

# チャットボットの会話テクニック

## Chatbot Conversation Techniques

● 倉知 陽一      ● 川上 真一      ● 宮澤 孝彰

---

### あらまし

AI(人工知能)などのデジタルテクノロジーを活用することで、顧客とのエンゲージメント(絆)を強化する「感情的価値」の提供、つまり、ポジティブなサービス利用体験(カスタマーエクスペリエンス)による顧客の心の満足の実現が求められている。そこで注目されているのがチャットボットである。まるで人のように会話し、「そこに知能がある」と人に錯覚させることができるものがチャットボットの本質である。

本稿では、「どのように実装すれば、うまく人に錯覚してもらえるようになるのか」という観点で、チャットボット実装に活用できる会話のテクニックを紹介する。

### Abstract

Today, businesses are required to wield digital technology to offer their customers emotional value that boosts their engagement, positive service experiences (customer experiences) that deliver a sense of satisfaction. In this context, chatbots are garnering attention. Their intrinsic value is in convincing people of their intelligence by simulating human manners while acting as conversational partners. This paper presents the techniques of conversation that can be implemented in chatbot, from viewpoint of “how to give people an illusion.”

---

### まえがき

AI（人工知能）などのデジタルテクノロジーを活用することで、顧客とのエンゲージメント（絆）を強化する感情的価値の提供が求められている。これはすなわち、ポジティブなサービス利用を体験（カスタマーエクスペリエンス）することによる顧客の心の満足の実現である。

そこで注目されているのがチャットボットである。チャットボットとAIが同列、あるいは関係深いものとして語られることは多い。しかし、チャットボットはAIではないし、必ずしもAI技術を実装する必要もない。もちろん、Deep LearningのようなAI技術を用いてチャットボットに実装することもできる。AI技術の有無に関わらず、まるで人のように会話し、「そこに知能がある」と人に錯覚させることができるものがチャットボットの本質である。

本稿では、富士通のチャットボット（Customer Engagement Solution CHORDSHIP）<sup>(1)</sup> 開発の経験を踏まえ、どのように実装すれば、うまく錯覚してもらえるようになるのかという観点で、チャットボット実装に活用できる会話のテクニックを紹介する。

### チャットボットの基本的な仕組み

チャットボットは、とてもシンプルな仕組みで作ることができる。ユーザーからのメッセージを「受信」し、受信したメッセージを「理解」して「発話」すべきメッセージを組み立て、組み立てたメッセージを「送信」する。基本的にはこれだけである。この四つのステップを繰り返すプログラムを実行すれば、それがチャットボットになる。この中で、「受信」と「送信」については、一般的な通信アプリケーションと実装方法はさほど変わらないため本稿では割愛し、次章以降、「理解」と「発話」の実装テクニックについて述べる。

### 理解のテクニック

会話の理解において最も重要なのは、話者の意図あるいは会話の主題の理解である。この部分を誤解してしまうと、その後の会話は全く噛み合わない。例えば、別れの挨拶を意図して発せら

れた「じゃあ、また今度」という言葉を次に会う日を約束したいと誤解して「今度っていつですか？」と返してしまうといったケースである。

そもそも、人間であっても話者の意図を正確につかむことは難しい。全く同じ言葉が発せられたとしても、話者の性格や置かれた立場・状況などによって、意図は全く異なる可能性があるからである。

しかし、コンタクトセンターへの問い合わせといった特定の場面で交わされる会話を前提とすれば、話は変わってくる。会話の場面を限定することで話者の意図を比較的簡単に推測でき、会話を正しく誘導することも可能になる。

コンタクトセンターへの問い合わせでは、主に2種類の話者の意図が存在すると想定できる。一つは、商品の詳細を知りたい、あるいは購入した商品の使い方を教えてほしいといった、商品に関する質問である。もう一つは、契約したい、あるいは解約したいといった、手続きに関するものである。この前提は、チャットボットにも適用できる。このケースでチャットボットの提供者が最初にやるべきことは、どちらの会話意図に対応するかを決めることである。それによって、後はその方向に会話を誘導しさえすれば、会話が噛み合わない、という根本的な問題は避けられる。

### 誘導のテクニック

会話の誘導に関しては、好みの開発言語を用いてコーディングしてもよいが、フローチャート形式のグラフィカルユーザーインターフェース（GUI）で会話を組み立てられるソフトウェアを活用すれば、より短時間で構築できる。こういった機能を実装した製品はいくつかあるが、富士通が開発しているCHORDSHIPも会話をフローチャートとして描ける製品である（図-1）。

### 手続き系業務に対応するためのテクニック

手続き系業務に対応するためのテクニックとしては、スロットフィリングと呼ばれる仕組みがよく利用される。スロットフィリングとは、チャットボットがユーザーと会話しながら、内部的なエントリフォームを埋めていき、全ての入力項目が埋まれば業務手続きを実行する、という仕組みで

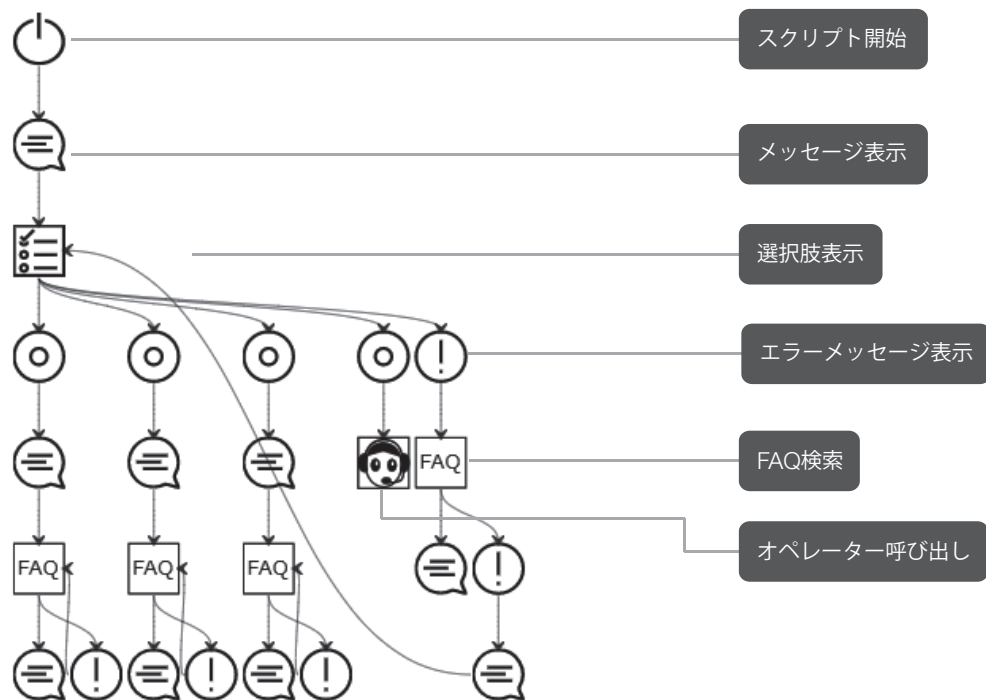


図-1 CHORDSHIPのGUIによる会話シナリオ構築の例

ある。

### 質問対応系業務に関するテクニック

質問対応系業務に関するテクニックの基本は、文書検索である。ユーザーからの質問文をキーにして、該当する答えを検索して返す。これだけで会話は成り立つ。この際に用いる質問と答えのセットは、チャットボット用に新しく作る必要はなく、既存のFAQをそのまま転用できる。ただし、検索エンジンの使い方に関しては、少し工夫が必要になる。それは、FAQ用の検索エンジンをそのままチャットボットに転用しても、期待どおりの答えが得られないことが多いためである。その原因は、チャットで質問する際と、FAQサイトで検索する際とでは、ユーザーの行動が異なる、というところにある。

検索文字列の入力を例にとると、FAQ検索においては、一つの単語、あるいは複数の単語を空白で区切ったものを入力するのが一般的である。一方、チャットでユーザーが入力するのは単語ではなく文章である。検索対象の単語のみが入力されれば正しくヒットする検索エンジンも、検索対象とは関係のない不必要な情報を含んだ文章の場合

では、うまくヒットしないケースが多い。

これを避けるためには、検索の前にユーザーが入力した文章のクレンジング処理を行うことである。例えば、不必要な情報を削除するといった処理である。更には、ユーザーの文章を単語に分割して抜き出し、その単語をよりヒットしやすい単語に置き換えて検索するといった処理も行う。

FAQ検索とチャットボットのもう一つの大きな違いは、ユーザーインターフェースである。FAQ検索の場合、通常は多数の候補を検索結果として表示してユーザーに選択させる。一方、チャットボットでは、ユーザーインターフェースの制約上、多数の選択肢を提示できない(図-2)。更に、チャットボットとユーザーとの会話に人間らしさをうまく演出したいという点にこだわる場合は、選択肢の提示ではなく、一つの答えだけを返すことが重要である。

このように、チャットボット特有のユーザーインターフェースを考慮すると、検索のヒット率の向上は至上命題となる。そして、賢いチャットボットを実現するためには、このユーザーインターフェースの品質を向上させる必要がある。



FAQ検索画面

チャット画面

図-2 FAQ検索画面とチャット画面の違い

## ヒット率向上のための検索手法

### ● 様々な検索手法

文章を検索する手法には様々なものがある。例えば、SQL (Structured Query Language) の演算子の一つであるLIKEを使った部分一致検索や、文章同士の距離計算、テキスト分類といった手法である。

SQLのLIKEを使った部分一致検索も、工夫次第で文章検索に利用できる。例えば、ユーザーが入力した文章の中から単語を取り出し、その単語で部分一致検索を行い、検索対象の文章に存在すれば1ポイント加算する。この処理を単語ごとに繰り返し、最終的に最も多くの単語にヒットした文章を検索結果として提示する方法である。しかし、このポイント制による検索では、同一ポイントの結果が多数存在する可能性が高く、一つの答えだけを返すことは難しい。

文章同士の距離計算とは、二つの文章の類似度を数値化して比較する手法を指す。チャットボットでよく使われる手法として、レーベンシュタイン距離、ジャロ・ウィンクラー距離、ゲシュタルトパターンマッチング<sup>(2)</sup>が挙げられる。例えば「こんにちは」という文章に対して、「こんにちは」と「こ

んばんは」のどちらがより距離的に近いかを求める場合は次のようになる。まず、「こんにちは」と「こんにちは」の距離であるが、「こんにちは」に「に」を1文字挿入すれば「こんにちは」となるため、距離は1と計算される。これに対して、「こんにちは」を「こんばんは」にするためには、「こんにちは」に対して、「ち」の削除、「ばん」の追加で合計3の距離となる。これにより、「こんにちは」に対しては、「こんばんは」よりも「こんにちは」の方が近いと判断できる。

テキスト分類の代表的な手法としては、確率モデルによる分類 (例：ナイーブベイズ)、統計情報による分類 (例：TF-IDF)、ニューラルネットワークによる分類 (例：Doc2Vec)<sup>(3)</sup>が挙げられる。例えば、統計情報による分類の一つであるTF-IDFを用いると、FAQの文章中にある特徴的なトークンを捉え、ユーザー入力文中にあるトークンとの一致から文章の類似度を算出できる。トークンは、主に二つに分類できる。

### ● トークンを「文字」にするか「単語」にするか

コンピュータが文章構成要素を認識する単位であるトークンは、文字ベースのもの (一定の長さを持つ文字列) と単語ベースのもの (特定の意味を表す一塊の文字列) の二つのタイプに分けられ

る。例えば、ユーザーが入力した文章が「パソコン」であったとすると、文字トークンの場合、「パソ」「ソコ」「コン」に分割できる（2文字ずつ分割するバイグラムの場合）。一方、単語トークンとして扱うならば「パソコン」という一つのトークンになる。

採用するトークンタイプの違いによって検索のヒット率も変わってくるため、検索対象とするデータセットの性格や、単語切り出し処理（トークナイザ）の性能により、使い分けが必要になる。例えば、一般的な企業のFAQをデータセットとする検索の場合、単語切り出し処理が正確に働く可能性が高いため、単語トークンの選択が有利に働く場合が多い。一方、雑談対話を主目的とするコミュニケーション用チャットボットの場合、検索対象のデータセットは話し言葉が中心となり、単語切り出し処理が正確に働かない可能性が高くなる。このような場合には、文字トークン（バイグラム）を選択する方が有利になる。

### ● 複数の検索手法の組み合わせ

ヒット率を向上させるためには、複数の検索手法を組み合わせることも有効な手段である。例えば、二つの検索手法「単語トークン+ナイーブベイズ」と「文字トークン+レーベンシュタイン距離」の組み合わせを考えてみる。想定質問文として「CHORDSHIP って何？」が登録され、Rが抜けた「CHODSHIP って何？」という検索文が入力された場合を想定する。単語トークン+ナイーブベイズでは、「CHODSHIP」と「CHORDSHIP」を全くの別物として判断するため、想定質問文にはヒットしない。しかし、1文字トークン+レーベンシュタイン距離で検索した場合は、「CHODSHIP って何？」と「CHORDSHIP って何？」は距離1の類似文と判定し、正しいFAQの選択肢を提示してくれる。

一つの検索手法では期待される結果が得られない場合も、それぞれの特性を補完し合う形で複数の検索手法を組み合わせることで、ヒット率向上につながる（表-1）。組み合わせを間違えると、逆にヒット率を下げってしまうことになるため、組み合わせが難しい場合もある。筆者らの製品開発においては、この問題に対応するために、実際に業務に用いられているFAQのデータセットやチャットのログを収集し、分析を繰り返した。

表-1 検索手法の組み合わせ効果の例

| 検索方式                        | ヒット率  |
|-----------------------------|-------|
| 文字トークンでの検索                  | 70.3% |
| 単語トークンでの検索                  | 74.9% |
| 単語トークンでの検索と文字トークンでの検索の組み合わせ | 81.5% |

その結果、最適な組み合わせはデータセットによって全く異なる、というシンプルな事実が分かった。つまり、チャットボットで使用するデータセットが異なれば、それに適合するように異なった組み合わせに設定しなければいけないということである。富士通のサービスにおいては、それぞれの顧客が最適な設定でチャットボットを提供できるようにするために、製品導入時の精査サービスで、こういったチューニングを実施している。

### 発話のテクニック

前章では、ユーザーの入力内容に対して適切な回答を返すための様々な手法を述べた。しかし、これらの手法によってヒット率が向上したとしても、それだけでは単なる賢い検索エンジンである。的確な答えを返せるだけでなく、人間らしい会話ができることもチャットボットにとっては重要である。

チャットボットが上手にAIの振りをできるかどうかは、的確な答えを出せるかどうかではなく、人間らしいかどうかにかかってくる。実際、Elizaや人工無脳と呼ばれる古典的なチャットボットは、主にこの人間らしさの実装に重きが置かれていた。

人間らしいかどうかを決定づけるための要因は、チャットボットの発話である。例えば、同じ内容の答えであったとしても、会話のたびに文章を変化させるランダムな回答をすることで、より人間らしく感じられる。また、特定の人格をイメージさせるために、特徴的な語尾表現や方言の利用などを一貫して行うペルソナ、発言内容によって画面に表示するアバター画像の表情を変化させる（図-3）といった、発話に関する工夫が人間らしさをもたらす。

このように、人間らしさをうまく演出できれば、あたかもチャットボットが心を持っているように錯覚させることが可能となる。



図-3 発言内容に応じたアバターの表情変化

### む す び

本稿では、チャットボット実装に活用できるテクニックを紹介した。

話者の発言を的確に理解できることで賢さを示し、自らの応対で心を伝える、そのような賢さと心を併せ持ったチャットボットを作ること、これまで述べてきたように理論上は可能である。

とはいえ、賢さを獲得するためのチューニングにはどうしても試行錯誤の時間が必要であり、心を伝えるための演出には様々なアイデアの取捨選択と、ノウハウの蓄積が重要であることも分かってきた。筆者らは、これからもCHORDSHIPを継続的にエンハンスすることによって現場での実践を積み重ね、いつかは顧客との真のエンゲージメント（絆）を実現するチャットボットを生み出したいと考えている。

### 参考文献

- (1) 富士通：Customer Engagement Solution CHORDSHIP.  
<http://www.fujitsu.com/jp/services/knowledge-integration/chordship/>
- (2) Python Software Foundation：6.3. difflib — 差分の計算を助ける — Python 3.6.5 ドキュメント.  
<https://docs.python.jp/3/library/difflib.html>
- (3) gensim：models.doc2vec – Deep learning with paragraph2vec.  
<https://radimrehurek.com/gensim/models/doc2vec.html>

### 著者紹介



#### 倉知 陽一 (くらち よういち)

富士通（株）  
デジタルフロントビジネスグループ  
ビッグデータ、AIを活用したソリューションの企画、事業化に従事。



#### 川上 真一 (かわかみ しんいち)

富士通（株）  
デジタルフロント事業本部  
顧客接点高度化ソリューションの開発に従事。



#### 宮澤 孝彰 (みやざわ たかあき)

富士通（株）  
デジタルフロント事業本部  
顧客接点高度化ソリューションの開発に従事。