で一つの音素列に絞り込む．この音素列と単語認識結果の音素列で，DP マッチングを行い，誤りを含む単語区間を誤り箇所とした．

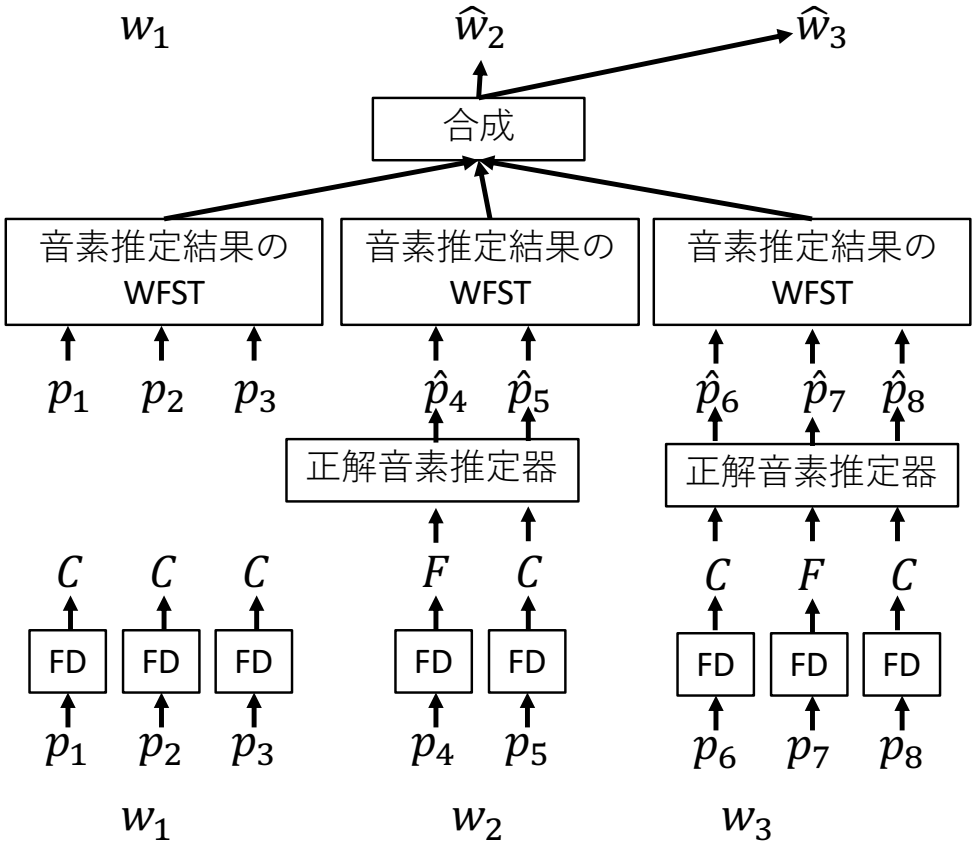

図 7.2: 正解音素推定結果からの単語変換処理の概要

本研究では，誤り箇所検出は簡単な実装で実現している．この方法では，誤り箇所となりえる箇所はほとんど全て検出することができるが正しい箇所も検出されると考えられる．このため，高精度な誤り箇所検出を行うよりも，さらに正解音素推定器からの単語列の精度を調査できると考えられる．

7.4 単語変換処理の流れ

正解音素推定器の結果を用いて音声認識単語列に変換する流れを説明する．

7.4.1 入力音素列の決定

変換処理を行う音素列の区間を決定する．

WFST に入力するための音素列は，前後の単語との繋がりを考慮することができるように 3 つ組の単語区間の音素列を入力する．誤り区間検出においての正解区間では入力する音素列は単語音声認識の結果を用いる．また，誤り箇所検出された区間の音素列には正解音素推定結果で置き換えることで，誤っている音素列に対しても正解音素推定結果を活用することで正しい音素列が獲得できる．

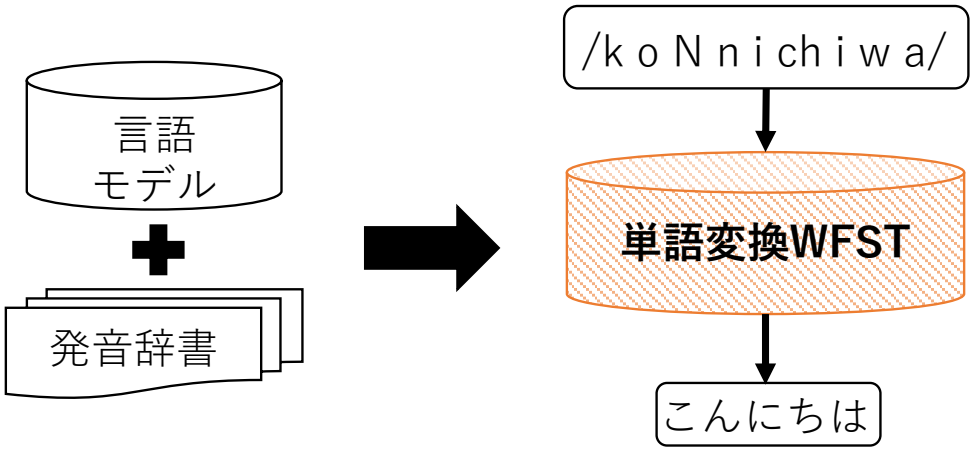

図 7.3: 単語変換用の WFST 作成方法

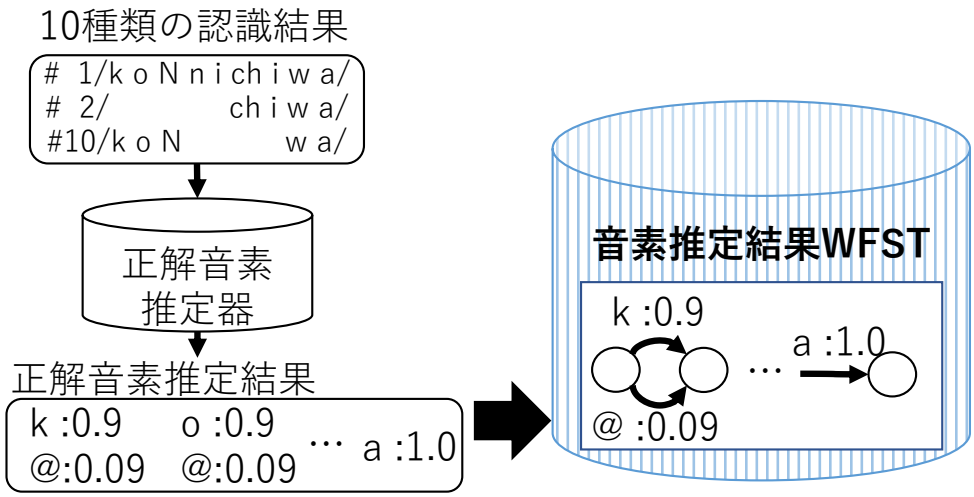

図 7.4: 音素推定結果の WFST 作成の概要

7.4.2 単語変換用の WFST 作成方法

単語変換用の WFST を作成の概要を図 7.3 に示す．音声認識ツールキットである Kaldi の CSJ レシピ [67] を参考に，言語モデルを用いて音素から単語に言語つながりを考慮して変換する WFST を作成する．

7.4.3 音素推定結果の WFST 作成

音素推定結果の WFST 作成の概要を図 7.4 に示す．3 つ組の単語の音素推定結果ごとに WFST モデルに変換する．各音素の出現確率を WFST の遷移の重みとしてモデルを作成する．ここで，単語認識結果が正しい場合は重みを 1 として単語認識結果の音素を持った単一の遷移とする．

このとき，出現確率が低い音素は信用できない音素であるため使用しない．この閾値として出現確率が 0.1% 未満の音素を信用できない音素とした．

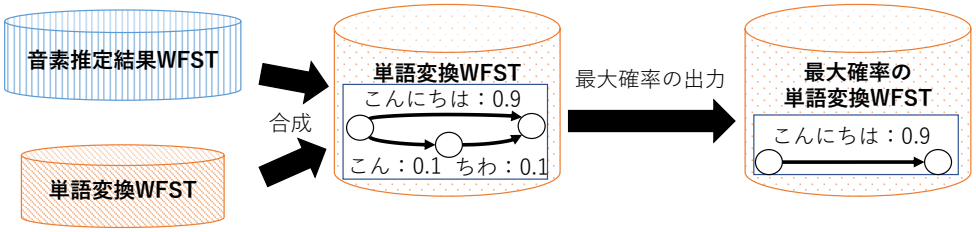

図 7.5: 2 つの WFST を合成の概要

7.4.4 2 つの WFST を合成

2 つの WFST を合成の概要を図 7.5 に示す．2 つの WFST を合成して，音素推定結果から構成される音素列が通ることができる WFST を作成する．

結合したのち，確率が一番高いパスだけを出力する．これにより，音素推定結果から一番高い単語系列を出力することができる．

7.5 評価実験

7.5.1 評価データ

音声データは，コア講演音声の中の一部を用いて評価を行った．コア講演から 17 講演を選択し，その講演に含まれる 4,696 発話を評価対象とした．講演 ID のリストを表 7.1 に示す．

7.5.2 評価尺度

評価尺度には，単語音声認識結果を用いて単語単位の音声認識率で評価を行った．また，正解音素推定で誤り単語を減らすことが目的であるため，認識結果が誤っている箇所のみを対象とした音声認識率でも評価を行う．

7.5.3 実験結果

単語音声認識率を表 7.2 に示す．正解音素推定器の単語変換器を用いて誤り単語を入れ替えることにより，単語音声認識率が改善していることが分かる．また，単語音声認識で認識誤りしている単語のみに着目した場合には，16.8% の単語を修正することができていた．このことから，正解音素推定を行うことで正しい単語に変換することができ，音声認識の精度を改善することができることが分かった．

表 7.1: 実験に用いる講演 ID のリスト

講演 ID
A01F0055
A01F0067
A01F0122
A01F0132
A01F0143
A01F0145
A01M0007
A01M0015
A01M0020
A01M0021
A01M0025
A01M0030
A01M0048
A01M0056
A01M0065
A01M0070
A01M0074

表 7.2: 正解音素推定結果からの単語音声認識率 [%]

	単語正解率
単語音声認識結果	59.4
正解音素推定結果からの単語変換	64.1

7.6 まとめ

本章では，正解音素推定結果からの単語変換器について述べた．

具体的には，正解音素推定から単語に変換することで音声認識システムを活用した応用システムに適用可能であることについて述べた．次に，音素列を単語系列に変換することが可能である WFST について述べた．そして，WFST を用いた正解音素推定器の結果を単語に変換する方法について述べた．最後に評価実験では，単語音声認識結果の誤り箇所を，正解音素推定結果を用いた単語に入れ替えることで認識率が改善したことについて述べた．

第8章　結言

本論文では，深層学習を用いた正解音素推定器と推定結果からの応用技術について述べた．

第2章では，本研究で用いる複数の音声認識システムを述べた．まず，音声認識システムについて述べ，音響モデルと言語モデルについて述べた．実際に本研究では，複数の音声認識システムとして5種類の言語モデルと2種類の音響モデルを用意した．この複数の音声認識システムのそれぞれの認識性能について述べた．

第3章では，深層学習について述べた．深層学習とはどのような技術なのか述べた．具体的に，基本的な深層学習がどのような構成なのか述べ，深層学習に用いられる一般的な要素について述べた．さらに，時系列を考慮するためにどのような要素が用いられているのかについて述べた．また，深層学習の汎化性能を改善させる技術についても述べた．

第4章では，正解音素推定器の概要について説明した．次に，正解音素推定器の学習に用いるために，複数の音声認識システムの結果を利用することについて述べた．そして単純な構造の正解音素推定器がどのような構造なのか説明した．最後に，評価実験を行い，音声認識結果の音素列を正解音素推定器により性能の改善を行うことができることを示した．

第5章では，時系列を考慮した正解音素推定器について述べた．具体的には，正解音素推定器に対して時系列を考慮する必要性について述べた．次に，時系列を考慮できる正解音素推定器の構造について述べた．そして，評価実験から時系列を考慮することにより正解音素推定の精度を改善できることを述べた．

第6章では，正解音素推定器の応用技術として，音声中の検索語検出について述べた．音声中の検索語検出について述べ，正解音素推定結果からどのように検索を行うか述べた．そして，比較対象としてCRFを用いたtriphone音素検出器とその結果からの検索方法について述べた．結果として，深層学習を用いた正解音素推定器が，CRFを利用した検索よりも高精度な検索が実現でき，検索インデックスを高精度にすることで検索精度を改善することを述べた．

第7章では，正解音素推定器の応用技術として，単語変換器について述べた．まず，正解音素推定器の推定結果をWFSTを利用して単語列に変換する方法について述べた．そして，単語音声認識結果の誤り単語に対して正解音素推定結果からの単語変換器で単語を置換することで，認識結果の単語列の精度を改善したことを述べた．

今後の課題として，以下のことが挙げられる．

本研究の正解音素推定器では，事前処理として時間情報に基づきアライメントを行っていた．しかし，アライメント処理を行うとアライメント精度に依存し性能が低下してしま

うことが考えられる．そこで，アライメント処理を行わず，認識結果の文字列を処理をそのまま入力し，複数の認識システムの結合を深層学習により実現することで性能が改善する可能性が存在する．

謝辞

本論文は，筆者が山梨大学大学院医工農学総合教育部情報機能システム工学専攻博士後期課程に在籍中の研究成果をまとめたものである．山梨大学大学院総合研究部工学域准教授西崎博光先生には，研究に関して手法や考察などを様々なご助言や，メールの書き方など一般的な知識に関することまで様々なご指導を頂き，深謝の意を表する．

山梨大学大学院総合研究部工学域教授鈴木良弥先生には，指導教員として様々な場面で学生生活における面倒を見ていただき，また本論文の主査としてご指導いただき深謝の意を表する．

山梨大学大学院総合研究部工学域教授宗久知男先生，山梨大学大学院総合研究部工学域教授大渕竜太郎先生，山梨大学大学院総合研究部工学域教授福本文代先生，山梨大学大学院総合研究部工学域准教授渡辺喜道先生，山梨大学大学院総合研究部工学域准教授丹沢勉先生には，博士論文の副査としてご指導いただき感謝の意を表する．

また，本論文の研究に対して音声・言語系の研究者として様々な先生方にご助言をいただいた．中部大学大学院工学研究科情報工学専攻教授中川聖一先生，筑波大学システム情報系知能機能工学域教授宇津呂武仁先生，徳島大学大学院ソシオテクノサイエンス研究部教授北岡教英先生，静岡大学学術院工学領域数理システム工学系列准教授甲斐充彦先生，筑波大学産業技術学部産業情報学科准教授小林彰夫先生，豊橋技術科学大学工学部情報・知能工学系准教授秋葉友良先生，静岡大学学術院情報学領域情報科学系列准教授小暮悟先生，中部大学大学院工学研究科情報工学専攻准教授山本一公先生，徳島大学大学院社会産業理工学研究部講師西村良太先生には，研究内容や研究方針に関してご助言いただき深謝の意を表する．

豊橋技術科学大学大学院工学研究科情報・知能工学専攻関博史氏には，他大学の同期として切磋琢磨し，相談など乗っていただき感謝の意を表する．

また，研究室生活において，研究室の先輩・同期・後輩方には，研究室における生活において，共に苦労や楽しみを共に充実した研究生活を過ごせたこと，そして研究に関して討論をしていただき感謝の意を表する．

学部時代から含めて西崎研究室で 6 年間過ごせたことを改めて感謝の意を表する．

参考文献

[1] D. Can and M. Saraclar, "Lattice indexing for spoken term detection," IEEE Trans. on Audio, Speech and Language Processing, Vol.19, No.8, pp.2338–2347, 2011.

[2] D. Kaneko, R. Konno, K. Kojima, K. Tanaka, S. wook Lee, and Y. Itoh, "Constructing acoustic distances between subwords and states obtained from a deep neural network for spoken term detection," Proc. of INTERSPEECH 2017, pp.2879–2883, 2017.

[3] S. Natori, Y. Furuya, H. Nishizaki, and Y. Sekiguchi, "Spoken Term Detection Using Phoneme Transition Network from Multiple Speech Recognizers' Outputs," Journal of Information Processing, Vol.21, No.2, pp.176–185, 2013.

[4] J. Kang, W. Zhang, and J. Liu, "Gated convolutional networks based hybrid acoustic models for low resource speech recognition," Proc. of 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pp.157–164, 2017.

[5] G. Kurata, B. Ramabhadran, G. Saon, and A. Sethy, "Language modeling with highway LSTM," Proc. of 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pp.244–251, 2017.

[6] S. Karita, S. Watanabe, T. Iwata, A. Ogawa, and M. Delcroix, "Semi-supervised end-to-end speech recognition," Proc. of INTERSPEECH 2018, pp.2–6, 2018.

[7] S. Braun, D. Neil, J. Anumula, E. Ceolini, and S.C. Liu, "Multi-channel attention for end-to-end speech recognition," Proc. of INTERSPEECH 2018, pp.17–21, 2018.

[8] T. Zenkel, R. Sanabria, F. Metze, and A. Waibel, "Subword and crossword units for ctc acoustic models," Proc. of INTERSPEECH 2018, pp.396–400, 2018.

[9] K.C. Sim, A. Narayanan, A. Misra, A. Tripathi, G. Pundak, T. Sainath, P. Haghani, B. Li, and M. Bacchiani, "Domain adaptation using factorized hidden layer for robust automatic speech recognition," Proc. of INTERSPEECH 2018, pp.892–896, 2018.

[10] A. Stolcke, L. Ferrer, S. Kajarekar, E. Shriberg, and A. Venkataraman, "Mllr transforms as features in speaker recognition," Proc. of the 9th European Conference on Speech Communication and Technology, pp.2425–2428, 2005.

[11] D. Povey, M.J.F. Gales, D.Y. Kim, and P.C. Woodland, "MMI-MAP and MPE-MAP for acoustic model adaptation," Proc. of EUROSPEECH 2003, 2003.

[12] G. Hinton, L. Deng, D. Yu, G.E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T.N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," IEEE Signal Processing Magazine, Vol.29, No.6, pp.82–97, Nov 2012.

[13] H. Hadian, D. Povey, H. Sameti, and S. Khudanpur, "Phone duration modeling for lvcsr using neural networks," Proc. of INTERSPEECH 2017, 2017.

[14] E. Arisoy, A. Sethy, B. Ramabhadran, and S. Chen, "Bidirectional recurrent neural network language models for automatic speech recognition," Proc. of 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.5421–5425, 2015.

[15] M. Singh, Y. Oualil, and D. Klakow, "Approximated and domain-adapted lstm language models for first-pass decoding in speech recognition," Proc. of INTERSPEECH 2017, 2017.

[16] Z. Tüske, R. Schlüter, and H. Ney, "Investigation on lstm recurrent n-gram language models for speech recognition," Proc. of INTERSPEECH 2018, pp.3358–3362, 2018.

[17] B. Srivastava, S. Sitaram, R. Kumar Mehta, K. Doss Mohan, P. Matani, S. Satpal, K. Bali, R. Srikanth, and N. Nayak, "Interspeech 2018 low resource automatic speech recognition challenge for indian languages," Proc. of INTERSPEECH 2018, pp.11–14, 2018.

[18] H.B. Sailor, M. Venkata Siva Krishna, D. Chhabra, A.T. Patil, M. Kamble, and H. Patil, "Da-iict/iiitv system for low resource speech recognition challenge 2018," Proc. of INTERSPEECH 2018, pp.3187–3191, 2018.

[19] S. Watanabe, T. Hori, S. Kim, J.R. Hershey, and T. Hayashi, "Hybrid ctc/attention architecture for end-to-end speech recognition," IEEE Journal of Selected Topics in Signal Processing, Vol.11, No.8, pp.1240–1253, 2017.

[20] L. Lu, X. Zhang, K. Cho, and S. Renals, "A study of the recurrent neural network encoder-decoder for large vocabulary speech recognition," Proc. of INTERSPEECH 2015, pp.3249–3253, 2015.

[21] E. Beck, M. Hannemann, P. Dötsch, R. Schlüter, and H. Ney, "Segmental encoder-decoder models for large vocabulary automatic speech recognition," Proc. of INTERSPEECH 2018, pp.766–770, 2018.

[22] S. Karita, A. Ogawa, M. Delcroix, and T. Nakatani, "Sequence training of encoder-decoder model using policy gradient for end- to-end speech recognition," Proc. of 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.5839–5843, 2018.

[23] C.X. Qin, D. Qu, and L.H. Zhang, "Towards end-to-end speech recognition with transfer learning," EURASIP Journal on Audio, Speech, and Music Processing, Vol.2018, No.1, p.18, 2018.

[24] J.G. Fiscus, "A Post-processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction (ROVER)," Proc. of the 1997 IEEE Workshop on Automatic Speech Recognition adn Understanding (ASRU'97), pp.347–354, 1997.

[25] L. Bai, P. Jancovic, M.J. Russell, P. Weber, and S.M. Houghton, "Phone classification using a non-linear manifold with broad phone class dependent dnns," Proc. of INTERSPEECH 2017, pp.319–323, 2017.

[26] T. Utsuro, Y. Kodama, T. Watanabe, H. Nishizaki, and S. Nakagawa, "An empirical study on multiple lvcsr model combination by machine learning," Proc. of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2004), pp.13–16, 2004.

[27] X. Liu, M.J.F.Gales, and P.C.Woodland, "Language model cross adaptation for lvcsr system combination," Proc. of INTERSPEECH 2010, pp.342–345, 2010.

[28] S. Meng, J. Shao, R.P. Yu, J. Liu, and F. Seide, "Addressing the out-of-vocabulary problem for large-scale chinese spoken term detection," Proc. of INTERSPEECH 2008, pp.2146–2149, 2008.

[29] H. Sagawa, T. Mitamura, and E. Nyberg, "Correction grammars for error handling in a speech dialog system," Proc. of HLT-NAACL 2004: Short Papers, HLT-NAACL-Short '04, Stroudsburg, pp.61–64, 2004.

[30] D. Griol and J.M. Molina, "A framework for improving error detection and correction in spoken dialog systems," SOFT COMPUTING Journal, Vol.20, No.11, pp.4229–4241, 2016.

[31] 中川聖一, 小林聡, 峯松信明, 宇津呂武仁, 秋葉友良, 北岡教英, 山本幹雄, 甲斐充彦, 山本一公, 土屋雅稔, 音声言語処理と自然言語処理, コロナ社, 2013.

[32] A. Lee and T. Kawahara, "Recent development of open-source speech recognition engine julius," Proc. of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), pp.131–137, 2009.

[33] X.D. Huang, Y. Ariki, and M.A. Jack, “Hidden Markov models for speech recognition,” Edinburgh information technology series, No.7, 1990.

[34] T. Marwala, U. Mahola, and F.V. Nelwamondo, “Hidden markov models and gaussian mixture models for bearing fault detection using fractals,” Proc. The 2006 IEEE International Joint Conference on Neural Network Proceedings, pp.3237–3242, 2006.

[35] A. Ganapathiraju, J. Hamaker, J. Picone, M. Ordowski, and G. R. Doddington, “Syllable-based large vocabulary continuous speech recognition,” IEEE Transactions on Speech and Audio Processing, Vol.9, pp.358–366, 2001.

[36] K. Maekawa, “Corpus of Spontaneous Japanese: Its design and evaluation,” Proc. of the ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition (SSPR2003), pp.7–12, 2003.

[37] “国立国語研究所. the corpus of spontaneous japanese [online],” 2011. http://www.ninjal.ac.jp/products-k/katsudo/seika/corpus/public/index_j.html.

[38] Y. Itoh, H. Nishizaki, X. Hu, H. Nanjo, T. Akiba, T. Kawahara, S. Nakagawa, T. Matsui, Y. Yamashita, and K. Aikawa, “Constructing japanese test collections for spoken term detection,” Proc. of INTERSPEECH 2010, pp.677–680, 2010.

[39] 元田浩, 麻生英樹, ” ニューラルネットワーク情報処理：コネクショニズム入門，あるいは柔らかな記号に向けて,” 人工知能学会誌, Vol.4, No.4, pp.470–471, 1989.

[40] R.O. Duda, P.E. Hart, and D.G. Stork, Pattern Classification (2nd Edition), Wiley-Interscience, 2000.

[41] 中野良平, ニューラル情報処理の基礎数理, 情報システム工学, 数理工学社, 2005.

[42] G.E. Hinton and R.R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” Science, Vol.313, No.5786, pp.504–507, 2006.

[43] G.E. Hinton, S. Osindero, and Y.W. Teh, “A fast learning algorithm for deep belief nets,” Neural Computation, Vol.18, No.7, pp.1527–1554, 2006.

[44] F. Rosenblatt, “The perceptron: A probabilistic model for information storage and organization in the brain,” Psychological Review, Vol.65, No.6, pp.386–408, 1958.

[45] F.A. Hayek, “The Sensory Order: An Inquiry into the Foundations of Theoretical Psychology,” the University of Chicago, 1952.

[46] D. Rumelhart, G. Hintont, and R. Williams, “Learning representations by back-propagating errors,” Nature, Vol.323, No.6088, pp.533–536, 1986.

[47] J. Hastad, "Almost optimal lower bounds for small depth circuits," Proc. of the Eighteenth Annual ACM Symposium on Theory of Computing, pp.6–20, 1986.

[48] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-stage architecture for object recognition?," Proc. of 2009 IEEE 12th International Conference on Computer Vision, pp.2146–2153, 2009.

[49] V. Nair and G.E. Hinton, "Rectified linear units improve restricted boltzmann machines," Proc. of the 27th International Conference on International Conference on Machine Learning, pp.807–814, 2010.

[50] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," Proc. of the Fourteenth International Conference on Artificial Intelligence and Statistics, Vol.15, pp.315–323, 2011.

[51] A.L. Maas, A.Y. Hannun, and A.Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," Proc. of ICML Workshop on Deep Learning for Audio, Speech and Language Processing, Vol.30 (1), 2013.

[52] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back propagating errors," Nature, Vol.323, pp.533–536, 1986.

[53] D. Epartement D'informatique, E. N, P. Esent, E. Au, F. Gers, P. R. Hersch, P. Esident, and P. Paolo Frasconi, "Long short-term memory in recurrent neural networks," Department of Computer Science, Swiss Federal Institute of Technology, Lausanne, EPFL, 2001.

[54] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp.1724–1734, 2014.

[55] Y. Kim, "Convolutional neural networks for sentence classification," Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp.1746–1751, 2014.

[56] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," Journal of Machine Learning Research, Vol.15, pp.1929–1958, 2014.

[57] T. Mikolov, W.t. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," Proc. of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp.746–751, 2013.

[58] T. Mikolov, K. Chen, G.S. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” Proc. of ICLR 2013, pp.1–12, 2013.

[59] L.F. D’Haro, O. Glembek, O. Plchot, P. Matejka, M. Soufifar, R. de Córdoba, and J. Cernocký, “Phonotactic language recognition using i-vectors and phoneme posteriogram counts,” Proc. of INTERSPEECH 2012, pp.42–45, 2012.

[60] S. Sukhbaatar, a. szlam, J. Weston, and R. Fergus, “End-to-end memory networks,” Proc. of NIPS 2015, pp.2440–2448, 2015.

[61] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L.u. Kaiser, and I. Polosukhin, “Attention is all you need,” Proc. of NIPS 2017, pp.5998–6008, 2017.

[62] Y. Akita and T. Kawahara, “Automatic comma insertion of lecture transcripts based on multiple annotations,” Proc. of INTERSPEECH 2011, pp.2889–2892, 2011.

[63] C. Parada, M. Dredze, D. Filimonov, and F. Jelinek, “Contextual information improves oov detection in speech,” Proc. of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp.216–224, 2010.

[64] 中谷良平, 岩橋直人, 中野幹生, 滝口哲也, 有木康雄, “未知語モデルを用いた CRF に基づく音声認識誤り訂正,” 日本音響学会 2012 年春季講演論文集, pp.67–70, 2012.

[65] M. Mohri, F. Pereira, and M. Riley, “Weighted finite-state transducers in speech recognition,” Computer Speech & Language, Vol.16, No.1, pp.69–88, 2002.

[66] 堀貴明, 塚田元, “音声情報処理技術の最先端：3. 重み付き有限状態トランスデューサによる音声認識,” 情報処理, Vol.45, No.10, pp.1020–1026, 2004.

[67] T. Moriya, T. Tanaka, T. Shinozaki, S. Watanabe, and K. Duh, “Automation of system building for state-of-the-art large vocabulary speech recognition using evolution strategy,” Proc. of 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), pp.610–616, 2015.

発表文献と本論文の関係

第 4 章の Deep Neural Network を用いた正解音素推定器と 6 章の正解音素推定器からの音声中の検索語検出と関係

1. Naoki Sawada, Hiromitsu Nishizaki, “Evaluation of DNN-based Phoneme Estimation Approach on the NTCIR-12 SpokenQuery&Doc-2 STD Subtask,” Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies, pp.211–216, 2016.6.（第 4 章，第 6 章）

2. 澤田直輝，西崎博光, “DNN による音素誤りパターン学習に基づく音声中の検索語検出”, 電子情報通信学会技術報告，SP2015-56，pp.33–38,2015 年 8 月．（第 4 章，第 6 章）

3. 澤田直輝, 西崎博光, “音素誤りパターンに基づく音声中の検索語検出の検討”，日本音響学会 2015 年秋季講演論文集，1-2-8，pp.21-22，2015 年 9 月．（第 4 章，第 6 章）

4. 澤田直輝, 西崎博光, “音声中の検索語検出のための回帰結合ニューラルネットワークを用いた正解音素推定”, 研究報告音声言語情報処理，2016-SLP-111，pp.1–5, 2016 年 5 月．（第 4 章，第 6 章）

第 5 章の時系列情報を考慮した正解音素推定器と 6 章の正解音素推定器からの音声中の検索語検出と関係

1. Naoki Sawada, Hiromitsu Nishizaki, “Recurrent Neural Network-based Phoneme Sequence Estimation using Multiple ASR Systems’ Outputs for Spoken Term Detection,” Proceedings of the 17th Annual Conference of the International Speech Communication Association (INTERSPEECH2016), pp. 3688–3692, 2016.（第 5 章，第 6 章）

2. Naoki Sawada, Hiromitsu Nishizaki, “Correct phoneme sequence estimation using recurrent neural network for spoken term detection,” Proceedings of the 5th Joint Meating of ASA and ASJ, 2016.11.（第 4 章，第 6 章）

3. 澤田直輝，西崎博光，“音声中の検索語検出のための双方向回帰結合ニューラルネットワークを用いた正解音素推定”，日本音響学会 2017 年春季研究発表会，2-Q-20，pp.189–190，2017 年 3 月．（第 4 章，第 6 章）

第 6 章の CRF を利用した音声中の検索語検出と関係

1. Naoki Sawada, Satoshi Natori, Hiromitsu Nishizaki, “Re-Ranking of Spoken Term Detections Using CRF-based Triphone Detection Models,” Proceedings of the 6th Asia-Pacific Signal and Information Processing Association Annual Summit and Conference 2014, 4 pages, 2014.12.（第 6 章）

2. Naoki Sawada, Hiromitsu Nishizaki, “Re-Ranking Approach of Spoken Term Detection using Conditional Random Fields-based Triphone Detection,” IEICE Transactions, Vol.E99-D, No.10, pp.2518–2527, 2016.（第 6 章）

3. 澤田直輝，名取賢，西崎博光，“2 つの手法を組み合わせた STD におけるクエリの特徴分類別性能調査と考察”，日本音響学会 2015 年春季講演論文集，1-P-3，pp.95-98，2015 年 3 月．（第 6 章）

第 7 章の正解音素推定結果からの単語変換器と関係

1. 澤田直輝，西崎博光，“正解音素推定器を用いた音素列からの単語変換器の検討”，日本音響学会 2018 年秋季研究発表会，1-R-12，pp.983–984，2018.9．（第 7 章）

学外発表

投稿論文

1. Kentaro Domoto, Takehito Utsuro, Naoki Sawada, Hiromitsu Nishizaki, “Spoken Term Detection Using Spoken Document Index Based on Keyword Corrected from Automatic Speech Recognition Result,” International Journal of Signal Processing Systems, Vol.4, No.4, pp.282-288, 2016.8.

2. Kentaro Domoto, Takehito Utsuro, Naoki Sawada, Hiromitsu Nishizaki, “Spoken Term Detection using SVM-based Classifier Trained with Pre-indexed Keywords,” IEICE Transactions, Vol.E99-D, No.10, pp.2528–2538, 2016.10.

3. Naoki Sawada, Hiromitsu Nishizaki, “Re-Ranking Approach of Spoken Term Detection using Conditional Random Fields-based Triphone Detection,” IEICE Transactions, Vol.E99-D, No.10, pp.2518–2527, 2016.10.

国際会議発表（査読付き)

1. Naoki Sawada, Satoshi Natori, Hiromitsu Nishizaki, “Re-Ranking of Spoken Term Detections Using CRF-based Triphone Detection Models,” Proceedings of the 6th Asia-Pacific Signal and Information Processing Association Annual Summit and Conference 2014, 4 pages, 2014.12.

2. Kentaro Domoto, Takehito Utsuro, Naoki Sawada, Hiromitsu Nishizaki, “Selection of Best Match Keyword using Spoken Term Detection for Spoken Document Indexing,” Proceedings of the 6th Asia-Pacific Signal and Information Processing Association Annual Summit and Conference 2014, 8 pages, 2014.12.

3. Kentaro Domoto, Takehito Utsuro, Naoki Sawada, Hiromitsu Nishizaki, “Two-Step Spoken Term Detection using SVM Classifier Trained with Pre-Indexed Keywords based on ASR Result,” Proceedings of the 16th Annual Conference of the International Speech Communication Association (INTERSPEECH2015), pp. 834-838, 2015.9.

4. Hiromitsu Nishizaki, Naoki Sawada, “Score Normalization using Phoneme-based Entropy for Spoken Term Detection,” Proceedings of the 7th Asia-Pacific Signal and Information Processing Association Annual Summit and Conference 2015, 8 pages, 2015.12.

5. Naoki Sawada, Hiromitsu Nishizaki, “Recurrent Neural Network-based Phoneme Sequence Estimation using Multiple ASR Systems’ Outputs for Spoken Term Detection,” Proceedings of the 17th Annual Conference of the International Speech Communication Association (INTERSPEECH2016), pp. 3688–3692, 2016.

6. Naoki Sawada, Ryo Masumura, Hiromitsu Nishizaki, “Parallel Hierarchical Attention Networks with Shared Memory Reader for Multi-Stream Conversational Document Classification,” Proceedings of the 18th Annual Conference of the International Speech Communication Association (INTERSPEECH2017), pp. 3311–3315, 2017.

国際会議発表（査読なし)

1. Hiromitsu Nishizaki, Naoki Sawada, Satoshi Natori, Kentaro Domoto, Takehito Utsuro, “Combination of DTW-based and CRF-based Spoken Term Detection on the NTCIR-11 SpokenQuery&Doc SQ-STD Subtask,” Proceedings of the 11th NTCIR Conference on Evaluation of Information Access Technologies, pp.402–408, 2014.

2. Naoki Sawada, Hiromitsu Nishizaki, “Evaluation of DNN-based Phoneme Estimation Approach on the NTCIR-12 SpokenQuery&Doc-2 STD Subtask,” Proceedings of the 12th NTCIR Conference on Evaluation of Information Access Technologies, pp.211–216, 2016.

3. Naoki Sawada, Hiromitsu Nishizaki, “Correct phoneme sequence estimation using recurrent neural network for spoken term detection,” Proceedings of the 5th Joint Meating of ASA and ASJ, 2016.

国内発表 (査読なし)

1. 澤田直輝，古屋裕斗，名取賢，西崎博光，関口芳廣，“STD システムへの音素間距離の導入方法の検討”，日本音響学会 2014 年春季講演論文集，3-Q5-11，pp.213-216，2014 年 3 月.

2. 米倉千冬，古屋裕斗，澤田直輝，名取賢，西崎博光，関口芳廣，“音声ドキュメントからの頻出発話語句の発見”，第 8 回音声ドキュメント処理ワークショップ講演論文集，pp.1-7，2014 年 3 月.

3. 堂元健太郎，宇津呂武仁，澤田直輝，西崎博光，“最良照合 STD による音声ドキュメント索引付けの評価および分析”，日本音響学会 2014 年秋季講演論文集，2-Q-9，pp.137-140，2014 年 9 月．

4. 澤田直輝，名取賢，西崎博光，“2 つの手法を組み合わせた STD におけるクエリの特徴分類別性能調査と考察”，日本音響学会 2015 年春季講演論文集，1-P-3，pp.95-98，2015 年 3 月．

5. 堂元健太郎，宇津呂武仁，澤田直輝，西崎博光，“音声認識結果から生成した補助的キーワード集合を利用する最良照合 STD”，日本音響学会 2015 年春季講演論文集，1-P-4，pp.99-102，2015 年 3 月．

6. 澤田直輝，西崎博光, “DNN による音素誤りパターン学習に基づく音声中の検索語検出”, 電子情報通信学会技術報告，SP2015-56，pp.33–38,2015 年 8 月．

7. 澤田直輝, 西崎博光, “音素誤りパターンに基づく音声中の検索語検出の検討”，日本音響学会 2015 年秋季講演論文集，1-2-8，pp.21-22，2015 年 9 月．

8. 堂元健太郎, 宇津呂武仁, 澤田直輝, 西崎博光, “認識結果から生成したキーワード集合を用いた分類器による最良照合 STD”，日本音響学会 2015 年秋季講演論文集，1-Q-18，pp.117-120，2015 年 9 月．

9. 澤田直輝, 西崎博光, “音声中の検索語検出のための回帰結合ニューラルネットワークを用いた正解音素推定”, 研究報告音声言語情報処理，2016-SLP-111，pp.1–5, 2016 年 5 月．

10. 中村卓磨, 澤田直輝, 西崎博光, “音素遷移ネットワークを用いたリアルタイムキーワードスポッティングの検討”，日本音響学会 2016 年秋季研究発表会，2-Q-11，pp.65–68，2016 年 9 月．

11. 澤田直輝，西崎博光，“音声中の検索語検出のための双方向回帰結合ニューラルネットワークを用いた正解音素推定”，日本音響学会 2017 年春季研究発表会，2-Q-20，pp.189–190，2017 年 3 月．

12. 中村卓磨，澤田直輝，西崎博光，“関連キーワードを使用したリアルタイムキーワードスポッティングの精度向上”，日本音響学会 2017 年春季研究発表会，2-Q-22，pp.195–196，2017 年 3 月．

13. 澤田直輝，増村亮，西崎博光，“マルチストリーム音声ドキュメントのための Parallel Hierarchical Attention Network の検討”，研究報告音声言語情報処理，2017-SLP-116，pp.1–7，2017 年 5 月．

14. 澤田直輝，増村亮，西崎博光，“コンタクトセンタ通話分類のための注意機構共有型ネットワーク”，日本音響学会 2017 年秋季研究発表会，1-10-12，pp.33–36，2017 年 9 月．

15. 澤田直輝，西崎博光，“複数認識システムの重要度を用いた正解音素推定器に基づく音声中の検索語検出の検討”，日本音響学会 2017 年秋季研究発表会，2-Q-10，pp.149–152，2017 年 9 月．

16. 齋藤友菜，澤田直輝，西崎博光，“音声中の検索語検出を用いた発話内容に着目した音声感情分類”，日本音響学会 2018 年春季研究発表会，2-Q-5，pp.145–148，2018 年 3 月．

17. 澤田直輝，西崎博光，“正解音素推定器を用いた音素列からの単語変換器の検討”，日本音響学会 2018 年秋季研究発表会，1-R-12，pp.983–984，2018 年 9 月．

18. 澤田直輝，齋藤拓哉，西崎博光，“音波形を直接入力とするニューラルネットワークを用いた音響イベント分類”，日本音響学会 2019 年春季研究発表会，1-P-1，pp.899–900，2019 年 3 月．

19. 大川正暉，澤田直輝，西崎博光，“事前処理不要な深層学習に基づく音楽・音声分類”，日本音響学会 2019 年春季研究発表会，1-P-2，pp.901–902，2019 年 3 月．

付 録A　正解音素推定器の構造実験

本付録では，正解音素推定器の構造を変更し最適な構造を調査した．

A.1　正解音素推定器の各モデル構造

正解音素推定器の構造は，全結合層は7層で固定，全結合層の下層にリカレント層，畳込み層を追加した場合の性能の違いを調査した．また，全層の次元数（チャンネル数）は128次元で固定している．

各モデルの名称で層の構造を示している．“GRU”，“BGRU” はそれぞれ単方向のGRU，双方向のGRUであることを示す．また，“CNN1” はカーネルサイズが3×1，“CNN2” はカーネルサイズが3×5，“CNN3” はカーネルサイズが5×1，“CNN4” はカーネルサイズが5×5という畳込み層を表している．また，それぞれの層の後の$\times N$はN層の同じ構造が続くことを表している．例えば，“CNN1×2-CNN2-DNN×7” である場合は，最初にカーネルサイズ3×1の畳込み層が2層あり，次にカーネルサイズ3×5の畳込み層が1層，そして全結合層が7層の構造を表している．

A.2　正解音素推定器のハイパーパラメータ

最適化手法は，“MomentumSGD” を使用している．“MomentumSGD” では，学習率が0.08，momentumが0.9，学習率は学習データに対する損失値が0.01下がらなかった場合に0.9倍ずつ下げていく．

各層の重みパラメータの初期値は，-0.1から0.1の正規分布に従ってランダムに決めている．

その他のパラメータは，Dropoutを20%で各層に行い，学習回数は検証用データに対して正解率が上がらなくなってから10エポックで終わる．

A.3　実験結果

コア講演音声に対する音素正解率を表A.1に示す．SDPWS講演音声に対する音素正解率を表A.2に示す．

実験結果から，リカレント層を導入した結果が比較的高い性能となった．特に2層の“BGRU” を導入した結果が比較的どの性能においても高い性能が得られた．また，畳込

表 A.1: 音素推定性能調査（コア講演：音素正解率）

DNN 名	音素正解率 [%]				
	1-best	2-best	3-best	4-best	5-best
DNN×7	90.2	95.3	96.5	97.2	97.7
GRU-DNN×7	90.7	95.6	96.7	97.4	97.8
BGRU-DNN×7	91.0	95.7	96.8	97.3	97.7
CNN1-DNN×7	90.6	95.8	96.8	97.5	97.9
CNN3-DNN×7	90.6	95.7	96.7	97.3	97.8
CNN2-DNN×7	90.6	95.4	96.6	97.2	97.7
CNN4-DNN×7	90.4	95.4	96.4	97.1	97.5
CNN1×2-DNN×7	91.1	95.9	96.9	97.5	97.9
CNN3×2-DNN×7	91.0	95.7	96.7	97.2	97.7
CNN1-CNN2-DNN×7	91.0	95.7	96.8	97.3	97.8
CNN3-CNN4-DNN×7	90.8	95.6	96.7	97.3	97.7
LSTM-DNN×7	90.7	95.5	96.6	97.2	97.6
BLSTM-DNN×7	91.3	95.9	97.0	97.5	97.9
BGRU×2-DNN×7	91.7	96.0	97.0	97.6	98.0
BLSTM×2-DNN×7	91.0	95.5	96.5	97.1	97.5
CNN1-BGRU-DNN×7	91.5	95.9	96.9	97.4	97.8
CNN1-BLSTM-DNN×7	91.4	95.7	96.7	97.3	97.8
CNN1×4-DNN×7	90.5	95.0	96.0	96.5	97.5
CNN3×4-DNN×7	90.9	95.6	96.7	97.3	97.8
CNN1×3-CNN2-DNN×7	91.0	95.7	96.7	97.3	97.8
CNN3×3-CNN4-DNN×7	90.9	95.5	96.5	97.1	97.6
CNN1×4-BGRU-DNN×7	91.4	95.9	96.9	97.5	97.9
CNN1×4-BLSTM-DNN×7	91.4	95.7	96.7	97.3	97.8

み層である “CNN” を導入した場合でも，単純な構造である “DNN” と比較して良い性能は得られている．しかし，“CNN” を導入した構造と比較して，“BGRU” や “ BLTM” を導入した構造のほうが高い性能が得られている．このことから，双方向のリカレント層，特に “BGRU” を導入することが高い性能になりやすいことが分かる．

表 A.2: 音素推定性能調査（SDPWS 講演：音素正解率）

DNN 名	音素正解率 [%]				
	1-best	2-best	3-best	4-best	5-best
DNN×7	83.1	90.7	93.9	95.3	96.0
GRU-DNN×7	84.9	91.0	93.2	95.5	96.1
BGRU-DNN×7	85.4	91.2	93.3	95.4	96.0
CNN1-DNN×7	84.0	91.1	93.6	95.6	96.2
CNN3-DNN×7	84.0	92.5	94.3	95.4	96.1
CNN2-DNN×7	84.0	91.1	93.3	95.2	95.9
CNN4-DNN×7	83.6	90.4	92.1	95.0	95.7
CNN1×2-DNN×7	84.5	91.2	94.4	95.6	96.2
CNN3×2-DNN×7	84.7	91.4	94.1	95.3	95.9
CNN1-CNN2-DNN×7	84.6	91.0	93.6	95.4	96.0
CNN3-CNN4-DNN×7	84.2	90.9	93.5	95.4	96.0
LSTM-DNN×7	84.9	90.8	92.5	95.2	95.9
BLSTM-DNN×7	85.6	91.7	93.9	95.7	96.3
BGRU×2-DNN×7	86.4	91.3	94.0	95.6	96.2
BLSTM×2-DNN×7	85.4	90.8	92.9	95.1	95.7
CNN1-BGRU-DNN×7	86.0	91.3	93.4	95.5	96.1
CNN1-BLSTM-DNN×7	85.8	91.2	93.4	95.5	96.1
CNN1×4-DNN×7	84.0	90.5	93.3	94.5	95.8
CNN3×4-DNN×7	84.3	90.9	92.9	95.4	96.0
CNN1×3-CNN2-DNN×7	84.5	91.0	92.8	95.4	96.1
CNN3×3-CNN4-DNN×7	84.4	91.2	93.8	95.2	95.9
CNN1×4-BGRU-DNN×7	85.7	91.2	93.1	95.6	96.2
CNN1×4-BLSTM-DNN×7	85.6	91.1	92.8	95.5	96.1

付 録B　日本語STD用テストコレクションのコア講演用未知語テストセットの50検索語

本論文で用いた日本語 STD 用テストコレクションのコア講演用未知語テストセットの 50 検索語を表 B.1 に示す.

表 B.1: コア講演用未知語テストセットの 50 検索語

モーラ	検索語	tf	df
13	石川島造船所	1	1
12	コンテキストディペンデント	5	1
11	クリントイーストウッド	2	1
10	ボスニア・ヘルツェゴビナ	1	1
	ユニバーサルスタジオ	3	2
	ホテルニューハンプシャー	2	1
9	春桜亭円紫	1	1
	談洲楼焉馬	1	1
	竹取物語	5	1
	高島平駅	2	1
	タンチョウの飛来地	2	1
	チトー大統領	2	1
8	スティーブンキング	1	1
	名犬ラッシー	2	1
	駒沢公園	8	1
	まほろば連邦	5	1
	南大泉	5	1
	伊曽保物語	2	1
	営団赤塚	1	1
	キラウエア火山	5	1
7	ユーゴスラビア	7	1
	代々木上原	2	2
	釧路湿原	3	2

表 B.1: コア講演用未知語テストセットの 50 検索語

モーラ	検索語	tf	df
7	コザクラインコ	4	1
	奄美大島	1	1
	オスマントルコ	6	1
	奥穂高岳	1	1
6	光が丘	9	3
	ノーベル賞	2	1
	西日暮里	7	1
	常盤平	7	1
	拝島駅	12	1
	本駒込	3	1
	メーンランド	2	1
	バンクーバー	4	2
5	アルバニア	9	1
	三河島	3	1
	美堀町	4	1
	屈斜路湖	3	1
	スリーピー	7	1
	ワイコロア	6	1
	九品仏	6	1
	NATO 軍	3	1
4	那覇港	2	1
	ネパール	27	1
	安保理	5	1
	ヒマラヤ	4	2
	知床	14	1
	八潮市	7	1
	ケベック	7	1

付 録C　NTCIR-11 SpokenDoc-2 タスクのmoderate-sizeサブタスクの100検索語

本論文で用いたNTCIR-11 SpokenDoc-2 タスクのmoderate-sizeサブタスクの100検索語を表C.1に示す.

表 C.1: moderate-sizeサブタスクの100検索語

モーラ	検索語	tf	df
18	ＷＷＥＲ最小化	9	1
12	質問応答システム	12	2
	サポートベクターマシーン	12	5
11	ＳＴＤの性能	9	2
	ＰＬＳＡモデル	12	1
	音声ドキュメント処理	25	17
10	機械翻訳モデル	5	3
	発話区間検出	10	2
	ＷＦＳＴ	6	1
	短時間スペクトル	10	1
	ＭＭＩシステム	19	1
	ジェフェリー情報量	4	1
9	パッセージ検索	16	3
	多項式カーネル	4	4
	マイクロフォンアレイ	19	3
	転置インデックス	9	2
	デモンストレーション	3	2
	背景と目的	12	10
	おはようございます	4	4
	ＳＰＳモデル	8	1
8	村山富市	4	1
	弁別特徴	12	5
	擬似三音節	12	1

表 C.1: moderate-size サブタスクの 100 検索語

モーラ	検索語	tf	df
	アーティキュレーション	7	1
	非可逆圧縮	4	1
	センシングデータ	4	1
8	カラオケ方式	9	1
	五体不満足	8	1
	キタちゃんキタロボ	7	1
	ＣＪＬＣ	12	5
	スピードワープロ	7	1
	情報工学	5	3
	ＭＲＲ	8	4
	ＱＡシステム	9	1
	高次モーメント	8	1
7	講義スライド	13	2
	プログラミング	8	5
	単語トレリス	9	2
	バタチャリヤ距離	18	10
	携帯電話	7	3
	ＭＰ３	13	3
	Ｐｏｄｃａｓｔｌｅ	11	3
	パワーポイント	12	9
	時論公論	9	3
	名古屋大学	5	3
	セミクローズド	6	1
	木構造辞書	11	1
6	ハンズフリー	6	1
	ウェーブレット	4	1
	マトリックス	7	5
	ＡＰＩ	7	4
	ＩＢＭ	7	3
	プロトタイプ	11	1
	ＳＬＰ	9	6
	ＧＭＭ	20	7
	ヒストグラム	14	6
	プライバシー	3	1
	産総研	3	2

表 C.1: moderate-size サブタスクの 100 検索語

モーラ	検索語	tf	df
	Ａｄａｂｏｏｓｔ	10	2
	バッファサイズ	7	1
	バイノーラル	4	2
	ＬＤＡ	21	2
	エントロピー	24	6
5	緑色	10	7
	聴診器	6	1
5	不完全	5	2
	エンドレス	4	1
	チューニング	5	2
	ｖｏｔｉｎｇ	24	1
	モダリティー	25	1
	プロポーズ	11	2
	非流暢	11	2
	ＮＡＭマイク	17	1
	大丈夫	8	6
	句読点	14	5
	ソーティング	4	2
	ウィキペディア	12	6
	ワイヤレス	15	5
	シーケンス	8	3
4	東北	5	4
	きらきら	3	1
	ラッシー	6	1
	爆発	10	6
	色相	4	1
	デフォルト	3	3
	シラバス	8	1
	キャプション	5	3
	折れ線	8	2
	東京	6	6
	ロボット	12	3
	中国	5	1
	ＳＰＯＪＵＳ	29	9
	投球	7	2

表 C.1: moderate-size サブタスクの 100 検索語

モーラ	検索語	tf	df
	三振	11	2
	声量	9	2
3	茶筌	9	6
	ブログ	12	1
	ＮＩＳＴ	5	3
	アニメ	14	2
	劣化	9	4