

SNS 投稿履歴から音声対話コンテンツへの変換手法

前田 一樹¹ 山本 大介¹ 高橋 直久¹

概要: 近年、Siri や Google 音声検索等の音声対話システムが広く普及している。また、我々は、MMDAgent という音声インタラクションシステム構築ツールキットを用いた研究を行っている。MMDAgent とは、名古屋工業大学国際音声技術研究所で開発された音声対話システム構築ツールキットであり、音声認識、音声合成、3D モデルの描画と物理演算などのモジュールを統合したシステムである。実行すると画面上にキャラクターが表示され、対話を行うことが可能である。また、FST と呼ばれる対話スクリプトを編集することで、対話内容を変更することが可能である。しかし、一般的なユーザにとってこのような音声対話コンテンツを手作業で作成することは難しい。FST に対するコマンドや編集に関する知識が必要なためである。本研究の目的は、前述の音声対話コンテンツ作成の難しさを解決する為、Twitter を元に簡単に音声対話コンテンツを作成することのできるシステムを提案することである。システムの実現のため、MMDAgent EDIT という音声対話コンテンツ作成システムを利用する。提案システムは、Twitter のツイートを MMDAgent EDIT の入力に変換するツイート変換機能、変換されたデータを MMDAgent EDIT へ登録するためのデータ登録機能を持つ。さらに、提案システムに基づいて実装したプロトタイプシステムを用いて評価実験を行い、提案システムの有用性を評価した。

Conversion Method from SNS Timeline to Voice Interaction Content

KAZUKI MAEDA¹ DAISUKE YAMAMOTO¹ NAOHISA TAKAHASHI¹

1. はじめに

近年、Siri や Google 音声検索等の音声対話システムが広く普及している。これらは、音声によってシステムとコミュニケーションを取りながら情報の取得や操作を行う事ができる。そこで、我々は音声インタラクション構築ツールキットである MMDAgent[1] を用いて、一般ユーザによる利用を前提とした音声対話システムの研究を行っている。MMDAgent は、FST ファイルと呼ばれる対話スクリプトの編集を行うことにより対話内容を変更することが可能である。FST ファイルは状態番号、次状態番号、遷移条件、遷移時に出力されるコマンドの 4 つ組から構成されており、編集には一定の知識を要する。そのため、一般的なユーザにとって、音声対話コンテンツを手作業で作成する事は困難である。

本研究では、このような音声対話コンテンツ作成の難し

さを解決するため、Twitter を元に簡単に音声対話コンテンツを作成するシステムを提案する。Twitter とは、半角 280 文字、全角 140 文字以内のメッセージや画像等をつぶやきという形で投稿できるソーシャルネットワーキングサービス (SNS) である。投稿のことをツイートと呼び、他者のアカウントのツイートを見るために購読することをフォローと呼ぶ。また、フォローしたユーザのツイートは、タイムラインと呼ばれるログに時系列順で表示される。

提案システムを実現するに当たって、Web ページから音声対話コンテンツを作成することのできるシステム、MMDAgent EDIT を使用する。MMDAgent EDIT は、専用の Web ページを通して「発話文」「認識キーワード」「認識キーワードの読み」「画像」「読み上げ時の感情」「読み上げ時の動き」を登録することで音声対話コンテンツを作成することができる。提案システムは、ツイート 1 件から前述の MMDAgent EDIT の登録項目を取得するためのツイート変換機能、また、変換されたデータを MMDAgent EDIT へ登録するためのデータ登録機能を有する。しかし、実現上の課題として、これらの登録項目の全てをユーザが

¹ 名古屋工業大学大学院工学研究科
Graduate School of Engineering, Nagoya Institute of Technology

考案し、それらを手で入力しなければならない為、膨大な時間がかかってしまう点が挙げられる。そこで、ツイート変換機能ではツイート1件を分析して全ての登録項目の抽出を行い、データ登録機能ではツイート変換機能で抽出されたデータを MMDAgent EDIT のデータベースへ SQL 文を用いて登録する。尚、本研究においては、MMDAgent EDIT の登録項目における「読み上げ時の感情」と「読み上げ時の動き」に関しては考慮せず、初期設定である「普通の感情」「普通の動き」を採用している。

2. 関連研究, 関連システム

2.1 音声インタラクション構築ツールキット MMDAgent

MMDAgent は、名古屋工業大学国際音声技術研究所で開発された音声対話システム構築ツールキットであり、音声認識、音声合成、3D モデルの描画と物理演算などの機能を統合したシステムである。音声認識では Julius を、音声合成では Open JTalk を、3D モデルでは MikuMikuDance 形式を、物理演算では Bullet Physics を採用している。MMDAgent は対話シナリオの記述形式に有限状態オートマトンの一種である有限状態トランスデューサ (FST) を採用している。FST をテキストで記述する際の記述言語を FST スクリプトと呼び、FST スクリプトが記述されたファイルを FST ファイルと呼ぶ。この FST ファイルを編集することで対話シナリオのカスタマイズが可能となっている。実行画面を図 1 に示す。ユーザが画面中央のキャラクター“メイちゃん”に話しかけると、認識した文字列に対応した言葉を音声によって返す。



図 1 MMDAgent 動作画面

2.2 MMDAgent EDIT

MMDAgent EDIT は、Web ブラウザを用いてユーザ

が手軽に編集を行うことのできる、MMDAgent による音声対話コンテンツ生成環境を構築したシステムである。MMDAgent EDIT を使用することで、MMDAgent の対話の編集を行うにあたって必要とされる FST への知識、手間を必要とすること無く、対話内容を投稿フォームから入力するだけで音声対話コンテンツを作成する事が可能である。

MMDAgent EDIT では、ユーザの1つの発話に対して1回の発話を行う、一問一答方式のコンテンツを作成することができる。1つの対話の登録のために必要な項目として、「発話文」「キーワード」「キーワードの読み」「画像」「読み上げ時の感情」「読み上げ時の動作」という6つの項目がある。「発話文」とは、認識キーワードを皮切りにメイちゃんが発話する文章の事である。「キーワード」とは、発話文の認識キーワードである。ユーザがキーワードを含んだ文章を発話すると発話文をメイちゃんが読み上げる。「キーワードの読み」とは、認識キーワードの読みであり、ユーザはこれを頼りにキーワードを発話する。「画像」とは、発話文やキーワードに関連する画像であり、後述するバルーンの中に貼り込まれる。「読み上げ時の感情」「読み上げ時の動き」とは、メイちゃんが発話文を読み上げる際の声色、表情、動作等の感情表現を示す。登録された項目は修正することも可能である。

MMDAgent EDIT を用いて作成されたコンテンツの例を図 2 に示す。一つの対話はバルーンという形でメイちゃんの背後に浮遊する。バルーンの中にはキーワードとその読みと画像が貼り込まれていて、ユーザはそれらを見て対話を行う。バルーン内のキーワードを読み上げると、該当するバルーンが強調表示され、メイちゃんがキーワードに対応する発話文を読み上げる。



図 2 MMDAgent EDIT で作成した音声対話コンテンツの例

2.3 ツイート読み上げ時間短縮システム

文献 [3] では、ツイートの一部を特徴語として抽出し、その特徴語を”メイちゃん”に読み上げさせ、そのツイート全文を読み上げるかどうかの確認を行うことで、タイムラインのツイートを全て読み上げる時間を短縮するシステムを提案している。ツイート内の特徴語を形態素解析による品詞分類と TF-IDF 法を用いて取得している点が本研究と同様である。文献 [3] にて扱われる特徴語は、「記号」「助詞」「助動詞」以外のすべての品詞について取得しているが、本研究における特徴語は、ユーザが読み上げる物である事を考慮しているため、より具体性が高いと思われる「名詞」のみに着目している。

また、文献 [3] では、TF-IDF 法と異なる手法として、ツイート文頭の数字を特徴語とする手法も提案している。本研究でも、TF-IDF 法以外にツイートの文頭から特徴語を取得する手法を提案しているが、本研究では、ユーザによる読み上げを考慮し、より意味の明瞭な特徴語を取得するため、文字ではなく、単語で区切りをつけて取得している点が異なっている。

2.4 SNS 連動型デジタルサイネージ

文献 [4] では、広報アカウントや一般ユーザが投稿したツイートからキーワード検索により興味のあるツイートを自動的に検索しサイネージに掲示するシステムを提案している。ツイート取得対象が単体であり、全てのツイートを取得している本システムに対し、ツイート取得対象が多数であること、キーワード検索によりツイートを選定していることが相違点である。また、文献 [4] ではツイートをテキスト形式でそのままサイネージに掲載しているが、本システムではツイートの特徴語を対話のきっかけとして、ツイートを音声対話形式で掲載している点が異なっている。

2.5 Twitter からの地域特徴語の自動抽出

文献 [5] では、ツイートが緯度経度情報とテキストを同時に含むデータであることを利用し、位置に関係した単語(地域特徴語)を抽出する手法を検討している。本システムでは、TF-IDF を既存式で利用しているが、文献 [5] では、緯度経度情報より抽出されるエリアの大きさによって変わる重み、ローカリティ L を加えて計算している点に違いがある。

2.6 タグクラウド共有に基づく協調的映像アノテーション

文献 [6] では、映像共有サイトにおける映像コンテンツに対して映像シーンごとの検索を可能にするため、映像シーンの内容を記述したアノテーションを自動的に付与する手法を提案している。アノテーションとして、ユーザが生成したコメントからコンテンツに関連するタグを TF-IDF を用いて自動的に抽出し、それらをタグクラウドとして複数

のタグを表示させている。本システムにおける TF-IDF 複数単語法は、このタグクラウドから構想を得ている。

2.7 Twitter に関する研究

他にも、文献 [7], [8], [9] など、Twitter の投稿を分析しデータマイニングを行ったり、ツイートからキーワードを抽出しイベント検出を行ったりと、Twitter を活用する研究は数多く存在する。しかし、本システムのように Twitter の投稿を音声対話コンテンツへ向けて変換するシステムは少なく、そこに新規性が見られる。

3. 提案システムの概要

3.1 提案システムの構成

提案システムの構成を以下に示す。提案システムの構成図は図 3 のようになる。提案システムはまず、Twitter に

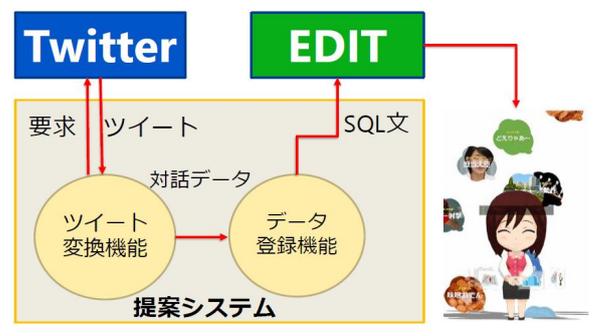


図 3 提案システム構成図

1 ユーザのツイート取得要求を行い、ツイートを取得する。取得されたツイートは、ツイート変換機能により対話データへと変換される。変換された対話データはデータ登録機能へと受け渡される。データ登録機能は、対話データを MMDAgent EDIT のデータベースへ受け渡すための SQL 文を発行する。MMDAgent EDIT は対話データを受け取り、音声対話コンテンツを作成する。

3.2 提案システムの機能

ツイート変換機能

Twitter へ 1 ユーザのツイートの取得要求を行い、ツイートを取得する。取得したツイート 1 件につき、MMDAgent EDIT の登録項目である「発話文」「キーワード」「キーワードの読み」「画像」をそれぞれ抽出する。つまり、ツイート変換機能は「発話文取得機能」「キーワード取得機能」「キーワードの読み取得機能」「画像取得機能」の 4 つに分かれる。「発話文取得機能」は、取得したツイートから URL を除き、発話文を取得する。「キーワード取得機能」は、発話文に対して形態素解析を行い、キーワードを取得している。提案手法では、3 つの取得法を提案している。この 3 つの取

得法に関しては後述する。「キーワードの読み取得機能」は、取得したキーワードを再度形態素解析し、読み仮名を取得する。「画像取得機能」は、ツイート内の画像を示す URL を解析することで画像付きツイートであるかどうかを判定し、画像付きツイートであった場合、URL を参照し画像を取得する。1 ツイートから取得される以上の 4 つのデータをまとめて「対話データ」とし、データ登録機能に受け渡す。

ユーザが MMDAgent EDIT の登録項目と情報ソースを見ながら対話データを思案するのではなく、全ての要素は本機能により機械的に取得されるため、自動変換が実現されている。

データ登録機能

ツイート変換機能から受け渡された対話データを、SQL 文を用いて MMDAgent EDIT のデータベースへ挿入する。MMDAgent EDIT の投稿フォームへの手入力をするのではなく、SQL 文を用いて機械的に登録処理を行うため、自動登録が実現されている。

4. 実現方法

ここでは、提案システムの実現法について述べる。まず、ツイート変換機能の詳細について説明し、データ登録機能の詳細について説明する。

4.1 ツイート変換機能

ツイート変換機能は、ツイート 1 件から発話文、キーワード、キーワードの読み、画像を取得する。つまり、ツイート変換機能は発話文取得機能、キーワード取得機能、キーワードの読み取得機能、画像取得機能の 4 つの機能から成っている。それぞれの機能の実現法を以下に示す。

4.1.1 発話文取得機能

Twitter API を用いてユーザのタイムラインを取得し、タイムライン内のツイートを投稿日時の新しいものから順に String 型リストへ格納することでツイートの取得が可能となる。ツイートの中には画像を含むツイートが存在し、画像は URL として示される。MMDAgent は、URL 等の意味のない文字列をそのまま読み上げてしまうため、リストへの格納前にツイートから URL を取り除く処理を追加した。尚、後述する画像取得機能にてツイート中に URL が含まれるかどうかの判定を行うため、URL は一度“replace”という文字列に置き換えられ、判定後、消去される。以上の操作により、ツイートから URL を削除したものを再び String 型リストに格納する事で、発話文の取得が完了する。

4.1.2 キーワード取得機能

キーワード取得機能では、3 種類のキーワード取得法を実装した。それぞれの取得法についての実現法を以下に述べる。

TF-IDF1 単語法

ユーザがバブルン内のキーワードを見て発話文を想像し、気になったものを読み上げる事を考えると、より具体性の高い単語を認識キーワードとして取得すべきである。そのため、まず、発話文取得機能により取得した発話文を形態素解析し、名詞以外の単語を全て除外する。これらを String 型のキーワード候補 2 次元リストに格納し、これを用いて各発話文中の各名詞についての TF-IDF 値を計算する。

TF-IDF は文書中の単語の重みの一種であり、特定の文書にしか出現しない単語の重要度は上がり、多くの文書に現れる単語の重要度は下がる。文書中の単語の TF-IDF を求める式を式 1 から式 3 に示す。tf(Term Frequency) は単語の文書内での出現頻度を表し、多くの文書に現れる単語の値が大きくなる。idf(Inverse Document Frequency) は逆文書頻度を表し、幾つもの文書に現れる単語の値は小さくなる。tfidf は tf と idf の積で求められる。

$$tf(t, d) = \frac{n_{t,d}}{\sum_{s \in d} n_{s,d}} \quad (1)$$

$$idf(t) = \log \frac{N}{df(t)} + 1 \quad (2)$$

$$tfidf = tf \cdot idf \quad (3)$$

$n_{t,d}$ = 単語 t の文書 d 内での出現回数

$\sum_{s \in d} n_{s,d}$ = 文書 d 内の全ての単語の出現回数の和

N = 全文書数

$df(t)$ = 単語 t が出現する文書数

TF-IDF 値は以下の式 4 から式 6 により算出する。tf はあるキーワード候補 2 次元リスト中の TF-IDF 値を計算したい単語 (目的単語) の出現回数と、キーワード候補 2 次元リスト中の全ての単語数を用いて算出する。idf については、多くの文書に出現する単語の重要度を下げるため、キーワード候補 2 次元リストだけでなく、3000 件の無作為に選ばれた日本語ツイートから名詞のみを抜き出し 2 次元リスト化した重要度補完 2 次元リストを使用し、計算した。TF-IDF 値が最も高い名詞をその対話文中のキーワードとして、String 型配列に格納する。

$$tf = \frac{\text{ある文書の目的単語数}}{\text{ある文書の単語数}} \quad (4)$$

$$idf = \log \frac{\text{全文書数}}{\text{目的単語が含まれる文書数}} + 1 \quad (5)$$

tf · idf = ある文書での目的単語の重み (6)

TF-IDF 複数単語法 取得法 1 で示した TF-IDF 値の計算を行い算出された TF-IDF 値に基づき、TF-IDF 値の高い名詞から順に 3 単語をキーワードとして取得する。尚、名詞の少なく短いツイートである場合は、2 単語以下になる。取得した単語をカンマで区切り、String 型配列に格納する。

文頭 5 単語法 発話文を形態素解析した後、文頭から 5 単語をキーワードとして取り出す。取り出されたキーワードを String 型配列に格納する。

キーワードの例 それぞれの取得法で取得されるキーワードの例を以下の表 1 に示す。例文を「山本研究室の広報を担当します。よろしくお願ひします。」とし、TF-IDF1 単語法を 1、TF-IDF 複数単語法を 2、文頭 5 単語法を 3 とする。

表 1 評価項目

取得法	取得されるキーワード
1	広報
2	山本研究室, 広報, 担当
3	山本研究室の広報を担当

4.1.3 キーワードの読み取得機能

キーワードの読み取得機能は、前節で取得したキーワードを再度形態素解析し、キーワードの読み仮名を取り出す機能である。取り出された読み仮名はカタカナであるが、MMDAgent EDIT では仕様上ひらがなで登録する必要があり、カタカナをひらがなへ変換しなければならない。これを解決するため、ひらがなもカタカナも文字コード上で対応する並びになっていることを利用し、文字コード同士の計算によって文字コードを変換し、ひらがなに変換している。ひらがなに変換された読み仮名は String 型の配列に格納される。各取得法によって取得されたキーワードに対する読み仮名の例を表 2 に示す。尚、取得法の番号とキーワードは表 1 に従う。

表 2 評価項目

取得法	取得されるキーワード
1	こうほう
2	やまもとけんきゅうしつこうほうたんとう
3	やまもとけんきゅうしつこうほうをたんとう

4.1.4 画像取得機能

画像取得機能は、画像付きツイートであった場合、その画像を取得する機能である。画像は、ツイート中に URL として現れる。Twitter API を用いてユーザのタイムライン上の画像が含まれるツイートの URL を解析し、内容を jpg 形式で保存する。ファイル名は” Twi [取得した順番].jpg ”として保存する。画像付きツイートと画像ファイルそのもの

の紐付けを行うため、発話文取得機能で示した、URL の代わりとなる “replace” を利用する。取得したツイート中に URL を示す “replace” が存在したら、ツイートの順番と同様の添字を持った初期値 null の String 型配列に、ツイートの順番と同じ番号を格納する。これにより、配列の添字と中身、画像ファイルのファイル名に付与された番号を順に参照することで、画像付きツイートと画像ファイルを紐付ける事が可能である。

4.2 データ登録機能

MMDAgent EDIT はデータベースにより対話を管理している。MMDAgent EDIT のデータベースは以下の表 3 から表 4 で示される属性により構成されている。尚、本機能を実装するにあたって値の設定の必要がない属性は省略している。

表 3 MMDAgent EDIT データベースの属性 1

属性名	id	content_id	user_id	keyword	yomi
型	serial	serial	text	text	text

表 4 MMDAgent EDIT データベースの属性 2

属性名	status	speech	expression	motion	imagefile
型	text	text	text	text	bytea

id とは、バルーン ID を示しており、主キーとなっている。対話ごとに固有の ID を付与するため、0 から順に数字を挿入する。content_id ではコンテンツ全体の ID を示しており、ここでは 1 と入力する。user_id はコンテンツ管理者の ID を示しており、ここでは管理者としてログインするため admin と入力する。keyword はバルーン内に表示されるキーワードを示しており、ツイート変換機能で取得したキーワードリストの中身を順に挿入する。yomi はキーワードの読みを示しており、ツイート変換機能で取得した読み仮名リストの中身を順に挿入する。status は対話の状態を示しており、非公開と公開を選択できる。全て公開にするため open と入力する。speech はメイちゃんが発話文を示しており、ツイート変換機能で取得した発話文リストの中身を順に挿入する。expression はメイちゃんの読み上げ時の感情を示しており、本研究では読み上げ時の感情を考慮していないため、通常感情を示す mei_voice_normal を入力する。motion はメイちゃんの読み上げ時の動作である。expression と同様、本研究では読み上げ時の動作を考慮していないため、通常動作を示す mei_normal を入力している。imagefile はバルーン内の画像を示しており、ツイート変換機能における画像取得機能により用意した紐付け配列を利用して画像を挿入する。

以上の操作を SQL 文を用いて実現する。各データの挿

入においては INSERT 文を用いて該当する属性に適切なデータを挿入している。画像の挿入において、紐付け配列の添字と内容が一致したときに、紐付け配列の添字と同じ id を持つバブルへ画像を挿入している。画像は Twi [取得した順番].jpg というファイル名で保存されているので、挿入時に画像のファイル名を表す変数をインクリメントすることで正しい画像を挿入することが可能になる。

5. プロトタイプシステム

プロトタイプシステムは、提案したツイート変換機能とデータ登録機能を持ったプログラム、MMDAgent EDIT で構成されている。プログラミング言語として Java を利用し、Java プログラムとして実装した。開発環境は Eclipse を用いた。MMDAgent EDIT の動作のため、Tomcat8.5 を使用した。また、MMDAgent EDIT 内のデータベースの構築と管理のため、PostgreSQL と pgAdminIII を使用した。作成された音声対話コンテンツを実行するため、MMDAgent 1.7 を使用した。

5.1 プロトタイプシステム使用時の流れ

このプロトタイプシステムを使用して、音声対話コンテンツへ変換されたツイートを確認する流れは以下の通りである。

- (1) キーワード取得法を選択し、プログラムを実行する
- (2) MMDAgent EDIT へアクセスし、管理者としてログインする
- (3) コンテンツの一覧から作成されたコンテンツを選択する
- (4) 画面右上のダウンロードをクリックし、ダウンロードされたコンテンツを実行する
- (5) 起動した MMDAgent に対し、背景に浮かぶバブル内の認識キーワードを読み上げる
- (6) 読み上げられたキーワードを含んだバブルが強調表示され、メイちゃんが発話文(ツイート)を読み上げる

作成された音声対話コンテンツの例を以下の図 4 に示す。

6. 評価実験

6.1 評価実験の目的

実験の目的は、プロトタイプシステムを使用して作成された音声対話コンテンツを使用するに当たって、認識キーワードが「発話文との関連性」「読み上げのし易さ」「意味のわかり易さ」の点で妥当なものであるかどうかを検証する事である。ジャンルの違う 3 種のツイッターアカウントに、提案した 3 つのキーワード取得法をそれぞれ適応し音



図 4 作成されたコンテンツ (TF-IDF1 単語法)

声対話コンテンツを作成後、アカウントのジャンルと取得法の特性を照らし合わせながら、比較検証する。

6.2 評価実験の方法

被験者(本学学生 7 人)に、プロトタイプシステムを使用して作成されたコンテンツを使用してもらう。ツイッターアカウントを 3 種(研究室の広報、飲食店の広報、美術館の広報)用意する。それぞれ、使用されている用語の専門性に差がある。研究室の広報アカウントはツイート中に専門用語と一般的な用語が同程度に存在している。飲食店の広報アカウントは、一般的な用語がほとんどで、美術館の広報アカウントは専門用語と造語が多い。それぞれに対してキーワード取得法の異なるコンテンツを 3 つずつ作成する。被験者に使用法を説明し、合計 9 つのコンテンツを全て使用してもらう。その後 5 段階評価のアンケートを実施し、回答してもらう。

6.2.1 実験の手順

実験の手順について以下に述べる。

手順 1 音声対話コンテンツの使用法について説明し、用意した音声対話コンテンツを使用してもらう

手順 2 一つのコンテンツを使用する毎に、逐一アンケートに答える。

6.2.2 アンケート内容

アンケート項目として 3 項目を用意し、5 段階評価を行う。評価基準は「1(当てはまらない), 2(やや当てはまらない), 3(どちらとも言えない), 4(やや当てはまる), 5(当てはまる)」である。以下に、アンケート内容をまとめた表 5 を示す。項目 1 から 3 に関しては、手順 2 で行う。

表 5 各キーワード取得法に対する評価項目

項目番号	質問内容
Q1	キーワードから連想される発話文は想像通りだ
Q2	キーワードは言葉として発しやすい
Q3	キーワード自体の意味はわかりやすい

6.3 結果と考察

アンケートの結果を質問毎に図 5 から図 7 に示す。各図の青軸は TF-IDF1 単語法、橙軸は TF-IDF 複数単語法に対する評価、灰軸は文頭 5 単語法に対する評価である。また、各図左から、研究室、飲食店、美術館のツイッターアカウントをソースとして作成した音声対話コンテンツに対する評価がアカウント毎に並んでいる。

「発話文との関連性」に対する評価値を示す図 5 を見ると、アカウント毎の差はあまり見られないが、TF-IDF1 単語法の評価が落ち込み、TF-IDF 複数単語と文頭 5 単語法の評価が高いことがわかる。これは、取得される単語数が多いほど認識キーワードと発話文との関連性が高くなることを示している。次に、「読み上げのし易さ」に対する評価値を示す図 6 を見ると、これもアカウント毎の差はあまり見られないが、TF-IDF 複数単語法、文頭 5 単語法は低い評価を受け、TF-IDF1 単語法は高評価を受けた。ここから、読み上げに関しては単語数が少ないほど行いやすいということが読み取れる。最後に、「意味のわかり易さ」に対する評価値を示す図 7 を見ると、TF-IDF1 単語法と文頭 5 単語法ではアカウント毎の差は見られないが、TF-IDF 複数単語法では専門用語と造語の多い美術館のコンテンツの評価値が下がっている事が見て取れる。これは、ユーザにとって馴染みのない単語が複数並んでしまうとキーワード自体の意味が難解なものになってしまうということを示している。また、TF-IDF1 単語法、TF-IDF 複数単語法は高い評価を得ているが、文頭 5 単語法は低い評価を受けている。これは、単語数が多いほどキーワード自体の意味が分かりづらくなっていることを示している。

以上より、認識キーワードの読み上げの容易さ、意味の明瞭性に関しては TF-IDF1 単語法が最も適しているが、発話文との関連性に関しては単語数の多い TF-IDF 複数単語法、文頭 5 単語法が適しているということがわかる。また、コンテンツ作成元が使用している用語の専門性が高いほど、TF-IDF 複数単語法で取得されるキーワードは難解なものになってしまうため相応しくないということも見て取れる。

以上を踏まえた上で最適な取得法を考えると、認識キーワード自体は文頭 N 単語法、またはコンテンツ作成元の内容に応じて TF-IDF 複数単語法を使用し、キーワードの読みに関しては TF-IDF1 単語法で取得されたキーワードの読みを使用することで、より実用性の高い認識キーワードが得られると考える。

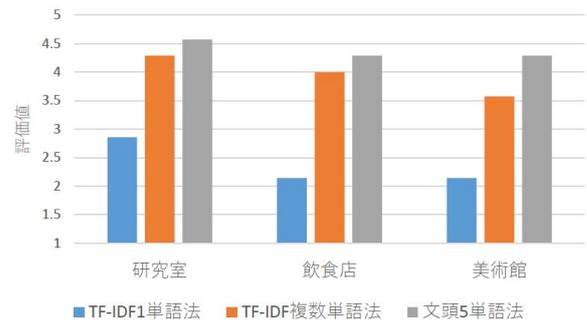


図 5 Q1 の評価結果

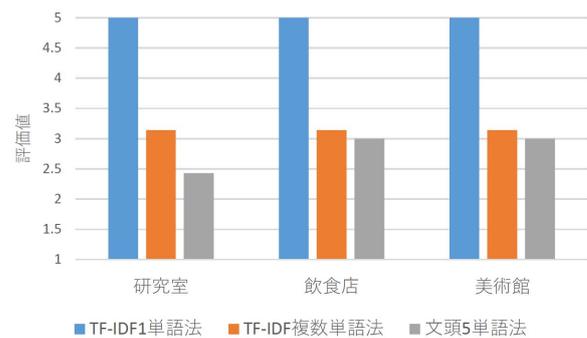


図 6 Q2 の評価結果

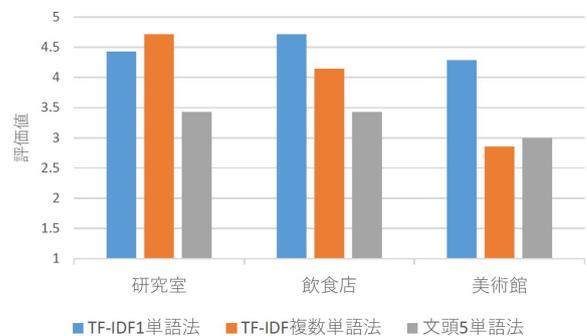


図 7 Q3 の評価結果

7. おわりに

本研究では、SNS 投稿履歴から音声対話コンテンツへの変換手法を提案し、ツイートを発話文や認識キーワードへ変換後、データベースへ登録し、MMDAgent EDIT を用いて素早く簡単に音声対話コンテンツを作成するシステムの実現法について述べた。また、提案した実現法を元にプロトタイプシステムを実装した。そして、評価実験として実際にプロトタイプシステムから生成された音声対話コンテンツを利用してもらうことで、システムの有用性を示した。

評価実験により、各取得法の長所と短所を明らかにした。発話文との関連性に関しては単語数の多い取得法が高い評価を得た。反対に、読み上げのし易さでは単語数の少ない取得法が高い評価を得た。また、意味の明確さに関しては、単語数が少ないものほど高評価を得たが、TF-IDF 複数単語法に関しては、コンテンツ作成元の専門性が高いほど低い評価を受けた。また、以上の結果を踏まえて、認識キーワードは文頭 5 単語法、またはコンテンツ作成元の内容に応じて TF-IDF 複数単語法を使用し、キーワードの読みに関しては TF-IDF1 単語法で取得されたキーワードの読みを使用することで、より実用性の高い認識キーワードが得られると考察した。

今後の課題として、評価実験で考察したキーワード取得法の実装と評価、また、キーワードが重複してしまう場合があるため、重複した場合次に特徴量の高い単語をキーワードとする機能を実装する事が挙げられる。

8. 謝辞

本研究は JSPS 科研費 25700009, および、総務省 SCOPE の助成を受けたものです。この場を借りて、感謝の意を表します。

9. 参考文献

参考文献

- [1] 李晃伸, 大浦圭一郎, 徳田恵一, 魅力ある音声インタラクションシステムを構築するためのオープンソースツールキット MMDAgent, 電子情報通信学会技術研究報告.NLC, 言語理解とコミュニケーション, Vol.111, No.364, pp. 159-164, 2011.
- [2] 山本大介, 堤修平, 打矢隆弘, 内匠逸, Web ユーザによる音声対話コンテンツ生成環境の構築とそれに基づく実証実験の評価 電子情報通信学会技術研究報告., Vol.115, No.346, pp. 111-116, 2015.
- [3] 三浦直明 Android 版 MMDAgent を用いた Twitter のタイムライン読み上げ時間の効率化 名古屋工業大学情報工学科卒業論文 pp. 5-15 (未公刊)
- [4] 近藤貴裕, 杉山航平, 赤堀優志, 大多和均, 渡邊貴之, SNS 連動型デジタルサイネージにおけるコンテンツ推薦機能, 情報処理学会第 76 回全国大会講演論文集, pp. 135-136, 2014.
- [5] 伊藤晶, 荒川豊, 田頭茂明, 福田晃, Twitter からの地域特徴語の自動抽出に関する一検討, 情報処理学会第 75 回全国大会講演論文集, pp.101-102, 2013.
- [6] 山本 大介, 増田 智樹, 大平 茂輝, 長尾 確, タグクラウド共有に基づく協調的映像アノテーション, 人工知能学会論文誌, Vol.25, No. 2, pp. 243-251, 2010.
- [7] Pak, Alexander, and Patrick Paroubek. Twitter as a corpus for sentiment analysis and opinion mining, Proceedings of the LREC 2010, pp. 17-23, 2010.
- [8] Sakaki, T., Okazaki, M., & Matsuo, Y. (2010, April). Earthquake shakes Twitter users: real-time event detection by social sensors, Proceedings of the 19th international conference on World wide web, pp. 851-860, 2010.
- [9] Java, A., Song, X., Finin, T., & Tseng, B. (2007, August). Why we twitter: understanding microblogging usage and communities. In Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis, pp. 56-65, 2007.