音声認識フレームワーク SPEECH RECOGNITION API の日本語学習アプリ開発への応用

クロス　尚　美

1．はじめに

大学等の高等教育機関や日本語学校では，「自然な発音・イントネーションで話す」ニーズが初級・上級を問わず非常に高い（松崎 寛　2016）にもかかわらず、限られた授業時間内で発音練習にかける時間がなく、実際には十分に音声教育実践が行われているとは言い難い（戸田　2009)。教師側の「発音が多少不正確でも、意味が通じれば問題ない」というビリーフがある一方で、学習者は「言いたいことが伝わらない」という経験をしており（戸田　2009)、それが発音上の問題であると学習者は認識している（戸田　2008)。学習者の不安と焦りは、なかなか解消しない。

学習者が自分たちの発音の問題を自覚していても、その矯正は容易ではない。対面式の授業時間に十分に組み込まれていないため、学習者は個々に自律学習として発音、発話練習を行わねばならない。従来の自律学習では、教科書に合わせたCD教材を活用するなどして、学習者がお手本を聞き、発音するという一方向性の学習形態が主流であったが、CD教材では正しい発音ができているかどうかのフィードバックがないため、自律学習の継続維持は難しい。留学生も含めて昨今の若者のパソコン離れ、書物離れ、スマホ依存が叫ばれて久しいが、今こそ双方向性を持つ新たな自律学習用教材、学習環境が求められている。従来のパソコンソフトではなく、スマホアプリとしての学習教材も、考慮に値するのではないだろうか。

音声認識の技術を取り入れた語学教材は、それ自体目新しいものではない。本稿では、従来の音声認識エンジンと、Appleが公開した音声認識フレームワークSpeech Recognition APIとを比較し、Speech Recognition APIの特徴を考え、その「音声認識」技術がどのように語学教育に貢献し得るかを考えるにあたり、日本語の音声認識状況を検証する。

2．音声認識技術の展開

筆者は、電子工学はもとより、音響音声学の知識もない。浅学のそしりを免れないことを承知で、河原（2015）に従い、これまでの音声認識技術の展開を概観し、大きく二つに分けることとする。まず、一般人が考える音声認識の機能を、音声認識システムとする。音声認識システムは、音声を音素レベルで解析する音響モデルと、「辞書」として言語のデータを蓄え、統計を使って確率で変換候補を見つける言語モデル（統計モデルとも呼ぶ）に分けることができる。これまで、音響モデルの充実に重点を置いていたものが、最近は辞書の充実に重きが置かれるようになってきているのではないだろうか。「辞書」の充実が、音声認識の精度へとつながる。「ニューラルネットワーク」ということばが身近に感じられるようになって久しいが、昨今の音声認識の技術は、このニューラルネットワークによって新境地にはいったようである。それが二つ目の音声認識であり、一つ目と違うのは、ICT の専門家でなくても最新鋭の音声認識システムにアクセスできるようになったという点である。例えば、Apple の Siri などは、音声を認識し、デバイスのそのほかの部分を起動させたりできる。AI（Artificial Intelligence）または人工知能と同義語に使われがちであるが、あくまでも音声を認識する「音声アシスタント」にすぎないとされる。Siri、あるいは他社の音声アシスタントの、音声認識を担うシステムが、アプリ開発のためのフレームワークとして独立したものが Speech Recognition API（Application Programming Interface）であると考えてよさそうである。Speech Recognition API とは、アプリと、音声認識システムとを介するプラットホームである。Speech Recognition API を介して、一般人が自らの音声認識エンジンを構築することなく、ニューラルネットワークに基づく巨大な音声認識システムにアクセスすることが可能になった。Speech Recognition API は Apple の他にも、Google や Microsoft など多数の企業が独自に開発している。

2.1. 音声認識システム

これまでの音声認識の流れは、音声情報の中から周波数特性を抽出し（音響モデル）、それと膨大な語彙データとを照らし合わせ、隠れマルコフモデル（HMM）の統計モデル（語彙データと統計モデルを合わせて、言語モデルとも呼ぶようである）とその改良版を用いて「認識」し（会沢　2015）、テキスト変換するというものであった。大量の発話を記録した学習用データから音声の特徴を蓄積し、認識対象となる入力音声から抽出された特徴と蓄積された特徴

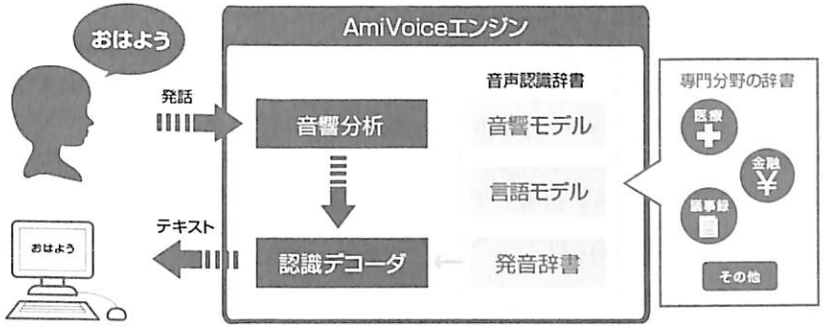
とを比較しながら、最も近い言語系列を認識結果として出力する。入力単語を最初の数音節 から予測する音声入力システムは、韻律情報の一種であるアクセント情報を入力音声 から抽出し、アクセント型の一致するものを上位 候補にすることによって、予測単語の絞込みを行なっている（荒木・大宮、2004）。

音響データを扱うのであるから、認識対象の音声情報の分析にも長けている。学習者の音声入力をその韻律情報に基づいて認識し、アクセントやイントネーションを視覚的情報に変換して出力することが可能となった。さらには、音声合成を行い、正しい発音やイントネーションの指導に役立つ教材の開発にも繋がる。

韻律情報の応用例としては、東京大学が公開する OJAD 日本語アクセント辞書と「韻律読み上げチュータスズキクン」が知られている。これらは学習者の発音とモデル音との波形やイントネーション曲線を示すことで聴覚的・視覚的フィードバックを与え、学習者の「気づき」を促す。

熊本県立大学の「ゆにおん」は、当初からスマホアプリとして開発されたものである。促音を「認識」する技術として、音声そのものではなく、音と音の間を計測している。「ゆにおん」もまた、その目的のために開発された音声認識システムを擁すると推察される。

また、企業が立ち上げた音声認識システムとして、アドバンスト・メディア社の開発した音声認識エンジン AmiVoice が挙げられる。当然ながら、これは有料のソフトウェアであり、身近なところでは会議の議事録などのアプリケーションが考えられる。



（出典　https://www.advanced-media.co.jp/amivoice）

発話が音声入力され、それをまず音響的に分析する。発話の波形を切り出し、特徴量を調べ、音素モデルを作成し、音の特徴を数値化するのである。分析されたデータを認識デコーダにかける。発話された声の特徴量が、どの音素モデルにどのくらい近いかが計算される。次に日本語の文を数多く集め、統計処理したものをベースにして、文字列や単語列が日本語として適切かを判定する。その際に用いるのは様々な活用場面に即した専門分野の辞書を持つ音声認識辞書である。認識デコーダは音声認識辞書を参照しながら、入力された発話に対して最も適切な分析結果をテキストとして出力する。

以上に挙げた韻律情報に基づく CALL 開発には、まず目的用途に特化した辞書を持つ音声認識エンジンを作らなければならない。従って、これまで音声認識、音声合成を行うための環境を作り出すのは、ICT のプロフェッショナルの領域であり、現場の言語教師が音声認識エンジンの開発に直接参加することは、極めて異例のことだったと言えよう。

2.2. Speech Recognition API の音声認識

Siri は音声認識、自然言語理解、命令の実行、返答の 4 つから構成されている音声アシスタントである。Apple が開発した機械学習テクノロジーが組み込まれているという点において、AI（人工知能）と呼べるかもしれない。前述の通り、本稿で取り上げる Speech Recognition API は、Siri の音声認識の部分を独立させ、Siri と同じ音声認識のネットワークにアクセスするプラットホームであり、音声を認識し、テキスト化することができる。Siri が公開以来、爆発的な進化を遂げてきたことからも明らかなように、Speech Recognition API のテキスト変換の精度も非常に高くなってきている。

本研究では、2016年 6 月の WWDC（Worldwide Developers Conference）で発表、公開された Apple の音声認識フレームワーク、Speech Recognition API を用いた。Google の Cloud Speech Recognition API に比べ、僅かながらも無償で利用できる時間が長く、回数、件数が多いことと、筆者らが手がける「漢越 Go!」アプリ（クロス、山崎、チャン、トラン　2017）開発時において、日本では Android 端末より Apple の iOS 端末の方が比較的優位にあったからである。

Apple Speech Recognition API は、発話データとデバイスの環境データを Apple に送信することで、リアルタイムで音声を認識し、テキスト化を行う。

Speech Recognition API の機能は、Mac などのおける文書作成ソフト上の音声入力の機能に通じるところがある。ユーザの名前、ニックネーム、アドレス

帳に登録された人名、ニックネーム、ユーザとの関係などのユーザ情報も、Apple に送信される。これらの個人データは、音声入力機能がユーザ本人を認識し、ユーザの発話を正しくテキスト変換するために用いられる。Apple では、Apple に送信されるその他のデータと関連付けられることはないとしているが、個人情報の保護については、懸念の残るところである。日々Apple によって収集されるデータをもとにした莫大な量の言語データは、ディープラーニングの糧となり、ニューラルネットワークに組み込まれていく。これまでの、専用音声認識システムとの大きな違いは、音声認識のブレーンとも言える言語モデルの部分が、常に進化しているということではないだろうか。しかも、そのブレーンにほぼ無償でアクセスができるのである。現場の語学教師としては、是非ともその言語教育への応用を考えたいところである。

3．SPEECH RECOGNITION API の音声認識・テキスト化機能

本稿では、音声認識・テキスト化機能の詳細を知るため、アプリのプロトタイプを試作し、検証を行った。検証に用いたのは、ひとまとまりの文章、単語（その多くが二字漢語）とその単語を含む短文あるいは語句である。

3.1. 文レベルの音声認識

まず、自己紹介における発話を想定し、以下の文を読み上げることでまとまりのある文レベルの音声認識状況を検証した。

自己紹介には筆者の個人情報を取り入れた。文中の英語（人名、地名）については、特に英語を意識せず、日本語風に発音した。

改行や句読点、引用符の指示も発話に取り入れた。

それを会話口調の、やや速いスピードで発話したところ、図1の結果が得られた。

私の名前はクロス尚美です（丸）姫路獨協大学で（点）日本語を教えています（丸）（改行）
月に一回（点）読書会が開かれます（丸）来月の課題は（点）天童荒太の（二重鉤括弧）あふれた愛（二重鉤括弧閉じる）の中から（点）（鉤括弧）とりあえず（点）愛（鉤括弧閉じる）です（丸）（改行）
夏休みに（点）英国のメルクシャムに住んでいるマーサ・ヤング・ショールテンさんが来日しました（丸）

個人名（クロス尚美、Martha Young-Scholten）と地名（Melksham）は、筆者のスマホ内の情報にアクセスした上で変換されたものと考えられる。固有名詞（姫路獨協大学）は、インターネット上の検索では旧字体を含めて正しく表記されるため、もともとSpeech Recognition APIの認識情報収集の範囲内であると考えられる。

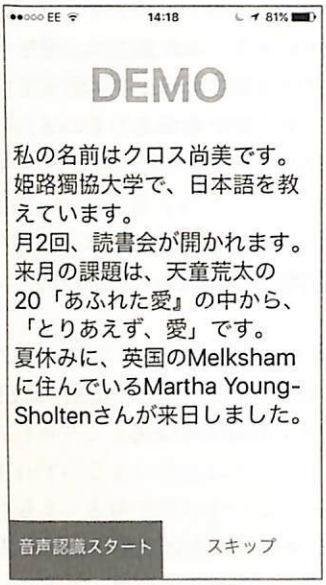


図1　会話文

作家名の「天童荒太」は、最初に「天童新た」と変換し、続く文面で著作名が入力されて後、正しく再変換された。固有名詞ではあるが、著名人であり、インターネット上でその情報が得やすいことが考えられる。

書籍名を示す「二重鉤括弧」は、開く方では認識されず、「20「」と表示された。「二重鉤括弧閉じる」は正しく認識、表示された。「二重鉤括弧」については、単独で、あるいは他の組み合わせで試行したが、正しく認識表示ができなかった。Speech Recognition APIであらかじめ用意されているべき文章校正の用語や記号のリストから漏れていることによる不具合であると考えられる。

3.2. 単語レベルの音声認識　同音異義語

次に、日本語に多い、漢語の同音異義語の認識について検証を行った。同音異義語の音声認識では、アクセント・韻律情報の他に、前後音素環境情報が必要である。従来の音声認識システムに基づく検証では、アクセント情報と韻律的情報を含む特徴パラメータとしてFBANK（filter bank, フィルタバンク対数パワー）を用いた実験の結果、アクセントと前後音素環境情報を用いたモデルと特徴パラメータにMFCC（Mel Frequency Cepstral Coefficient、メル周波数ケプストラム係数）を用いた方法が堀田・村上・池原（2006）によって紹介されている。不特定話者のほとんどの実験結果において、FBANKの同音異義語認識精度はMFCCと比べて低いが、特定話者においてFBANKを用いた同音異義語認識精度はMFCCと比べ高く、FBANKはMFCCより話者とモデルの依存度が高い結果となっているとのことである。この方法を用いて、当時としては非常に高い89%の精度が得られたとしている。このように、同音異義語の同定は非常に難易度が高いものである。

では、Apple Speech は同音異義語をどのように判別するのであろうか。

まず、同音異義語の例として、表 1 に「いし」、「こうしょう」、「こうせい」、「さんか」、「せいか」を挙げる。

表 1　同音異義語

1	いし	石、医師、意思、意志、遺志、縊死、遺址など
2	こうしょう	交渉、高尚、考証、口承、厚相、哄笑、公称、校章など
3	こうせい	後世、構成、恒星、更正、更生、公正、攻勢、校正、抗生など
4	さんか	参加、酸化、産科、傘下、賛歌、惨禍、讃歌など
5	せいか	成果、聖火、盛夏、製菓、聖歌、生花、正貨、生家など

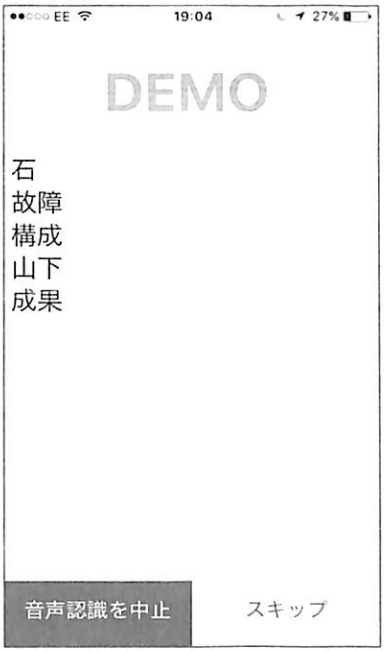


図 2　認識 1 回目

図 3　認識 2 回目

認証テストでは、「いし、こうしょう、こうせい、さんか、せいか」の 5 語を、特にアクセントを気にすることなく音声入力した。アクセントを意識しないで発音すると、それぞれ「石、交渉、構成、参加、成果」と同型となった。同じアクセント型で、2 回入力をなった。1 回目と 2 回目の結果は、以下の通りで

ある。

変換結果には、大きなと揺れが見られる。1回目では、候補のリストのうち「石」と「構成」と「成果」がテキスト化された。2回目では、「成果」のみテキスト化された。

1回目と2回目に、恐らくは人名の音読みと推察される「山下（さんか）、宏昭（こうしょう）、広瀬（こうせ（い））」がある。これら姓と思しき語は、筆者個人のアドレス帳になかった。音読みにしてまで優先的に人名に変換されるのは興味深い。

日本語には同音異義語が多いが、求める語を認識するには、まずアクセントを考える。日本語の標準的なアクセント（東京語アクセントとも呼ばれる）には、3つのルールがある。まず、第1拍と第2拍の高低が必ず逆になるというルールである。次に、一旦下がった（高から低に移行した）ら、同一語内では再び上がることはないというルールである。アクセントの下がり目（高から低に移行する拍）がアクセント核と呼ばれるが、アクセント核がない、つまりアクセントのない語が存在する。すなわち、3つ目は、アクセントのない語を許容するというルールである。日本語の語レベルのアクセントは、この3つのルールに支配され、アクセントは拍数＋1のバリエーションがあると言われる。表1からも明らかである通り、同音異義語をアクセント型のみで分別することは不可能である。

以上を踏まえ、表1に挙げた語のアクセント型は、それぞれ表2にあげる通りである。

表2　アクセント型

1　「石」以外、アクセント核は第1拍にある。つまり、「石」は第1拍が「低」であり、その他は「高」となる。
2　どの語も第1拍が「低」であり、アクセント核は語尾に来るか、語の中にない。
3　「後世」は第1拍にアクセント核があり、「高」から始まる。その他の語の場合は、アクセント核は語尾に来るか、語の中にない。
4　「参加、酸化、産科」は、第1拍が「低」であり、アクセント核は語尾に来るか、語の中にない。その他の語は、第1拍にアクセント核があり、「高」から始まる。
5　全て第1拍にアクセント核があり、「高」から始まる。

次に、表1の語群を作為的に二つの型に分けた。

① 第1拍を「高」とする(アクセント核が第1拍にある)

② 第1拍を「低」とする(語中にアクセント核がない/語の最終拍にある)

上の①と②のように意図的にアクセントの高低を操作した語を、①と②とを交互に5回ずつ10回ずつ入力したところ、以下の結果が得られた。

表4　同音異義語を独立語として発音した場合

	1		2		3		4			5
変換語	石	イシ	故障	宏昭	構成	広瀬	参加	山下	(再)山下	成果
変換回数	6	4	5	5	5	5	3	7	10	10
変換率%	60	40	50	50	50	50	30	70	100	100

(網かけは誤変換を示す)

1番の「いし」は、変換時に「石」と「イシ」の間に揺れが見られた。アクセントに関係なく、最終的には「石」6回、「イシ」4回で落ち着いた。2番の「故障」は、長音と単音の認識間違いが起こっている。同様に、3番の「広瀬」も、音読みした場合単音となるべきであり、認識間違いであると考える。4番については、当初「参加」とテキスト化しておきながら、後に全て「山下」と変換されている。5番のみ、問題なくテキスト化された。「宏昭、広瀬、山下」については、当初の変換テストでも同様の結果が得られている。Speech Recognition APIは、判断材料が不足する（単語レベルの変換）場合、人名に偏った変換を行うと考えられる。5番の「成果」は、アクセント型に関わらず10回とも正しく認識された。「せいか」については、同じアクセント型の同音異義語が多いが、おそらくはこれまでのSpeech Recognition APIの学習効果から、「成果」が使われるケースが多かったためこのような結果となったと考えられる。

3.3. 同音異義語の判別

これまでのテキスト化の検証結果から、単語を単独で発音した場合、語を正しく変換することは困難であることが分かった。それでは、正しく認識されるためには、どうしたら良いのであろうか。

次に、判別に必要最低限の情報（コンテクスト）を加え、短文として認識させることで、求める漢字変換ができるかを調べた。表1の同音異義語リストのうち、最初の2語について、それぞれ表5に挙げる短文として音声入力した。

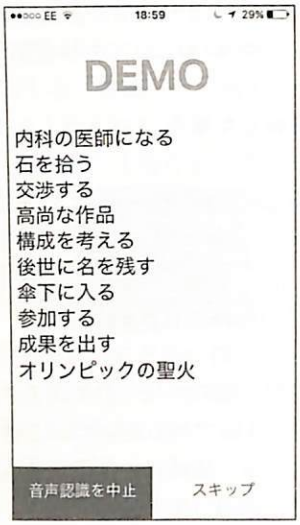


図5　短文の中の同音異義語

表5　短文の中の同音異義語

1	いし	医師	内科の医師になる
		石	石を拾う
2	こうしょう	交渉	交渉する
		高尚	高尚な作品
3	こうせい	構成	構成を考える
		後世	後世に名を残す
4	さんか	参加	参加する
		傘下	傘下に入る
5	せいか	成果	成果を残す
		聖火	オリンピックの聖火

短文としての音声認識は、アクセントを変えて2バージョンとし、それぞれ10回ずつ、合計で100回行なった。

1回目は第1拍を「高」（H）とし、2回目は第1拍が「低」（L）とした。まず、第1拍が高となるバージョンで同音異義語の2つの異なる例文を音声入力し、その後で第1拍が低となるバージョンを入力した。

短文レベルで音声認識を行なった場合、アクセントの違いは、認識結果に影響しなかった。すなわち、どちらのアクセント型で入力しても、正しく認識され、テキスト化された。また、単語ベースの認識テストの時に起こった、「交渉」と「故障」に見られるような長音と短音の認識間違いは起こらなかった。音声認識の精度は全て100%であったため、表は省略する。

音声認識において、10回ずつの認識テストがどの程度有意であるかは不明である。認識のもとになる言語データの辞書に当たるものが、Speech Recognition APIの場合、日々進化していると考えられるからである。今後、どのようなユーザデータが取り込まれるかによって、精度が高まることが期待される中、コンテクストの提示の仕方がそれに合わなくなることも考えられる。

4．まとめと今後の課題

Speech Recognition APIは、アプリ開発の段階でアプリに取り組むことで、一般のユーザ（ここでは個々のアプリ開発者）が高度な音声認識機能を利用することができるようになった点で、画期的である。

Speech Recognition APIは、コンテクストのはっきりした文レベルのテキス

ト化において、非常に高い精度の音声認識が認められる。同時に、アクセントやイントネーション、スピードにおいては、許容範囲が広い。一方で、単語レベルにおいては、同音異義語をアクセントなど韻律情報によってどの語であるか特定することはできない。テキスト変換には、音声認識の媒体となるアプリがインストールされているモバイル機器（ここではiPhone）から、個人情報が優先して使われることが分かった。これは、主にアドレス帳からの情報と考えられる。テキスト変換時にコンテクストが曖昧、あるいは特定できない場合は、人名に変換されることが多いことが分かった。個人情報の中に該当する人名がない場合にも、人名変換の優先度が高い。アプリ開発の段階で、日本語と英語の2言語に対応するように設定はできない。しかし、個人のアドレス帳に英語の情報があった場合、たとえ日本語風に発音されても、オリジナルの英語情報が英語表記のままテキスト化されることが分かった。

アクセントやイントネーションなどの音韻情報が不正確であっても、コンテクストが明確であれば、正しくテキスト変換される可能性が非常に高いと言える。発音、イントネーションに自信がなく、日本語母語話者と話をするのが苦手だという学習者にとって、「機械にも認知されるレベル」であることがわかれば、学習の励みになるのではないだろうか。多少アクセントに問題があろうとも、コンテクストさえしっかりしていれば（途中で諦めずにセンテンスの最後まで発音すれば）、十分に日常のコミュニケーションに対応できるのかもしれない。

従来の音声認識システムを用いたCALLのように、学習者の発音の「どこがどのように間違っているか」という情報が提示されない点では、課題が残る。しかし、明示的フィードバックがあっても、すぐに発音矯正になることは少ないこと、ネガティブなフィードバックよりポジティブなフィードバックが好まれ、それが学習上有意であるとされることを考えると、学習者にとって自分の発音が認識されたという意識は有効なのではあるまいか。

Speech Recognition APIの最大の特徴は、その膨大な言語データの集積と、さらなる学習機能である。本稿では、個々のユーザレベルにおいてもテキスト変換に「学習」効果が見られるかどうかの検証には至っていない。今後の課題の一つとしたい。

参考文献

会沢 純将　2015　「ＨＭＭを用いた音声認識」日本大学理工学部　学術講演会論文集　pp.459-460

荒木雅弘、大宮広義　2004　「韻律情報を利用した予測型音声入力システム」第10回年年次大会発表論文集 pp. 4-6
河原達也　2015.「特別講演　音声認識技術の展開」信学技報　115（388), pp.111-116
クロス尚美、山崎恵、ディン・ティ・トゥー・チャン、トラン・バン・タン　2017「漢越語を生かした発音矯正・語彙習得のためのスマホアプリ開発に向けた日越音声コーパスの構築」『CASTELJ 予稿集』pp224-227
戸田貴子（2008)「日本語学習者の音声に関する問題点」『日本語教育と音声』第２章　くろしお出版　pp.23-41
戸田貴子（2009)「日本語教育における学習者音声の研究と音声教育実践」『日本語教育』142号　pp.47-57
堀田 波星夫・村上 仁一・池原 悟　2006.「アクセントを用いた同音異義語の不特定話者音声認識」『電子情報通信学会技術研究報告. SP、音声 105（686), pp.65-70
松崎 寛　2016　「日本語音声教育における韻律指導　－CALL システムを用いた教材開発の動向－」日本音響学会誌72巻４号　pp.213-220

参考 URL

OJAD（Online Japanese Accent Dictionary）
http://www.gavo.t.u-tokyo.ac.jp/ojad/ 参照 2017/09/05
ゆにおん - ユニティちゃんと日本語発音練習
https://play.google.com/store/apps/details?id=jp.ac.puk.union　参照 2017/09/05

What Can "Speech Recognition API" Do for Japanese Language Teaching?

-A Study of the Latest Sound Recognition Technology-

Naomi CROSS

In CALL (computer assisted language learning) the automatic sound recognition engine has been recognised as an interactive method for pronunciation training. Most CALL programmes to date use prosody dependent speech recognition supported by a large vocabulary. Whilst automatic speech recognition has played an important part in developing CALL programmes for prosodic exercises, many are technically demanding and are beyond the easy reach of frontline language teachers. Very recent developments, however, are combining sound recognition technology with AI (Artificial Intelligence). This has generated a profound leap in sound recognition accuracy. It would appear that the accuracy of sound recognition is no longer dependent on accurate prosodic information input. The latest Speech API (Application Programming Interface) released by Apple and Google for "app" (application, particularly for mobile devices) developers, for example, seem to work with inaccuracies of input prosody such as intonation and accent variances. This paper focuses on two features of Speech API; first the potential for frontline language teachers to incorporate the technology readily into their day to day teaching and second, the tolerance to variances in speech input can be seen as an advantage in certain language teaching situations for enhancing and reinforcing language learner motivation. Using Apple's Speech API, this paper examines the ways in which speech data may be converted into text.