



しゃべってコンシェルにおける質問応答技術

NTTメディアインテリジェンス研究所¹ / NTTドコモ²

ひがしなか りゅういちろう¹ きだみつ くがつか¹ うちだ わたる² よしむら たけし²
東中 竜一郎 / 貞光 九月 / 内田 渉 / 吉村 健

「しゃべってコンシェル」はユーザのさまざまな質問に答えることができますが、すべての質問に対する答えを前もって準備しているわけではありません。ユーザが質問するたびに、言語処理を使ってその内容を解析し、データベースやインターネットを検索して回答を探しています。ここでは、しゃべってコンシェルがどのように言語処理を駆使し、質問に答えているかを解説します。

しゃべってコンシェル

「しゃべってコンシェル^{*}」は、NTTドコモが提供する音声エージェントサービスです。スマートフォンに音声で話しかけるだけで端末機能の呼び出しや各種アプリケーションの実行などを行うことができます⁽¹⁾。本サービスは、2012年3月のリリース以降、2012年10月までに400万ダウンロード、1億8000万回のアクセスを達成するなど、人気のサービスとなっています。

しゃべってコンシェルは、2012年6月にバージョンアップされ、その際にQ&A機能、すなわちユーザの質問に回答する機能が追加されました。この機能によって、例えば図1のように、ユーザのさまざまな質問に答えることができるようになりました。ここでは、このQ&A機能を支える質問応答技術について解説します。なお、Q&A機能は、NTTメディアインテリジェンス研究所の技術を基に、NTTドコモが開発したものです。

質問応答技術

質問応答技術とは、ユーザの質問にズバリ回答を返す技術のことです⁽²⁾。例えば、「世界で一番高い山は？」という質問に「エベレスト」と回答するような技術です。しゃべってコンシェルはこの技術を用いて質問に答えています。

ユーザの質問は多種多様なため、あらかじめすべての質問を予想し、その答えを準備しておくことは困難です。そこで質問応答技術では、人間が質問から回答を導く手順をアルゴリズムとして実装し、多様な質問に答えられるようにしています。

人間は質問に答えるとき、まず、自分の知識で答えられるものであれば、それを用いて答え、もし答えを知らなければ、図書館に行ったり、インターネットを検索したりして、その答えを探します。質問応答技術も同様です。

しゃべってコンシェルは、自分の知識として、知識データベースを保持しています。そして、ユーザ質問に対する答えが知識データベースにあれば、それを用いて回答し、見つからなければ、インターネットを検索し回答を探します。

知識データベースには、ユーザが頻繁に尋ねるような事柄についてのデータを格納しておきます。そうすることで、よく聞かれる定番質問には、即座に正確な回答を返すことができます。知識デー

*「しゃべってコンシェル」は株式会社NTTドコモの登録商標です。



図1 しゃべってコンシェルが質問に回答している画面例

データベースでカバーできない事柄については、インターネット検索を用いることで、必ずしも正確な回答ではないかもしれませんが、ユーザの質問を無視することなく、何らかの回答を返します。何らかの回答を返すことは、ユーザ満足度の観点からとても重要です。

知識データベースを用いて回答する手法をデータベース型質問応答と呼び、インターネット検索を用いて回答する手法を検索型質問応答と呼びます。

データベース型質問応答

データベース型質問応答が扱うものとして、私たちは、ある対象についての属性を尋ねる質問に着目しました。このような質問は定番質問として多く見られるタイプで、例えば、「エベレストの高さは?」「日本の首相は?」「オバマ大統領の誕生日は?」などが挙げられます。「エベレストの高さは?」であれば、対象が「エベレスト」であり、属性は「標

高」となります。

データベース型質問応答では、まずユーザの質問を解析し、質問から対象と属性を抽出します。そして、知識データベースから、対象の属性値を検索します。エベレストの例の場合、知識データベース中の「標高」テーブルから、「エベレスト」の値を探すことになります。データベース型質問応答の構成を図2に示します。

一連の処理においてもっとも困難なポイントは、質問からの対象と属性の抽出です。なぜなら、対象の属性を尋ねる質問は、必ずしも「対象の属性は?」という単純な表現だけでなされるわけではなく、非常に多様だからです。エベレストの例でいうと、「エベレストって何メートル?」「エベレストはどのくらい高い?」「高さ教えて、エベレストの」など、いろいろな言い回しが考えられ、これらの表現のどれからも正しく「エベレスト」と「標高」を抽出することは容易ではあ

りません。

そこで私たちは、機械学習を用いることでこの問題を解決しました。機械学習とは、事例から統計処理によって、計算機に物事の判断基準を学習させる枠組みのことです。

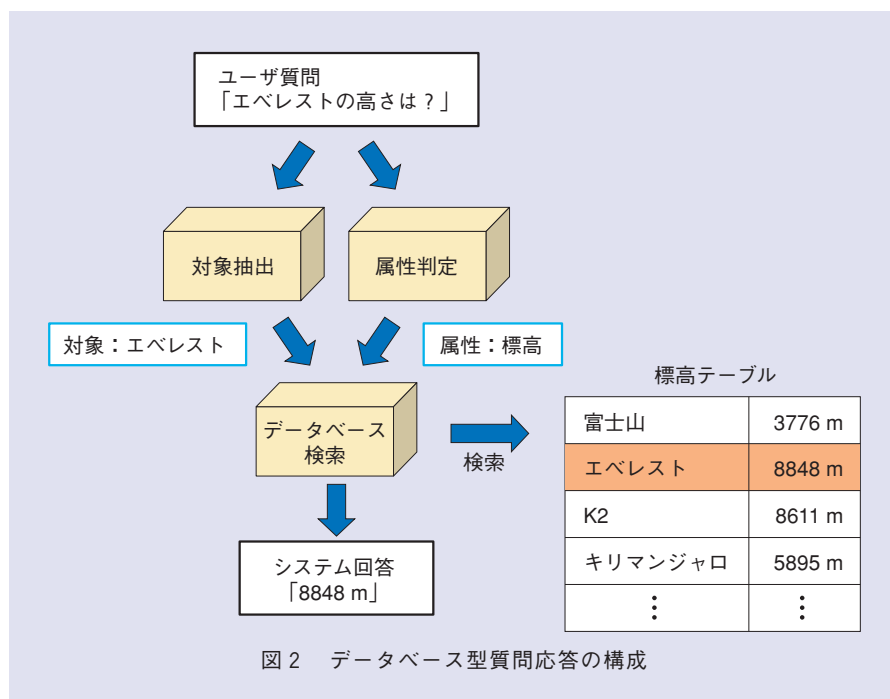
具体的には、対象の個所をラベル付けした質問文を大量に準備し、質問文中の単語系列がどのような場合に対象となるかという基準を学習しました。また、属性があらかじめ分かっている質問文を大量に準備し、質問中の単語列の情報から、特定の属性についてであると判断する基準を学習しました。結果として、いろいろな言い回しであっても、質問から高精度な対象と属性の抽出を実現することができました。

検索型質問応答

データベース型質問応答で回答が見つからない場合は検索型質問応答が実行されます。検索型質問応答では、質問解析、文書検索、回答抽出、回答評価の4つのステップによって、インターネットから、質問の回答を探します。図3は検索型質問応答の構成です。

(1) 質問解析

質問解析では、質問の「回答タイプ」を判定します。具体的には、まず単語で回答する「単語回答型」か、理由、方法、定義、評判など、文章で回答する「文章回答型」かを判定します。学校のテストでいえば、設問が穴埋め問題か、記述式かを判定しているのと似ています。そして単語回答型であれば、さらに単語がどの固有表現タイプであれば良いかを判定します。固有表現とは、固有名詞、時間、数値を指す表現の総称で、固有表現タイプとは、固有表現を類型化したとき、そのどれにあてはまるのかを示すものです。固有表現タイプの判定は、



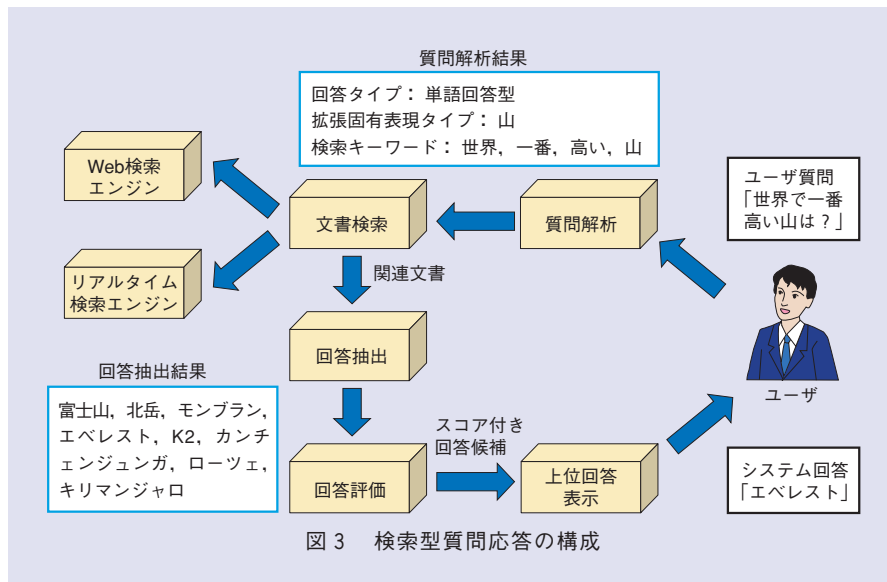


図3 検索型質問応答の構成

リアルタイム判定は、直近を示す「今日」「現在」「昨日」といった単語が質問に含まれているかだけでなく、日々指す対象が変わる単語（例えば、「先発」や「ゲスト」）が含まれているか、また、インターネット上で急に出現頻度が増えたような話題語が含まれているかなどに基づいて行います。この機構により、「今日のメダリストは？」「〇〇番組のゲストは？」「ノーベル賞を取ったのは誰？」（ノーベル賞が話題語だとします）といった質問について、最新の情報を基にした回答を提示することができます。

(3) 回答抽出

回答抽出では、検索された文書から回答タイプに合致した回答候補を抽出します。ここでは、回答タイプが単語回答型の場合について説明します。

単語回答型では、拡張固有表現タイプに合致する単語を回答候補として文書から抽出します。例えば、「世界で一番高い山は？」の場合、文書から「山の名前」をすべて抽出します。

私たちは、拡張固有表現タイプのそれぞれについて、文書から回答候補を抽出する抽出器を、機械学習の手法で構築しました。具体的には、拡張固有表現の個所をラベル付けした文章を大量に準備し、単語系列が特定の拡張固有表現タイプであると判断する基準を学習しました。今回、100を超える拡張固有表現タイプを扱うことから、100を超える抽出器を構築しました。「山の名前」の抽出器もその1つです。なお、ここで学習に用いた拡張固有表現のデータは、日本語では最大規模のものです。

学習された抽出器は高精度ですが、学習データに含まれない状況では何も抽出できないという問題が残ります。そこで別途、拡張固有表現の辞書を整備し、抽出器が見逃してしまいそうな回答候補

答えるべき単語がどんなものであれば良いかを絞り込む作業にあたります。

質問応答では、固有表現タイプとして8つのタイプ（人名、地名、組織名、固有物名、日付、時間、金額、割合）を用いることが一般的です。しかし、8タイプでは粗すぎて、答えるべき単語をうまく絞り込めない可能性があります。

そこで私たちは、拡張固有表現⁽³⁾と呼ばれるものを参考にし、100を超える拡張固有表現タイプを用いて、単語のタイプを絞り込むことにしました。例えば、「世界で一番高い山は？」では「山」、「黒澤明の代表作は？」であれば「映画」という拡張固有表現タイプであると判定し、回答すべき単語が山の名前や映画の名前であるというところまで絞り込みます。

単語回答型・文章回答型の判定や、拡張固有表現タイプの判定を高精度に行うことは、質問の表現が多様なことから、簡単ではありません。そこで私たちは、データベース型質問応答のときと同様、大量の事例を準備し、機械学習の手法を用いることで、高精度な判定を実

現しました。この学習には、質問中の単語が持つ意味的な情報を利用することが不可欠です。そのために、NTT研究所の成果である意味情報が付与された大規模な辞書⁽⁴⁾を最大限活用しています。

なお、質問解析では回答タイプの判定に加え、次段の文書検索で用いる検索キーワードも抽出します。例えば、「世界で一番高い山は？」であれば、形態素解析の結果から、「世界」「一番」「高い」「山」といった単語を検索キーワードとして抽出します。

(2) 文書検索

文書検索では、検索キーワードを用いてインターネットを検索し、関連する文書を取得します。検索には、基本的に一般的なWeb検索エンジンを用いますが、今まさに起こっているような最新の情報については、Web検索エンジンでは見つからないことがあります。そこで、リアルタイム判定という機構を導入し、最新の情報についての質問であると判断される場合には、検索先をリアルタイム検索（Twitterのツイートの検索）に切り替えます。



図4 プログラム型質問応答で「ロンドンの今の時刻」に答えた場合の画面例

も発見できるようにしました。この辞書には100万を超える単語がその拡張固有表現タイプとともに載っており、山の名前だけでも数千あります。抽出器とこの辞書を併用することで、文書に含まれる拡張固有表現の網羅的な抽出を実現することができました。

(4) 回答評価

回答評価は、回答候補に正解らしさの点数を与え、順序付けを行います。しゃべってコンシェルでは1位の回答候補が読み上げられるため、できるだけ1位に正解を持ってこることが重要です。

質問応答では、回答候補の周りに多くの検索キーワードが見つかる場合、正解になりやすいという経験則があり、多くの質問応答システムはこれに則って点数付けを行います。しかし、例外も多く、精度が伸び悩むという問題がありました。

そこで私たちはここでも機械学習を用い、高精度なランキングを実現しました。具体的には、周辺文脈がどのような状況において、ある回答候補が正解になるかという基準を、大量の事例から学習しま

した。回答候補には、この基準に照らし合わせた、正解らしさを表すスコアが付与されます。最終的に、回答候補はスコアでランキングされ、上位の候補が画面上に表示されます。

今後の展開

ここでは、2012年6月にリリースされた、しゃべってコンシェルにおけるQ&A機能について解説しました。言語処理、特に機械学習をフル活用して、人間が質問に答える過程を真似て答えている様子が分かっていただけではないかと思えます。

今回はデータベース型質問応答と検索型質問応答について説明しましたが、最近の取り組みとして、プログラム型質問応答も追加しました。これは質問に答えるために、何らかのロジックを駆動するタイプの質問に対応するためのものです。例えば、「ロンドンは今何時？」の答えは、データベースを検索しても、インターネットを検索しても見つかりません。世界時計の情報を参照し、その時刻情報を抽出するロジックを呼び出す必要があります。図4はプログラム型で質問に答えている画面の例です。プログラム型質問応答は、世界時計のほか、カレンダー、電卓、単位変換などにも対応しています。

質問応答の回答精度はまだまだ改善の余地があると考えていますし、今後も改善していきたいと思っています。また、文章回答型の質問についてはすべてに答えられるわけではありません。例えば、理由を聞く質問には答えられません。今後は、回答できる質問の幅も拡充していきたいと考えています。

■参考文献

- (1) 吉村：“しゃべってコンシェルと言語処理,” 情処学研報, 2012-SLP-93, pp.1-6, 2012.

- (2) 磯崎・東中・永田・加藤・奥村（監修）：“質問応答システム,” コロナ社, 2009.
 (3) S. Sekine, K. Sudo, and C. Nobata: “Extended named entity hierarchy,” LREC2002, Canary Islands, Spain, May 2002.
 (4) 池原・宮崎・白井・横尾・中岩・小倉・大山・林：“日本語語彙大系,” 岩波書店, 1997.



(左から) 吉村 健/ 内田 渉/
東中 竜一郎/ 貞光 九月

手元の端末に質問すれば答えが返ってくる。そんなSFのような世界に一歩近づくことができました。今後も未来志向のサービスを、タイムリに実現していきたいと思っておりますので、どうぞご期待ください。

◆問い合わせ先

NTTメディアインテリジェンス研究所
音声言語メディアプロジェクト
TEL 046-859-2027
FAX 046-855-1054
E-mail higashinaka.ryuichiro@lab.ntt.co.jp