特集《AI 技術の権利化》

音声対話システムの構成と今後

大阪大学 産業科学研究所　**駒谷　和範**

要　約

音声認識技術に基づいて人間と言葉を使って対話をするシステムが広く使われ始めている。スマートフォン上の音声応答アプリや対話ロボット，最近ではスマートスピーカと呼ばれる据置型のシステムも発売されている。本稿ではまず，音声対話システムの構成やその各部分において行われる処理について概説する。この処理のうち，入力処理と呼ばれる音声認識や言語理解について，音声認識技術のこれまでの進歩や，言語理解で用いられている技術を説明する。さらに，音声対話システムでは，単に言語内容をやりとりするだけでなく，発話のタイミングや，実環境内での対話相手との位置関係，さらには複数の参加者の中での社会的関係など，言語内容以外にも考慮すべき要素がある。これら現在のシステムではあまり扱われていないが必要となるであろう要素技術についても触れる。最後に，人工知能技術一般を含めた今後の展望を述べる。

目次

1．はじめに
2．音声対話システムの基本構成
3．音声認識技術の進歩
4．言語内容の理解
（1）　対話管理部を持つシステムの場合
（2）　一問一答型システムの場合
5．言語内容以外の理解
（1）　円滑な話者交替
（2）　音の到来方向の推定
（3）　複数人対話における課題
6．今後の展望

1．はじめに

人間の発話を入力として，それに対して応答を返すシステムが一般ユーザにも広く使われ始めている。音声認識技術に基づき，人と言葉を使ってやりとりするシステムを音声対話システムと呼ぶ。有名なものとして，Siri やしゃべってコンシェル，音声アシストなどのスマートフォン上の音声応答アプリや，Pepper のようなロボットが挙げられる。さらにここ数年では，Amazon Alexa や Google Home，LINE Clova のような，据置型のスマートスピーカも発売され，一般に使われ始めている。

現在，人工知能は第 3 次ブームにある。人工知能という用語は，1956 年に米国・ダートマスで開催されたダートマス会議で初めて提唱されたものであり，半世紀を超える歴史を持つ。第 1 次ブームはこのダートマス会議直後の 1950 年代後半から 1960 年代，第 2 次ブームは 1980 年代と言われ，それぞれにおいて人工知能に対する期待と失望が繰り返されてきた。人工知能とは，ダートマス会議での定義では，the science and technology of making intelligent machines, especially intelligent computer programs，つまり「知的な機械（とりわけ知的なコンピュータプログラム）を作る科学と技術」であり，元々は「電気工学」「土木工学」などと同様の，研究分野を指す用語である。近年では知的なシステムを指す語としても使われているが，単一のシステムや技術を指すわけではなく，ましてや「AI が○○をした」という文例のような，意図を持った特定の主体を指すものでは本来ないことに注意が必要である。

本稿では，人工知能の一分野である音声対話システムについて，その基本構成や課題を述べる。音声対話システムは，人間が普段使う音声言語を入出力とすることから，人工知能技術の中でもあたかも人のような主体として扱われることが多い。しかし，その内部での処理は，音声認識や自然言語処理といった昔から研究が続けられてきた分野における問題と同様であり，それが近年急速に進展した機械学習などの技術を組み合わせることで一般での使用が可能な水準で実現され

たものである。したがって，以降の各節で述べるように，その構成要素は具体的な個別の技術であり，決してブラックボックスではない。このことを音声対話システムという人工知能技術の一事例を通して理解することで，人工知能技術の権利化や，ひいては人工知能技術を我々の社会でどう受容するかを議論するための理解の糸口となることを期待する。

以降の節の構成を示す。まず2節では，音声対話システムの基本構成を述べ，音声対話システムの処理の概要を示す。続く3節と4節ではこの音声対話システムを構成する要素のうち，入力理解部分に関して掘り下げる。具体的には，3節では音声認識技術のこれまでの発展について述べ，4節では言語理解部の処理について述べる。これらは現在のスマートフォン上の応答アプリなどでも使用されている技術である。続いて5節では，図1に示す処理の枠組みには含まれていないが，今後重要となると思われる要素技術について述べる。6節では本稿のまとめとして今後の展望を示す。

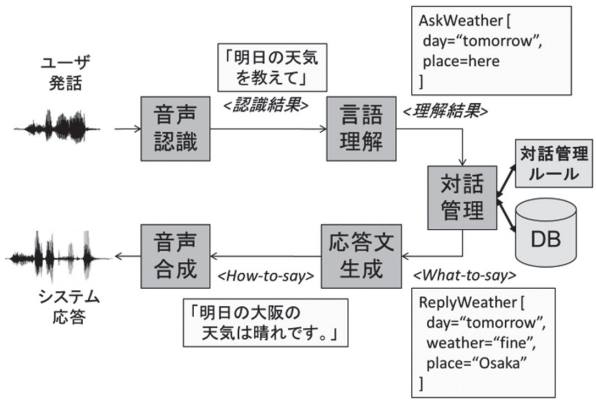

図1：音声対話システムの5つの処理

2．音声対話システムの基本構成

音声対話システムにおけるおおまかな処理の流れを図1に示す。主に，音声認識，言語理解，対話管理，応答文生成，音声合成の5つの処理により構成される[(1)]。

まず音声認識部では，ユーザの話す音声信号をテキスト（単語列）へと変換する。音声認識技術は現在一定の成熟を迎えており，音声対話システムに限らず様々な応用がある。例えば，日本の国会の議事録作成は音声認識を使って行われている。もちろん議事録に音声認識誤りが含まれてはならないので，人手での後修正を経て議事録とされている。これは，これまで速記官が人手で書き記していたのを，音声認識を道具として使って行っていることに相当する。また，コールセンターのモニタリング（自動応答ではない）にも使われている。コールセンターに電話をかけると「品質向上のために録音しています」というアナウンスが流れるが，その録音を事後に全て人手でチェックしているのではなく，例えばオペレータが不適切な単語を話していないかの確認や，オペレータが話した単語と顧客満足度との関係等を自動でチェックするのに使われている。これらの応用は，音声認識結果を顧客にそのまま見せるわけではないのに対して，音声対話システムにおける音声認識では，認識結果に基づいてその場でリアルタイムに応答する必要があるという点で難しい。また，周辺の雑音や，ユーザの発音が明瞭でないといった問題も起こりやすい。

次に，言語理解部では，音声認識により得られた単語列に対して，システムが応答するために必要な内容を得る。例えば図1の例では，「明日の天気を教えて」という入力に対して，「明日」が日付であることや，天気について尋ねていることを得る必要がある。この例では，場所についてユーザ発話に言及がないため，デフォルトで“here”としている。これは一問一答型のシステムにおける言語理解でしばしば行われる処理である。言語理解部での処理については，4節で詳しく述べる。

言語理解部での結果は，次の対話管理部に渡される。ここまでの「音声認識」と「言語理解」を合わせて入力理解と呼ばれるが，これだけではその発話以前に行われた発話の履歴を考慮して応答できない。例えば以下のような対話の例を考える。Uで始まるのがユーザの発話，Sで始まるのがシステムの発話である。

U1：明日の大阪の天気を教えて

S1 ：晴れです

U2：最高気温は？

S2 ：16℃です。

S2のような応答を実現するためには，U1の発話の結果得られた「日付は明日」「場所は大阪」という情報を保持し，U2の解釈で利用する必要がある。このような情報を含め，対話管理部ではシステムの内部状態が保持され，管理される。単純には，過去の理解結果を全て保持し，その後に入力された理解結果を逐次追加（同じ項目に対応する内容が入力された場合には上書き）するという処理が考えられる。この上書きの例として，上の例のU2が「名古屋は？」だった場合，場所の内容を上書きして「日付は明日」「場所は名古屋」の天気を答えるというものが挙げられる。しかし

ながら，音声対話システムでは音声認識結果に誤りが含まれることがあるため，その可能性を考えたうえで履歴を管理する必要がある。（さもないと，一度誤った音声認識結果が入力されると，その内容をその後ずっと引きずることになる。）このため，後述する一問一答型のシステムでは，履歴を基本的には管理しないことで，システムを単純化している[5]。また図中の対話管理部の部分では，例えば天気検索の場合は天気のデータベース（DB）にアクセスするなどして応答内容を得るといった処理も行う。

応答文生成部では応答内容を表す具体的な文を生成する。実際には，単純にテンプレートに基づいて応答文生成を行うことが多い。例えば，「day の place の天気は weather です。」のようなテンプレートを用意しておき，ここに対話管理部から得られるキーワードを代入して応答文とするという処理である。つまり，対話管理部からは，どのテンプレートを使うのか（応答の対話行為）と，その内容に相当するキーワードが渡され，それを表す文を生成することになる。また，対話システムの応答にキャラクタ性やスタイルを付与したい場合，例えば常体と敬体の使い分けや，老人っぽさや子供っぽさのようなキャラクタ性を持ったシステムを作りたい場合には，この部分での設計や処理が必要となる。

最後に音声合成部では，生成された応答文のテキストを音声信号に変換する。これは音声合成（Text-to-Speech；TTS）と呼ばれる処理であり，この分野の進展も著しい。発話の明瞭性という観点では概ね実用に足る性能が達成されており，固定的なアナウンス（例えばバスの車内案内）などでは既に広く使われている。対話システムという観点では，応答文のキャラクタ性に合致した声質の実現や，感情（喜怒哀楽や，ためらって話す感）を有する声の合成など，より多様な合成音声を作ることが課題である。

3．音声認識技術の進歩

音声認識とは，人の話す音声信号を入力として，それに対応する文字列や単語列を出力する処理である。この技術的な詳細については他にも優れた解説が多くあるため（例えば(2)），ここではその歴史や取り巻く状況について概観する。

音声認識の研究は古くから行われており，日本でも1960年前後からパターン認識の一分野として研究が続けられてきた[3]。2000年前後にはPCの普及の流れを受け，IBM社が一般ユーザ向けの音声認識ソフトウェア ViaVoice を開発して販売し，一時テレビCMにも登場していたが，広く使われるには至らなかった。その理由として，使用の前に話者ごとに数百文程度の音声を録音するエンロールと呼ばれる作業が必要であったことなど，当時の技術が未成熟であったことが挙げられる。また，持ち運びしにくいデスクトップ型PCが当時は主流であり，これを使って音声認識を行うというシチュエーションが限定的であったことも一因であろう。

2000年代後半から2010年代にかけて，クラウドコンピューティングに基づくクラウド型音声認識が登場し，音声認識システムの開発運用パラダイムは大きく変わった[4]。それまではユーザの手元にあるコンピュータ上で音声認識処理が行われていたが，クラウド型音声認識では音声データをサーバに送り，サーバ上で音声認識を行う。これによる運用上のメリットは非常に大きい。ユーザはプログラムや認識用モデルを端末にダウンロードしたりアップデートしたりすることを強いられずに，既存の音声端末（例えばスマートフォン）から音声認識を気軽に使うことができる。また辞書などのモデルもサーバ側にあるため，システム開発者が随時アップデート可能である。さらに音声データはサーバに送られるため，開発者から見るとデータ収集が極めて容易であり，収集したデータを用いた機械学習による性能向上に使うことができる。上記のパラダイムシフトにより，スマートフォンなどで使える一般ユーザ向けの音声認識アプリが多数登場した。ここ数年（2010年代後半）では，スマートスピーカと呼ばれる家庭用の商品も登場している。

技術的には，ここ数年は深層学習を用いた技術が席巻している[2]。音声認識の分野では，音声認識率（認識結果に占める正解単語の割合）という明確な評価尺度があり，正解データ付きの大規模音声データベースが共有されていることから，深層学習の適用が可能である。端的に言えば，学習データ量が多いほど，またそのデータが実使用に近い環境で得られたデータであるほど，音声認識の性能は高くなる。このため，利用ユーザが増えるとデータが集まり，データが集まると性能が上がるのでさらにユーザが増える，という正のスパイラルが回り始めると，性能向上は加速する。

現状，これらの技術を実現したサービスはほぼ無料

で提供されることが多い。例えばスマートフォン上の音声アプリは無料であり，スマートスピーカもほぼハードウェアの価格のみでソフトウェアの価格は含まれていないように見える。各企業におけるこのビジネスモデルの目的は，性能向上のための音声データの収集や，自社の技術力の誇示によるブランド価値の向上，また自社の他サービスへの導線とするユーザインタフェースの配布，などが考えられる。一方で，無料で提供されているのは一般的なモデルに基づくサービスであるため，例えば自社の商品名のような固有名詞を正確に認識させるなどのカスタマイズがないと実用的には使えない状況も多い。このようなカスタマイズや導入支援を行うというビジネスも価値を増してくると思われる。

4．言語内容の理解

言語理解部は，音声認識結果として得られる一発話（単語列）を入力として，システム応答に必要な内容を取り出す。本節では，対話管理部を持つシステムにおける具体的な処理をまず説明する。その後に，近年広く使われ始めている一問一答型システムにおける処理についても触れる。

なお，上記の「入力とされる単語列から取り出す」という処理ではカバーされない処理もある。例えば「京都でこの週末に空いているホテルを教えて」というユーザ発話を考えよう。ホテルを検索するには，「この週末」という表現が具体的な日付に変換される必要がある。この際，まず週末が土曜と日曜を指すことや，さらにはこの状況では「前の週末」ではなく「次の直近の週末」を意味することも，システムに明示的に与える必要がある。このような知識は，人間にとってはほぼ常識であるが，常識であればあるほど人間がそれを口にすることは少なく，事例から収集した学習データの中には明示的に現れないことが多い。このため，深層学習を含む機械学習で全てが解決できるわけではなく，実用的にはルールを書いて補う部分も依然必要である。

（1） 対話管理部を持つシステムの場合

音声対話システムにおける言語理解とは，事前に定義されたキーワードやユーザ意図を，入力である単語列から同定するという処理である。一般的に想定される「人間が相手の言うことを理解する」ということとはやや乖離があることに注意が必要である。この言語理解結果は，対象とするタスクやドメインに依存して定められる。タスクとは検索や予約などシステムが遂行すべき課題を指し，ドメインとは天気予報やホテル情報などの話題である。

具体例として，「明日の大阪の天気を教えて」という発話に対する言語理解結果を考える。ここでは，「明日」が日時を表すことや，「大阪」が地名を表すこと，さらにはこの発話全体で天気予報を検索しようとしていることを理解する必要がある。つまり，AskWeather(Place=“大阪”，Date=“明日”)のような内容を取り出すことに相当する。これらの Place や Date はこのドメインにおいて事前に定められたスロットであり，「大阪」や「明日」はそれらの値である。このように，スロットとその値の組を見つける処理をスロット値同定と呼ぶ。また AskWeather() はユーザ意図であり，これを見つける処理を意図同定と呼ぶ。このユーザ意図やスロットとその値は，事前にシステム開発者により，ドメインやタスクごとに，次にシステムがどのような処理を行うべきか（ここでは2つのスロット値の条件で天気予報を検索する）などに対応させながら設計される。図2の例のように，データベースを検索するタスクの場合，スロットはデータベースの構造に合わせて定義され，またその値はデータベース中のエントリに対応させることが多い。なおこの例では，Rating に関する“評判がいい”という表現とデータベース中の値（図中では fair や good など）との対応を取る必要があるが，これは一般的には人手で設定される。

まとめると，音声対話システムにおける言語理解は，①意図同定と②スロット値同定という処理からなる。この例を図2に示す。

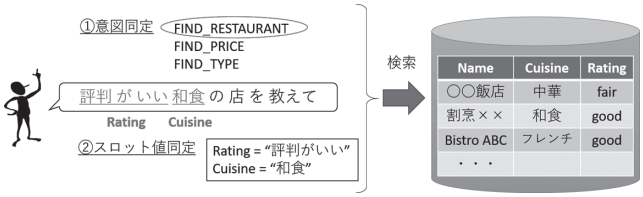

図2：音声対話システムにおける言語理解の概要

①の意図同定は，機械学習における多クラス分類問題として定式化できる。多クラス分類問題とは，入力（ここでは一発話）が，事前に定義した発話意図のうちのどれに対応するかを分類する問題である。入力は単語列であり，ここから各単語の表記や品詞の有無

（頻度）を特徴として，機械学習の入力として用いる。多クラス分類問題の概要を図3に示す。

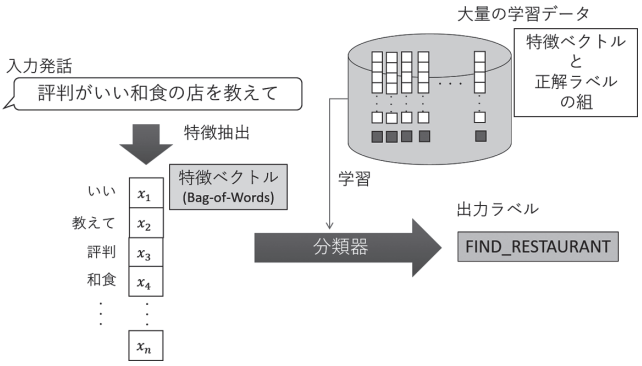

図3：多クラス分類問題の概要

図3の例では，単純に，入力発話内に含まれる単語の有無により特徴ベクトルを構成している。これは単語の並び順を考慮しないモデルであり，Bag-of-words モデルと呼ばれる。この場合，特徴ベクトルの次元数は，学習データに出現した単語の異なり数となる。また単語の有無に加えて，単語の品詞や，局所的な単語の二つ組や三つ組などを特徴に加えたり，近年では分散表現と呼ばれるベクトル表現に変換したものを特徴ベクトルとしたりすることも多い。この特徴ベクトルと正解ラベルの大量の組を学習データとして事前に分類器を学習しておき，実際に発話が入力された際には学習済みの分類器を使ってラベルを出力する。これは自然言語処理における典型的な分類問題の応用であり，例えば迷惑メールの自動判定と同様の問題設定である。分類器には，Support Vector Machine（SVM）や最大エントロピーモデル，さらに近年では深層学習に基づく分類器が用いられる。

一方，②のスロット値同定は，機械学習における系列ラベリング問題として定式化できる。これは，①の意図同定のように発話全体に対して一つのラベルを推定するのではなく，発話内の各単語にラベルを付与するという問題である。なお，単語が持つ辞書的知識（例えば「明日」は日付以外に使われることはないという知識）で解決できる場合も多いが，この一方で例えば「駅のそばに ATM はありますか」という入力に対して，「そば」という文字列から蕎麦を意味すると理解するのは不適切であるように，発話の単語系列全体を見て解釈する必要がある。

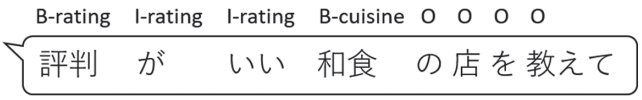

図4：系列ラベリング問題の出力例

系列ラベリング問題の出力例を図4に示す。ここでは，連続する複数の単語が一つのスロットに対応することがあるため，IOB2 と呼ばれるラベル付け方法が用いられることが多い。つまり，B を開始点（beginning），I をその内部（inside），O を外部（outside）として，B と I については対応するスロット名をその後につけて出力ラベルとする。これにより図4の例では，rating=“評判がいい”，cuisine=“和食”というスロット値であることが表現されている。予測性能を考慮して経験的に，スロット値の最終単語を（例えば記号 E などで）明示しないこのラベル付け方法が用いられることが多い。これも①と同様に，各単語から特徴ベクトルを得たうえで，上述した正解ラベル列を付与した大量の学習データを用意して，系列ラベリングを行うプログラム（正確にはそのモデルが持つパラメータの値）を学習する。系列ラベリング問題は，単語列に対して各単語の品詞を推定するのと同様の問題である。系列ラベリングのためのモデルとしては，Conditional Random Fields（CRF）が有名である。また近年では，十分な量の学習データが得られている場合，ニューラルネットワークの一形態である Recurrent Neural Network（RNN）や，さらにその特殊形である Long Short Term Memory（LSTM）を用いる場合も多い。

ここまでで述べた，①意図推定と②スロット値同定は，いずれも自然言語処理分野における機械学習を適用した問題である。より詳しい説明は例えば(6)のような良い入門書から得ることができる。

（2） 一問一答型システムの場合

ここまでで述べた入力理解部における処理は，発話から得た理解結果を蓄え，対話の履歴を管理しながら話すシステムで用いられることが想定されてきた。しかしながら，システムとの音声対話では，音声認識誤りや言語理解誤りが避けられない。対話状態追跡と呼ばれる履歴の管理を適切に行うための研究も行われているが，システム設計が複雑になることや，誤解が生じた場合にその解消が難しいことから，現在一般に使われているシステムではあまり使われていない[(5)]。

これに対して，現在スマートフォン上の音声応答アプリやスマートスピーカでは，履歴の管理を放棄し，入力されたユーザ発話を対象として，それに対してどう答えるかに注力したものが多い。これを一問一答型

システムと呼ぶ。図１で示したような対話管理は行わず，入力理解結果に応じて用意した応答文を，音声合成部に直接渡すことで応答を行う。

m	質問 Q_m	応答 A_m（動作，出力）
1	こんにちは	「どうもこんにちは」
2	愛してるよ	「ありがとう。でも仕事に戻りましょうよ。」 「まあ！でもみんなにそう言ってるでしょ。」
3	週末の大阪の天気は	天気検索（大阪，週末）
4	天気はどうなるかな	天気検索（現在地，今日）
5	タイマーをDURにセット	タイマー起動（DUR）
6	富士山の高さは？	質問応答モジュール（富士山の高さは？）
7	STATIONに行きたい	乗換案内（STATION，最寄駅）
…	． ． ．	

図５：応答データベースの例

一問一答型のシステムでは，内部では図５に示すような応答データベースを持っていると思われる。例えば，ユーザの発話が「こんにちは」であった場合には，それに最も近い質問 Q_m を探し，それに対応した応答 A_m を出力したり，それに対応した動作を実行したりするというものである。つまり入力の理解は，入力に対して最も近い質問を同定するという問題になる。最も単純には，入力文と質問 Q_m のそれぞれを単語に分割し，その間で共通する単語が多いものを選ぶという基準も考えられる。これは Web 検索で，いくつかのキーワードを入力して Web ページを検索する場合と同様の問題設定である。このため，単語への重みづけなどの技術が利用されたり，このタスクに最適となるようなベクトル空間への写像を学習するといった手法が開発されたりしている。

図５の例には，DUR や STATION のような，それぞれ時間の長さや駅名に対応する変数がある。これらの値は，入力文に対して系列ラベリングによりスロット値同定のような処理を行うことで得られる。つまり一問一答型システムにおいても，対話管理部を持つシステムの場合と同様の言語理解結果を利用する場合も多い。

また図５において「天気はどうなるかな」という質問に対して，場所を現在地，時間を今日としたり，「STATION に行きたい」という質問に対して，出発地を最寄駅としたりしている。これは検索を行うには情報が不十分であるものの，それをシステム側から逐一質問することはせず，デフォルト値を入れてひとまず回答することに相当する。対話管理部を持つシステムの場合は，検索に必要な内容が含まれていない場合には対話的にそれをシステム側から質問するように設計されることが多いが，履歴の管理をほぼ放棄した一問一答型ではひとまず出力を行い，出力がユーザの意図と異なる場合は，再度ユーザが条件を明示的に含めた発話をしてくれることを期待している。一発話に入る程度の短い内容であれば，履歴の管理を誤るリスクを減らせるため合理的な設計と言える。

このような一問一答型のシステムは，「この質問にはこう答えればよい」といった，簡単で定型的なタスクでは有効である。一方で，実際の人どうしでの対話では，もう少し込み入った内容について，それまでに話したことを考慮しながら，発話の解釈や応答が行われる。一問一答ではなく，それまでの履歴や文脈を考慮して，長く続く対話を実現する音声対話システムは，未だ研究途上にある。また近年では，対話への深層学習モデルの導入も盛んに研究されている[(7)]。例えば機械翻訳と同様のモデルを用いて，対話データから直接的に質問に対する応答を学習しようとする試みも盛んに行われている。

５．言語内容以外の理解

音声対話システムが対話を円滑に進めるには，音声認識結果に含まれる言語内容に加えて，音声対話に特有な，言語内容以外の要素も理解する必要がある。具体的には，まず発話のタイミングの管理が挙げられる[(8)]。単純な一問一答では扱われていないが，人間と行うような自然な対話を目指す場合には，発話タイミングの問題は避けて通れない。次に，音の到来方向の推定が挙げられる。実空間中に存在するシステムでは，話している人の方向を向くことは必須の要求仕様である。さらには，対話に参加するユーザが一人ではなく複数である場合には，全てのユーザ発話に対して必ず応答する必要はないため，受話者推定技術が必要となる。現状のスマートフォン上のアプリやスマートスピーカよりも，より自然な対話を目指したり，システムを実環境に設置したりする場合には，これらの技術の重要性はより高くなる。

（１） 円滑な話者交替

対話では，基本的に参加者は交互に話すことが想定される。すなわちある参加者が話し，それが終了した後に別の参加者が話し始める。これは話者交替，つまりターンテイキング（turn taking）と呼ばれる。単信式のトランシーバ（同時に送信か受信のどちらかし

かできない）での対話では，発話の終わりに「どうぞ」と言うなどして発話の終了を明示することで，ターンテイキングが実現される。

しかしながら，人間どうしの自然な音声対話では，参加者は必ずしも交互に話してはいない。つまり，同時に話し始めたり，話している最中に他の参加者が話し始めたりといった現象が頻繁に生じる。

発話は，音声対話を構成する基本的な単位であるが，常に明確に規定できるわけではない。単純には「400ミリ秒以上の無音区間で区切られた音声区間」という単位が考えられるが，例えば促音（「っ」で表記）は物理的には無音であり，この長さが400ミリ秒を超えることもよくある。したがって，単純に一定長の無音区間で区切っても必ずしも適切な単位とはならない。

ここで発話が過剰に細かく区切られてしまう例を示す。例えば，新幹線の予約窓口で，「帰り…は自由席でお願いします」という発話があったとしよう。話し手は考えながら話していて言い淀んでおり，この「…」は無音区間を表す。この時，単純に無音区間を用いて発話を認定すると，「帰り」と「は自由席でお願いします」という2つの発話の断片が得られるが，これらは「帰りは自由席でお願いします」という一つの発話として理解するのが妥当であろう。

一方で，無音区間がなくても内容から2発話に区切るべき場合もある。例えば，「新横浜からどこまで行きますか？」という質問に対して，「違います新大阪です」と無音区間なしに話したとしよう。これは「違います」という否定と，「新大阪（から）です」という出発地を示す2つの発話がなされたとして，「新横浜ではなく新大阪からどこかに行く」と言っていると理解するのが妥当である。しかしこれを一発話と捉えてキーワードだけを拾うと，「（新横浜から）新大阪まで」のように誤って理解してしまうことになる。

一発話は，(1) 音響信号，(2) 対話行為（発話の意味内容），(3) ターン，の3つの側面から規定できる。(1) はその区間内の音響信号が，物理的にある一定以上の振幅を持っていること，(2) はその内容がひとまとまりの意味を持っていること，(3) は対話相手の話している区間に挟まれた部分であること，である。現在の音声対話システムでは，この3つの条件を満たした発話が入力されることを仮定している。この3つが一致するのは，相手の発話が完全に終わるまで話し始めず，一発話には一つの対話行為のみを含めて，自分の発話を必ず一息で(無音区間なしに)話す場合である。

しかしながら，人間どうしの対話を考えると，このような発話しか許容しないのは不自然である。またシステムがターンの処理を誤ることも現実にはしばしば起こる。つまり，ユーザが話している最中であるにも関わらずシステムが誤って話し始める場合や，ユーザが話し終わったのにシステムが応答しない場合である。したがって，上述の3つの条件が満たされない場合でも，システムが適切に動作できることが望ましい。

音声対話システムでは，システム応答の大きな遅延や話者間の発話の重なりがない状態で対話が進むことが望ましい。このためには，システムはユーザが話し終わればすぐにそれを検出し，遅滞なく発話を開始する必要がある。この一方で，ユーザが発話を終了しないうちにシステムが話し始めると発話が重なってしまう。したがって，ユーザが話し終わったかどうかを正確に検出する必要がある。これは発話の終端検出（endpoint detection）と呼ばれる。素早く終端を検出しようとした場合には誤検出が増えるというトレードオフの関係にあるため，発話の終端検出の性能を高める取り組みとともに，誤検出を事後的に修復するというアプローチもある[9]。

このような話者交替は，現状の一問一答のやりとりではまだ重要視されていないが，より自然な音声対話を実現する際には重要となる。スマートフォン上のアプリでは起動ボタンにより発話の始端の候補区間を限定したり，スマートスピーカでは発話の冒頭にマジックワード（例：「アレクサ，」「オーケーグーグル，」など）を言わせたりすることで，発話区間検出の一部（発話の始端検出）を簡単化している。しかし今後，人型ロボットなどボタンを持たないインタフェースでの音声対話や，マジックワードなしでの音声対話を実現するには，本節で述べたような技術が重要となる。また，特に一般のユーザが内容を考えながら話す場合，発話に言い淀みが生じるのは不可避であるため，これを許容したシステム設計が望ましい。さらに音声対話システムをユーザインタフェースとして見た場合，ユーザの入力に対して素早く反応するのは必要条件と考えられるため，終端検出の高度化やその誤りの修復機構も，今後重要性が増すだろう。

（2） 音の到来方向の推定

スマートフォン上のエージェントでは，ユーザはほぼ必ずシステムに正対して話す。これに対して，実空間内に存在するシステムでは，ユーザとの位置関係が固定ではないため，話し手の存在位置を知るには音源定位と呼ばれる音響信号処理が必要となる。またロボットなど，システムが擬人化された顔や目を持つ場合は，人はしばしばシステムにも人間と同じようにふるまうことを期待するため，ロボットが話し手の方向を向くことは必須の機能である。

このような音響信号処理は，複数のマイク（マイクロホンアレー）を使って行われる。複数のマイクに入力される音響信号の到達時間差や位相情報を使うため，一般にマイクロホンアレーはマイク素子間の距離が大きい方が高性能である。これに対して，比較的小型のロボット上でこれを実現するには，従来の自由空間中でのマイクロホンアレーを使った音源定位と比べて，1. マイク間の距離が小さい，2. マイクの近くにロボットを駆動させるモータ用のファンなど雑音音源がある，3. 頭部の外装や肩など音響伝達特性に影響を与える障害物が近くにある，4. 各マイクの周波数特性も必ずしも同じではない，といった状況から問題は難化する。この問題に対して，我々は deep neural network（DNN）を使った音源定位にも取り組んでいる[10]。

また，認識すべき音声に他の音が混入している場合には，音声認識を行う前に音源分離を行う必要もある。この音源分離の際にも，どの方向から来た音かという情報を活用すべく，音源定位結果が用いられることも多い。そもそも，スマートフォン上のアプリのようにユーザの口元にあるマイク（接話型マイク）を使う場合と，ロボット自身に備え付けられたマイクを使う場合では，音声認識の難しさが大きく異なる。音源である口の位置と，それを受けるマイクの位置との距離が大きくなることで，SN 比（信号と雑音の比）が下がり，さらには周辺雑音や残響の影響を受けるためである。音声認識の前処理としても，これらの音響信号処理は重要である。

（3） 複数人対話における課題

ロボットなど物理的実体を持ったシステムは，複数の参加者がいる場で対話を行うことも多い。一般に人間どうしの対話では，対話の参加者は，他の参加者に適宜視線を配ったり，発話を促したり，全員が向かい合うように位置したりと，複数人での対話の場が構成されていることをお互いに認識できるよう行動している。このような現象のモデルとして，複数人対話における参加構造が知られている[11]。これを図 6 に示す。このモデルでは，まず，その時点での話者（話し手）と受話者（聞き手）が存在している。さらに，対話に参加しているがその時点では話し手でも聞き手でもない傍参加者，その場にいることは他の参加者に認識されているが対話には参加していない傍観者，その場にいることも認識されていないが対話の内容を聞いている盗聴者などが考えられている。またこの参加構造における役割は動的に変化する。例えば，傍参加者が新たに対話に加わったり，参加者が対話を離脱したりする。

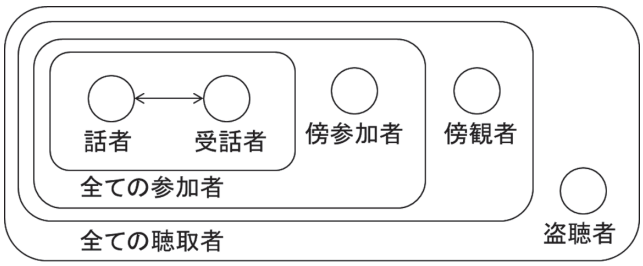

図 6：複数人対話の参加構造[11]

このような場においてシステムに適切に行動させるための技術のひとつとして，受話者推定がある。複数人対話では，システムが観測した全ての入力に対して必ずしもシステムが応答する必要はないため，その入力発話が誰に向けたものであるかの推定が必要となる。この結果をシステムの次の行動に結びつけ，応答義務推定[12]とも呼ばれる。入力された音声信号が周辺の雑音である場合や，自分以外に話しかけられた音声（例えば独り言）である場合には，システムは応答すべきではない。つまり，ユーザがシステムに向けて話した場合にのみシステムは応答すべきであり，それ以外の場合にシステムが勝手に話し始めると煩わしい。したがってシステムが観測した発話の受け手を推定し，それに応じて行動する必要がある。

このような技術は，複数人が参加する対話を理解するうえでも重要となる。どのユーザがどの程度対話に参加しているのかを表す概念として，エンゲージメント（engagement）がある。この推定は，対話システムが応答をするか否かの決定やその内容の決定において重要な課題である。例えば，対話に参加していない人に話しかけてしまうのを防いだり，対話ロボットが自身の視線や体の向きを適切に制御したりすることが

可能になる。

さらに人間は，対話において話されている内容だけでなく，参加者間の社会的な関係も認識できる。例えば，その対話の内容に対する積極性などの態度はもとより，参加者間の親密度や上下関係もある程度推測し，それに応じて行動を変えることができる。このような社会的関係を認識するための技術やそれを備えたロボットも，今後必要になってくるだろう。

6．今後の展望

賢く対話できるシステムを作ることは，人間がどのように話しているのかを知るための科学的課題であると同時に，いかに社会に役立つシステムを作るかという工学的課題でもある。音声という，人間が日常的に使うのと同じコミュニケーションメディアを用いていることから，これが完成した暁には多くの人にとって使いやすいインタフェースとなる可能性を秘めている。この一方で，システムに人間と同等の能力を求め，人間らしさを感じようとするほど，不自然な部分が目に付くという「不気味の谷」問題もはらんでいる。つまり，人間のように賢く対話できるシステムというのは，見果てぬ夢なのかもしれない。

工学的なシステム構築という観点から言うと，ユーザの挙動は常に問題となる。つまり，研究室環境で統制されたユーザとは異なり，実際のユーザはシステム開発者の想定を超える多様な挙動をする。ユーザテストを行うことで着実にシステムは改善されるものの，本質的にシステムにとっての想定外が起こり得ることを前提にしたシステム設計が必要である。

ここで，対話は共同行為，つまり，相手と互いに調整し合いながら行う行為である[11]。すなわち，ユーザとシステムは，相手とは独立に行動するのではなく，相手に応じたふるまいをする。つまり，対話である以上，システムが受け身にあらゆるユーザからの入力を処理できるようにするという枠組みではなく，ユーザを含んだ系として対話システムを考える必要があると考えている[13]。つまり，システムがうまく人間の助けを得られるような対話システムの設計論も今後必要となろう。

この一環として，対話を通じて知識を獲得する対話システムの研究を現在進めている[14]。対話システムにおいて，システムの知識にない単語（未知語）の存在は避けられない。つまり，どんなにシステムの知識を大規模化したとしても，そこにはない語彙が必ず現れる。特に，家庭用の対話システムが普及した場合を考えると，ローカルな地名や知り合いの人名など，一般的ではない語彙が対話において重要な役割を占めると考えられる。そのような語彙を，各家庭向けに人手でカスタマイズするのはコスト的に現実的ではない。このことからも，対話を通じて対象ドメインの知識を獲得する技術，ひいてはユーザの助けをうまく得ながら対話を通じてシステムを賢くする技術も，今後必要性が増すと考えられる。

音声対話システムを含む人工知能技術は第３次ブームを迎えており，その可能性ともに，「人工知能が人間を支配する」などといった懸念も喧伝されている。しかし冷静に考えると，人間の能力を超えた機械は既に多数ある。例えば，車は人間よりもはるかに速く走ることができ，ブルドーザは人間よりも重い物を持ち上げることができる。知的作業という面でも，電卓は昔から人間よりも速く正確に計算ができる。人工知能がこれら既存の事実と比べて，何が同じで何が違うと我々が感じているのか，考えてみる必要があるだろう。

この一方で，車は濫用すると人の命を奪うこともあり得る。このため，車が社会で人間と共存するために交通法規が定められた。ただし不安に煽られて過剰な規制をした場合，多国間競争下にある研究開発の抑制に繋がる。人工知能技術の中身をある程度知ったうえで，それが社会でうまく使われるための枠組み作りは必要となるだろう。その際には，技術者と市民との間で，互いを理解したうえでの建設的な議論が必要不可欠である。

（参考文献）

(1) 中野 幹生，駒谷 和範，船越 孝太郎，中野 有紀子，“対話システム”，コロナ社，2015。

(2) 河原 達也，“音声認識技術の展開，”信学技報 PRMU2015-111, pp.111-116, 2015。

(3) Toshiyuki Sakai and Shuji Doshita, “An Automatic Recognition System of Speech Sounds,” Studia phonologica, Vol.2, 1962, pp.83-95. http://hdl.handle.net/2433/52632

(4)「特集 人に近づく音声インタフェース 第１部＜市場動向編＞ クラウドの利用で飛躍 用途はスマホ以外にも拡大」日経エレクトロニクス 2012 年 12 月 24 日号，2014. http://techon.nikkeibp.co.jp/article/HONSHI/20121221/257654/

(5) 河原達也，“音声対話システムの進化と淘汰 －歴史と最近の技術動向－，”人工知能学会誌，Vol.28, No.1, pp.45-51, 2013。

(6) 高村 大也：“言語処理のための機械学習入門”，コロナ社，

2010。
(7) 東中 竜一郎，“最近の対話システム事情 ～深層学習・データセット・コンペティションの観点から～”，映像メディア学会誌，Vol.73, No.2, pp.271-276, 2019。
(8) 駒谷 和範，“円滑な対話進行のための音声からの情報抽出”（小特集：音声言語理解のこれまでとこれから），電子情報通信学会誌，Vol.101, No.9, pp.908-913, 2018。
(9) Kazunori Komatani, Naoki Hotta, Satoshi Sato, Mikio Nakano, “User-Adaptive A Posteriori Restoration for Incorrectly Segmented Utterances in Spoken Dialogue Systems,” Dialogue and Discourse, Vol.8, No.2, pp.206-224, 2017.
(10) Ryu Takeda, Kazunori Komatani, “Unsupervised Adaptation of Deep Neural Networks for Sound Source Localization using Entropy Minimization,” IEEE International Conference on Acoustics, Speech and Signal Processing（ICASSP）, pp.2217-2221, 2017.
(11) Herbert Clark, “Using Language,” Cambridge University Press, 1996.
(12) Takaaki Sugiyama, Kotaro Funakoshi, Mikio Nakano, Kazunori Komatani, “Estimating Response Obligation in Multi-Party Human-Robot Dialogues,” IEEE-RAS International Conference on Humanoid Robots（Humanoids 2015）, pp.166-172, 2015.
(13) 駒谷 和範，“階層的発話行動理解に基づく音声インタラクション，” 計測と制御 , Vol.55, No.10, pp.878-883, 2016.
(14) Kohei Ono, Ryu Takeda, Eric Nichols, Mikio Nakano, Kazunori Komatani, “Lexical Acquisition through Implicit Confirmations over Multiple Dialoguesm,” 18th Annual SIGDIAL Meeting on Discourse and Dialogue, pp.50-59, 2017.

（原稿受領 2019.4.24）