意味表現学習におけるツイート分散表現の解釈性評価と可視化の提案

芥子　育雄† 　松田　義貴†† 　鈴木　優†† 　吉野幸一郎 †† 　中村　哲††

† 福井工業大学 工学部 電気電子工学科　〒 910–8505 福井市学園 3 丁目 6–1
†† 奈良先端科学技術大学院大学 情報科学研究科　〒 630-0192 生駒市高山町 8916–5
E-mail: †keshi@fukui-ut.ac.jp, ††{matsuda.yoshitaka.mq2,ysuzuki,koichiro,s-nakamura}@is.naist.jp

あらまし　筆者らは，ニューラルネットワークにより獲得される分散表現において，有限個で意味を代表する特徴単語で隠れノードを表現し，辞書を用いてシードベクトルを与えることにより，人が解釈可能な形式による分散表現の学習手法を提案した．本辞書には，264 種類の特徴単語と約 2 万語の基本単語との意味的・連想的関連性を記述した単語意味ベクトル辞書を導入した．本稿では，クラウドソーシングにより収集を行った京都観光に役立つツイート 17,761 件（うち，食事 3232 件，交通 1619 件，天気 6395 件に人がカテゴリ付与）を対象に，教師なしの意味表現学習によりツイートに付与された特徴単語の重みの精度を評価した．17,761 件のツイートを特徴単語「食」の重みが大きい順にランキングし,「食事」ツイートの再現率 10% 時点の適合率 98.8%（同様に「交通」ツイートでは特徴単語「交通 · 輸送」の適合率 90%,「天気」ツイートでは特徴単語「気象 · 気候」の適合率 92%）を示した．また，特定の特徴単語に関連したツイートをフィルタリングし，他の特徴単語との関連性によりツイートを可視化する手法を提案する．

キーワード　分散表現，単語意味ベクトル辞書，パラグラフベクトル，word2vec，Twitter，カテゴリ分析，可視化，インタフェース

1. はじめに

IoT やソーシャルメディアの進展により，あらゆる人間活動が非構造化テキストデータとして蓄積されようとしている．例えば，視聴した TV 番組や閲覧した Web サイトおよびソーシャルセンサーとしての Twitter 情報 [1] が蓄積される．しかし，各領域データはスモールデータであり，このため各領域データのみによる価値発見は限定的になるが，他の領域と掛け合わせることにより，価値創出の可能性が高まる．

Mikolov らが 2013 年に発表した word2vec は，文脈情報を素性としてニューラルネットワークにより学習を行うと，語義の似た単語や語句が似たような重みをもつベクトルを構築することができると報告されている [2], [3], [4]．2014 年には単語を文書に拡張したパラグラフベクトルが提案され，発表時には様々な自然言語処理タスクにおいて最高水準を示した [5]．これらの単語や文書ベクトルは分散表現と呼ばれる．

非構造化テキストデータから価値を取り出すことが，ニューラルネットワークにより獲得される分散表現に期待されている．課題は，Twitter などの短文テキストはデータスパース性があること，およびニューラルネットワークにより獲得される特徴量の解釈が困難なことである．データスパース性とは，解析対象テキストの単語の多くが，分散表現を獲得した語彙辞書に含まれないことを指す．特徴量の解釈が困難とは，分散表現の各次元の意味や概念が明らかでないため，分散表現がブラックボックスとなっていることである．

非構造化テキストデータの分散表現学習において，スモールデータでも十分な精度が出て，異なる領域で学習された分散表現が共通に利用できれば，その有効性は格段に高まる．そこで，我々はニューラルネットワークにより獲得される分散表現に専門家が構築した単語意味ベクトル辞書 [6] を反映する手法を提案した [7], [8], [9], [10]．本辞書では単語は特徴単語との関係により意味を表現する．特徴単語数は 264 種類とした [8], [10]．特徴単語との関係を付与した単語を基本単語と呼ぶ．ニューラルネットワークとしてパラグラフベクトル [5] を用いて，その隠れノードに特定の意味（特徴単語）を割り当てる．単語意味ベクトル辞書を元に約 2 万語の基本単語に初期重み（シードベクトル）を設定することにより，学習後も重みの大きな隠れノードの意味が維持されることを示した．また，スモールデータに対しても従来のパラグラフベクトルと比較して，タスクの精度が高まることを確認した [9], [10].

本稿では，パラグラフベクトルの隠れノードの意味が維持される特徴を活かしたアプリケーションとして，収集したツイートのカテゴリ分析が可能な可視化手法を提案する．Twitter のような短文テキストのデータスパース性の課題を解消するため，最初に 120 万記事の日本語 Wikipedia コーパスを元に語彙辞書を作成し，単語ベクトルを学習させる．この単語ベクトル学習のシードベクトルに単語意味ベクトル辞書を用いる．次に Wikipedia コーパスで学習させた単語ベクトルをシードベクトルとして，収集したツイートのパラグラフベクトルを学習させる．ツイートベクトルの任意の特徴単語（あるいはその組み合わせ）の重みを概念軸（x 軸，y 軸）として，x-y 平面にツイートを可視化する．特定の特徴単語の重みが大きなツイートは，その特徴単語が表現するカテゴリに分類されるため，可視化することによりカテゴリ分析が可能となる．

本提案の可視化は，分散表現の解釈性が前提となる．このため，クラウドソーシングを利用して構築したカテゴリ分類のベンチマークを用いて定量的評価を行った．本ベンチマークは，京都観光に関する場所の様子が分かる 17,761 ツイートから構

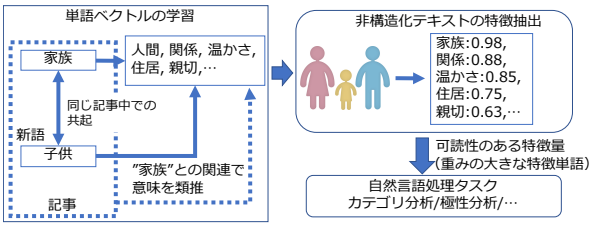

図 1　本研究のターゲット

成され，8 種類のカテゴリ（食事，交通，天気，混雑，イベント，景観，観光地，その他）の中から，適切なカテゴリを複数可で付与してもらった．食事カテゴリが付与されたツイート数は 3232 件だが，全ツイートを特徴単語「食」の重みが大きい順にランキングし，再現率 1%となる上位 32 件では食事が付与されたツイートの適合率 100%，再現率 10%となる上位 320 件では適合率 98.8%と非常に高い適合率を示した．

2.　研究の目的

本研究のターゲットを図 1 に示す．　単語ベクトルの学習，非構造化テキストの特徴抽出，自然言語処理タスクから構成される．単語意味ベクトルは，基本単語と特徴単語との関係を，関連あり，無しのバイナリ値として表現する．基本単語は，百科事典や新聞記事の説明に使われる用語，WWW ホームページの分類用語，取扱説明書などで使われる操作用語，および形容詞などの感性語から 2 万 336 語を選択した [6]．特徴単語は，概念的分類を体系的に網羅する 264 種類の概念であり，多くの特徴単語は，「人間」「関係」「温かさ」などの単語に対応する．単語意味ベクトルの各次元は特徴単語に対応する．単語意味ベクトル辞書は，基本単語に関連する特徴単語がリストされた辞書である．これら選択した基本単語に対して，辞書編纂の専門家が，論理的関連性と連想的関連性から，特徴単語を付与した．論理的関連は，基本単語に対して特徴単語が表 2 に示すような直接的関連性を有するものを指す．連想的関連は，基本単語に対して特徴単語が感覚的関連性，連想により想起される関連性を有するものを指す．例を表 3 に示す．特徴単語の上位概念，大分類は分類上の目安であり，付与判断の基準は特徴単語そのものである．例えば，特徴単語「温かさ」は上位概念「物理的特性」の下に分類されているが，“ 心の温かさ ”からの連想によって基本単語「愛」に付与する．

単語の論理的，連想的関係が辞書で表され，word2vec のような分散表現学習は単語の文脈を学習する．従って，word2vec のようにランダムな初期値から文脈を学習するのではなく，辞書と分散表現学習の両方を統合することにより，データスパース性の問題を解決することが期待される．人は辞書から単語の概念を理解し，それを少数の例文から使用方法を学ぶことができる．人が言葉を学習する仕組みを模倣することにより，辞書と分散表現学習を統合した単語ベクトルの学習を実現する．

非構造化テキストからの特徴抽出は，単語ベクトルに基づいて，教師なし学習により特徴量を抽出することができる．Twitter を対象とした我々の研究では，重みの大きな特徴単語

表 1　特徴単語の分類

大分類	上位概念	特徴単語例
人間 · 生命	人間	人間，人名，男性，女性，子供，…
	生物	動物，鳥類，虫，微生物，植物，…
人間環境	人造物	道具，機械機器，建造物，…
	交通 · 通信	通信，交通 · 輸送，自動車，…
自然環境	地域	地名，国名，日本，都会，地方，…
	自然	陸地，山岳地，天空，海洋，環境，…
抽象概念	精神 · 心理	感覚，感情，喜楽，悲哀，…
	抽象概念	変化，秩序 · 順序，勢力 · 程度…
物理 · 物質	運動	運動，停止，動的，静的，…
	物理的特性	温かさ，重さ，軽さ，柔軟…
文明 · 知識	人文	民族人種，知識，言論発話，…
	学術	数学，物理学，天文学，地学，…

表 2　論理的関連による特徴単語の付与基準

論理的関係	基本単語	特徴単語
集合包含	秋	季節
同義関係	アイデア	思想
部分全体関係	足	人間の身体

表 3　連想的関連による特徴単語の付与基準

基本単語	特徴単語
愛	優しさ，温かさ
アップ	経済，映像
足	自動車，交通 · 輸送

トップ 5 の 52%がツイートと関連しているとのユーザ評価を得ている [9], [10]．

本特徴量が分類などのタスクに利用される．大きな重みの特徴単語が類似したテキストは概念が近いと考えられるので，教師データを作成しなくても特徴単語（あるいはその組み合わせ）を概念軸としたテキスト分析や，特徴単語は共通なため，異なる領域で学習した分散表現を掛け合わせた分析が可能となる．

3.　関 連 研 究

word2vec のようにランダムな初期値から得られた分散表現の各次元に意味を与える手法の提案がなされている．最初に，ランダムな初期値からではなく，コーパスの共起パターンから導出された行列分解において，単語の分散表現にスパース性と非負性を導入することにより，各次元の解釈可能性が高まることが提案された (Non-Negative Sparse Embedding) [11]．ランダムな初期値から学習した分散表現の各軸に意味付けする方向の研究としては，非負オンライン学習 Skip-gram モデル [12] とスパース制約付きオンライン学習 SparseCBOW モデル [13] がある．提案手法の優位性と差異を以下に示す．

- 関連研究では次元毎に上位語から人が意味を類推する必要がある．提案手法は各次元の概念分類を固定している．
- 関連研究では次元の意味は学習領域に依存する．提案手法では各領域に共通である．
- 関連研究は単語の学習のみで評価を行なっている．提案手法は，単語の学習と比較して，より多くの学習繰り返しが必要な文書の特徴抽出でも効果を検証済みである．

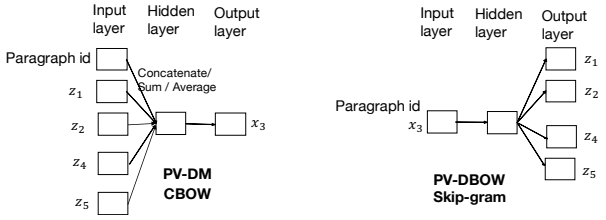

図 2　パラグラフベクトルの 2 種類のモデル

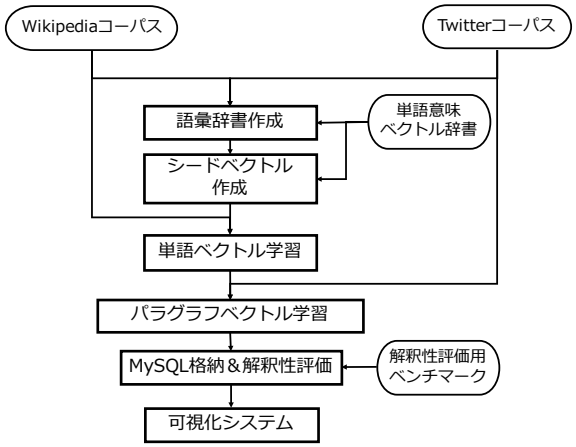

図 3　提案システムの流れ

4. 提 案 手 法

ニューラルネットワークの課題は，特徴量の解釈が困難なため，結果説明が出来ないことである．研究の目的に示した通り，我々は人の知見を分散表現に反映した意味ベクトル辞書を構築した．これをシードベクトルとすることにより，ニューラルネットワークの課題を解決できると考えた．本提案手法ではニューラルネットワークとして，パラグラフベクトルを用いる [5]．パラグラフベクトルには，図 2 に示す通り，2 種類のモデルがある．PV-DM（paragraph vector with distributed memory）モデルは語順の情報を保持し，周辺単語から中心単語を予測するモデルである．PV-DBOW (paragraph vector with distributed bag of words) モデルはパラグラフ中の周辺単語を予測するためのモデルである．PV-DBOW モデルにおいて単語ベクトルを学習する場合は，word2vec の Skip-gram モデルが用いられる [3]．提案手法ではツイートのパラグラフベクトル学習に PV-DBOW モデルを用いて，単語ベクトルの学習には Skip-gram モデルを用いる．

提案手法の流れを図 3 に示す．以下の節では提案手法の処理内容を順に説明する．

4.1 語彙辞書作成

コーパスから語彙辞書を作成する時に以下の 2 種類の初期ベクトルを最初に構築する．

- 264 種類の特徴単語を語彙として追加し，その初期ベクトルを特徴単語に対応する次元を 1 とする 264 次元の One-hot ベクトルとする．
- 264 種類の特徴単語を除く，コーパスから抽出された全単語（コーパスに含まれる基本単語を含む）の初期ベクトルは，264 次元のゼロベクトルとする．

語彙辞書の各単語は，入力ベクトルと出力ベクトルの 2 種類のベクトルを持つ．入力ベクトルは入力層の単語と隠れ層の 264 種類のノードとの間の重みであり，出力ベクトルは出力層の単語と隠れ層の 264 種類のノードとの間の重みに対応する．入力ベクトル，出力ベクトル共に上記の初期化を行う．

4.2 シードベクトル作成

辞書の単語に対する定義文を再帰展開することにより単語ベクトルを生成する手法は提案されている [14]．単語意味ベクトル辞書は，264 種類の特徴単語で基本単語を定義しているとみなすことが出来る．特徴単語も基本単語となるため再帰展開が必要だが，定義文が 264 語に限定されるため，数回展開すれば収束する．また，単語ベクトルを辞書の関連語に合わせて修正する手法も提案されている [15]．ここでは，Faruqui らのレトロフィッティングツール (注1) [15] を用いて，単語意味ベクトル辞書を再帰的に展開することにより，基本単語のシードベクトルを生成した．オンラインアップデートのアルゴリズムを以下に示す．

$$\mathbf{q_i} = \frac{\sum_{j:(i,j)\in E} \beta_{ij}\mathbf{q_j} + \alpha_i \hat{\mathbf{q}}_\mathbf{i}}{\sum_{j:(i,j)\in E} \beta_{ij} + \alpha_i} \quad (1)$$

$\mathbf{q_i}$ は基本単語 w_i のレトロフィットされた単語ベクトル，$\hat{\mathbf{q}}_\mathbf{i}$ は w_i の初期ベクトル，α_i は初期ベクトルの重み；現在は w_i に付与された特徴単語 w_j の数に設定．$\mathbf{q_j}$ は w_j のレトロフィットされた単語ベクトル, β_{ij} は基本単語 w_i に付与された特徴単語 w_j の重み；現在は重み β_{ij} は 1 に設定. 式 1 は基本単語 w_i の初期ベクトル $\hat{\mathbf{q}}_\mathbf{i}$ に重み α_i を掛けて, w_i に付与された特徴単語 w_j のレトロフィットされた単語ベクトル $\mathbf{q_j}$ に重み β_{ij} を掛けることによって得られるベクトルを足し合わせて，両方の重みで割ったものがレトロフィットされた単語ベクトル $\mathbf{q_i}$ となることを示す．コーパス中の全基本単語に対して，単語意味ベクトル辞書を用いたレトロフィットが終われば，レトロフィットされた基本単語 w_i の単語ベクトル $\mathbf{q_i}$ を初期ベクトル $\hat{\mathbf{q}}_\mathbf{i}$ として，本手続きを繰り返す．この手続きを 10 回繰り返すことにより，各基本単語と 264 種類の特徴単語との間の関係は平均 9 から，平均 100 にまで増加する．これは基本単語に付与された特徴単語を再帰的に展開することにより，基本単語と関係のある特徴単語の数が増加するためである．

本アルゴリズムの特徴は以下の通りである．

- レトロフィットされた単語ベクトルは，元の単語ベクトルに近い．特徴単語でもある基本単語は，対応する特徴単語の重みが大きい One-hot ベクトルに近い．
- レトロフィットされる基本単語に付与された特徴単語が，基本単語として展開されない場合は，各特徴単語の重みはほぼ等しくなる，特徴単語が基本単語として展開される場合は，展開される特徴単語の数に応じて重みが小さくなる．

基本単語「旅行」のシードベクトルを設定した Skip-gram（PV-DBOW）モデルの例を図 4 に示す．隠れ層は特徴単語に

(注1)：https://github.com/mfaruqui/retrofitting

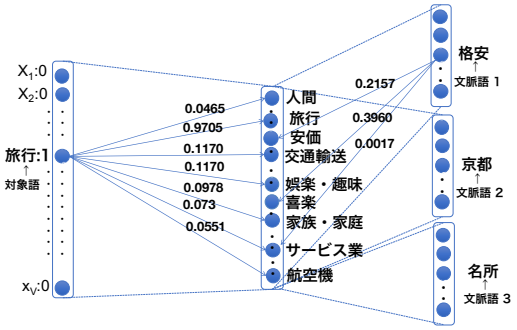

図 4　Skip-gram モデルへのシードベクトルの設定

表 4　keyword_master テーブル

カラム名	データ型	意味
keyword_master_id	整数	特徴単語の ID (0〜263)
word	文字列	日本語の特徴単語
english	文字列	英語の特徴単語

表 5　tweets テーブル

カラム名	データ型	意味
id	整数	レコードの識別子
tweets_id	整数	ツイートの ID
content	文字列	特徴抽出を行なったツイート本文
words	文字列	代表語

表 6　tweets_keywords テーブル

カラム名	データ型	意味
id	整数	レコードの識別子
tweets_id	整数	ツイートの ID
point	固定小数点	特徴単語の重み
keyword_master_id	整数	特徴単語の ID

対応する 264 種類のノードから構成される．入力層と隠れ層の間の重みが対象語（例では，「旅行」）のレトロフィットされたシードベクトル（入力ベクトル）である．出力層の各ノードは対象語の周辺単語（例では，「格安」，「京都」，「名所」）から構成され，出力層の各ノードと隠れ層の間の重みはレトロフィットされたシードベクトル（出力ベクトル）を示す．入力ベクトルと出力ベクトルの繰り返し回数は，タスクの精度を元に入力ベクトルは 10 回，出力ベクトルは 1 回と決めた [10]．

4.3　単語ベクトル学習

単語意味ベクトル辞書は約 2 万語と少ないため，この基本単語の入力・出力ベクトルを Skip-gram モデルのシードベクトルとして，約 120 万記事の日本語 Wikipedia コーパスを用いて，単語ベクトルの学習を行う．日本語類似性データセット [16] を用いた評価により，学習回数（Epoch 数）3 を採用した [10]．

4.4　パラグラフベクトル学習

分析対象となる Twitter コーパスを用いて，パラグラフベクトル（PV-DBOW モデル）の学習を行う．シードベクトルには Wikipedia コーパスで学習した単語ベクトルを用いる．パラグラフベクトルの学習回数は，タスクの精度から 10-60 回程度が適切である [10]．

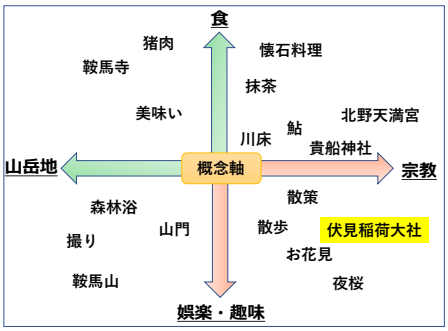

図 5　可視化イメージ

4.5　分散表現の MySQL 格納&解釈性評価

パラグラフベクトルを学習したツイートの中から特定の特徴単語（あるいはその組み合わせ）の重みを降順にソートし，上位 N 件のツイートを高速に抽出できるように各ツイートのパラグラフベクトルを関係データベース MySQL に格納する．MySQL には，Twitter ID, ツイートを代表する代表語，264 種類の特徴単語の重みを格納する．代表語としては，ツイートのパラグラフベクトルに最も距離が近いベクトルを持つ単語やアスキー文字を除く先頭単語が候補となる．データを格納するために表 4, 表 5, 表 6 に示す 3 種類の関係テーブルを設計した．これら 3 種類の関係テーブルを連結して，検索条件を満たすツイート ID を抽出する．

分散表現の解釈性評価は，ツイートにカテゴリを付与したベンチマークを作成し，特定のカテゴリに最も類似した特徴単語の重みが大きな上位 N 件のツイートの適合率，再現率により評価を行う．

4.6　可視化システム

任意の特徴単語（あるいはその組み合わせ）について重みの降順でツイートを表示すれば，特徴単語に関連した有用なツイートを得ることができることを検証する．ソーシャルメディアマイニングを目的に，図 5 のように，特徴単語（あるいはその組み合わせ）を概念軸として非構造化テキストを 2 次元面上に配置して，ツイートの可視化を行う．2 次元平面上には，x 軸，y 軸で指定された特徴単語の重みに応じて，代表語を配置する．代表語を選択した場合に Twitter ID を元にオリジナルツイートを取得して表示する．

5.　実　　験

5.1　ベンチマーク構築

クラウドソーシングを利用したベンチマーク構築フローを図 6 に示す．最初に京都観光に関するツイートについて以下の 2 種類の方法で収集を行った．

- 以下のキーワードを含むツイートを Twitter API を用いて自動収集を行った．

  京都 + (天気 OR 観光 OR 交通 OR 電車 OR バス) - (東京都 OR RT)

- クラウドソーシングを利用して，クラウドワーカーが京都観光に関するツイートを手動で収集を行った．

次に各ツイートに 10 名のクラウドワーカーを割り当て，京

表 7　Wikipedia コーパスと Twitter コーパスの統計情報

項目	値
パラグラフ数	1,206,703 件（Wikipeida:1,188,942 件，Twitter:17,761 件）
対象単語数	553,208 語 (10 パラグラフ以上出現)
基本単語数	15,866 語 (10 パラグラフ以上出現)
ダウンサンプル対象高頻度語数	3312 語 (高頻度語削減閾値:1e-05)

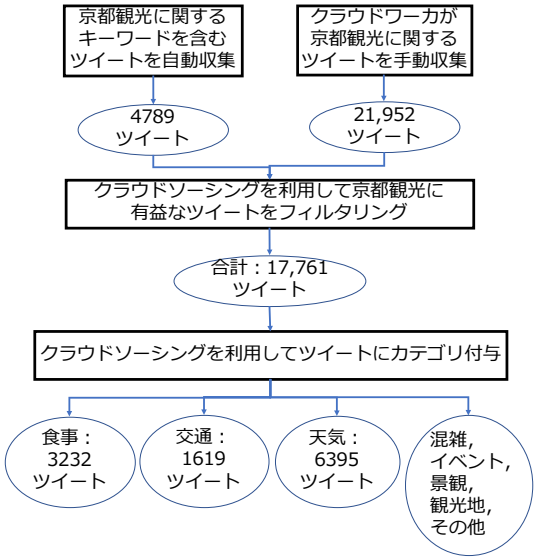

図 6　クラウドソーシングを利用したベンチマーク構築フロー

表 8　単語ベクトル学習とパラグラフベクトル学習のパラメータ

項目	値
単語ベクトル学習回数	3
パラグラフベクトル学習回数	50
学習係数（最小値）	0.025(0.0001)
ウインド長	5
負例サンプリングの単語数	5

都観光に関する場所の様子が分かる 1 年以内のツイートかどうかで 5 名以上が合意したツイートを有益なツイートとして採用した．手動収集の約 84%，自動収集の約 11%が採用され，合計 17,761 ツイートがベンチマークとして採用された．また，京都観光に関する場所の様子が分かる 1 年以内のツイートと判断した場合は，8 種類のカテゴリ（食事，交通，天気，混雑，イベント，景観，観光地，その他）の中から，適切なカテゴリを複数可で付与してもらった．ベンチマークとして採用した 17,761 ツイートに対して，一人でも付与したカテゴリの正解ツイートとした．「その他」を除く，7 種類のカテゴリのうち，特徴単語に直接対応するカテゴリは，「食事（特徴単語：食）」，「交通（特徴単語：交通 · 輸送）」，「天気（特徴単語：気象 · 気候）」の 3 種類である．これらの正解ツイート数は，「食事」3232 ツイート，「交通」1619 ツイート，「天気」6395 ツイートである．

クラウドワーカーによる京都観光に有益なツイートかどうかの判断，およびカテゴリ付与にはツイートに添付された写真が重要な役割を果たしたツイートも多い．本提案手法における分散表現の対象はテキストに限定されるが，実利用を考慮して，判断対象のテキストへの制約は付けなかった．

5.2　語彙辞書作成

日本語 Wikipedia コーパス (注2) からは，全てのタグを削除し，1 記事 1 行とした．Wikipedia コーパスや前節で作成した Twitter コーパスから単語を抽出するための日本語形態素解析には MeCab(注3) 及び Web 上の言語資源から数百万の新語や固有表現を拡充した MeCab 用辞書 mecab-ipadic-NEologd(注4) を用いた．また，語彙辞書の構築，単語ベクトルの学習，パラグラフベクトルの学習には，gensim の doc2vec ライブラリ (注5) を用いた．Wikipedia コーパスと Twitter コーパスによるパラグラフの統計情報を表 7 に示す．

5.3　単語ベクトル学習とツイートの特徴抽出（パラグラフベクトル学習）

単語意味ベクトル辞書を元に，4.2 節で述べた手法により，語彙辞書に含まれる基本単語 15,866 語のシードベクトルを作成した．シードベクトルには，入力層と隠れ層の間の初期重みである入力ベクトルと隠れ層と出力層との間の初期重みである出力ベクトルの 2 種類ある．このシードベクトルを元に Wikipedia コーパスを対象に Skip-gram モデルを用いて単語ベクトルを学習させ，学習した単語ベクトルをシードベクトルとして PV-DBOW モデルを利用して Twitter コーパスのパラグラフベクトルを学習させた．両方のモデルのパラメータの違いは学習回数（Epoch 数）のみである．他のパラメータは同じものを用いた．学習に用いたパラメータを表 8 に示す．

5.4　解釈性評価

17,761 件のツイートを特徴単語「食」，「交通 · 輸送」，「気象 · 気候」の各重みの降順にソートし，カテゴリ「食事」，「交通」，「天気」の正解ツイートの再現率が 1%，10%，20%，30%，40%のときの適合率，および 5 ポイント平均適合率を表 9 に示す．再現率 10%段階（食事ツイートは上位 323 件，交通ツイートは上位 161 件，天気ツイートは上位 639 件）では，いずれの特徴単語も適合率 90%以上と高い．特徴単語「食」の再現率 10%段階のエラーは 4 件だが，これらは「食」に関連した単語を含んでおり，人との感覚の違いによるものである．

5.5　可視化実験

可視化はサーバー側で PHP スクリプトを利用して MySQL の 3 種類の関係テーブルを連結し，可視化対象のツイートを抽出して，動的に HTML を生成する．ツイートの可視化の例を図 7〜図 10 に示す．図 7 は，特徴単語「食」の重みを y 軸，特徴単語「気象 · 気候」の重みを x 軸とし，それぞれの重みの降

(注2)：https://dumps.wikimedia.org/jawiki/latest/
(注3)：http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html
(注4)：https://github.com/neologd/mecab-ipadic-neologd
(注5)：https://radimrehurek.com/gensim/models/doc2vec.html

表 9　各カテゴリ正解ツイートの再現率における対応する特徴単語の重みを降順ソートした時のツイートの適合率

	適合率（チャンスレート）		
再現率	カテゴリ:食事－特徴単語:食	カテゴリ:交通－特徴単語:交通・輸送	カテゴリ:天気－特徴単語:気象・気候
1%	100% (18.2%)	100% (9.1%)	95.5% (36.0%)
10%	98.8% (18.2%)	90.0% (9.1%)	92.0% (36.0%)
20%	97.9% (18.2%)	77.7% (9.1%)	83.3% (36.0%)
30%	95.3% (18.2%)	71.2% (9.1%)	73.4% (36.0%)
40%	92.7% (18.2%)	63.1% (9.1%)	63.2% (36.0%)
5 ポイント平均	96.9% (18.2%)	80.4% (9.1%)	81.5% (36.0%)

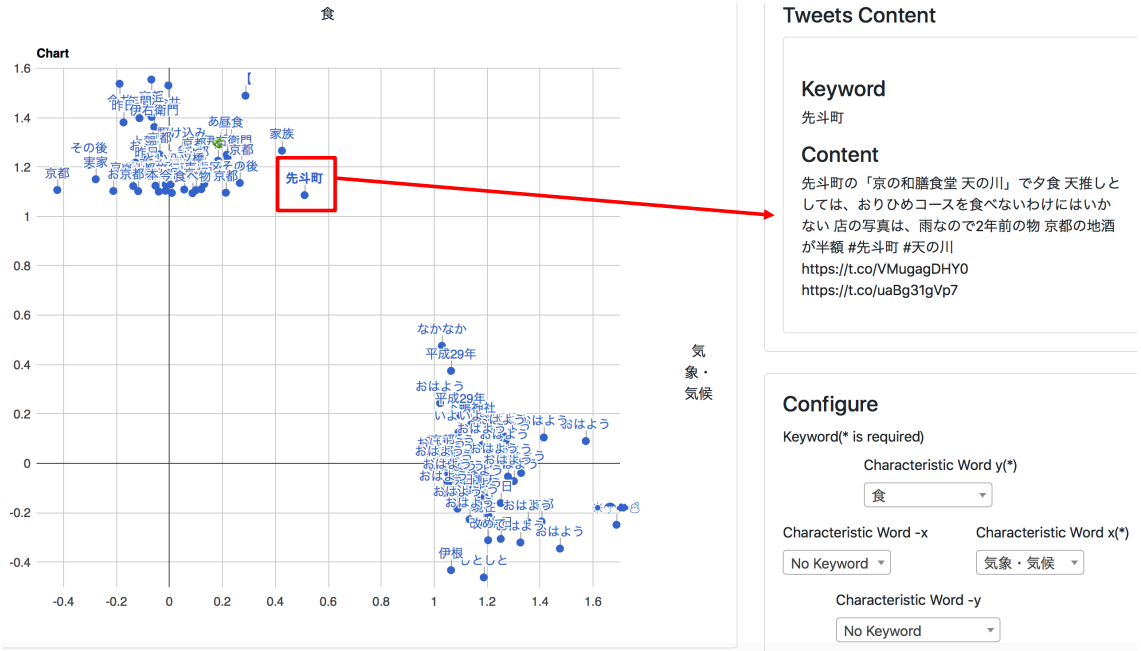

図 7　可視化例 1（y 軸：特徴単語「食」上位 50 ツイート，x 軸：特徴単語「気象・気候」上位 50 ツイート，代表語：ツイートの先頭単語）

順に上位 50 ツイートを抽出して xy 平面に各ツイートの代表語を配置したものである．食事カテゴリのクラスタと天気カテゴリのクラスタが明確に分離していることが分かる．食事カテゴリのクラスタの中で最も特徴単語「気象・気候」の重みが大きいツイート（代表語「先斗町」）を選択した．右端の Content にオリジナルツイートが表示されている．食事に関するツイートであり，天気が雨であることが分かる．TwitterAPI(注6)を用いたオリジナルツイートの取得は，JavaScript ライブラリの jQuery(注7)を利用して非同期通信により実現した．

図 8 では，x 軸を特徴単語「素材・材料」の重みに変更した．両方のクラスタが重なっていることが分かる．食事カテゴリのクラスタの中で最も特徴単語「素材・材料」の重みが大きいツイート（代表語「今井」）を選択した．Content のオリジナルツイートを見ると，食事に関するツイートであり，食事の素材を説明しているツイートであることが分かる．

前述の 2 種類の可視化例では，代表語をツイートの先頭単語（アスキー文字を除く）とした場合である．特徴単語「気象・気候」の重みが大きいツイートでは「おはよう」が多く，特徴単語「食」や「気象・気候」の重みが大きいツイートでは「京都」のような地名が多く，ツイート選択の具体性に欠ける．そこで，同じツイートの抽出条件で，代表語をツイートとベクトルのコサイン類似度が最も 1 に近い単語とした場合の可視化例を図 9 と図 10 に示す．「おはよう」や「京都」のような地名は代表語から無くなり，より具体性がありツイートの内容を表した単語（「雨具」，「おぼろ豆腐」など）が選択されていることが分かる．

2 次元の可視化の課題は，ツイートが密集しているところの代表語はズーミングをしないと読めないことだが，ツイート間のユークリッド距離が閾値以下の代表語を表示から間引くことにより，改善することが出来る．

6. おわりに

本稿では，ニューラルネットワークと単語意味ベクトル辞書を融合した意味表現学習によって獲得されたツイート分散表現の解釈性について，クラウドソーシングを利用して構築したカテゴリ分類ベンチマークを用いて定量的評価を行った．本ベンチマークは，京都観光に有益な 17,761 ツイートを抽出し，各

(注6)：https://apps.twitter.com/
(注7)：http://jquery.com/

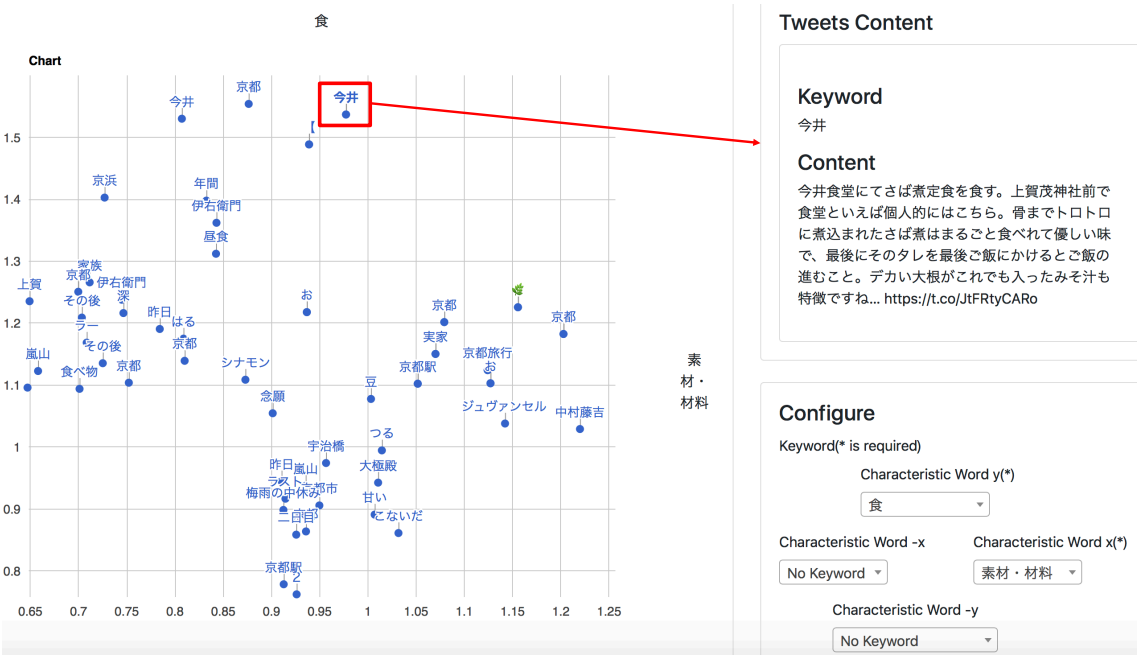

図 8　可視化例 2（y 軸：特徴単語「食」上位 50 ツイート，x 軸：特徴単語「素材・材料」上位 50 ツイート，代表語：ツイートの先頭単語）

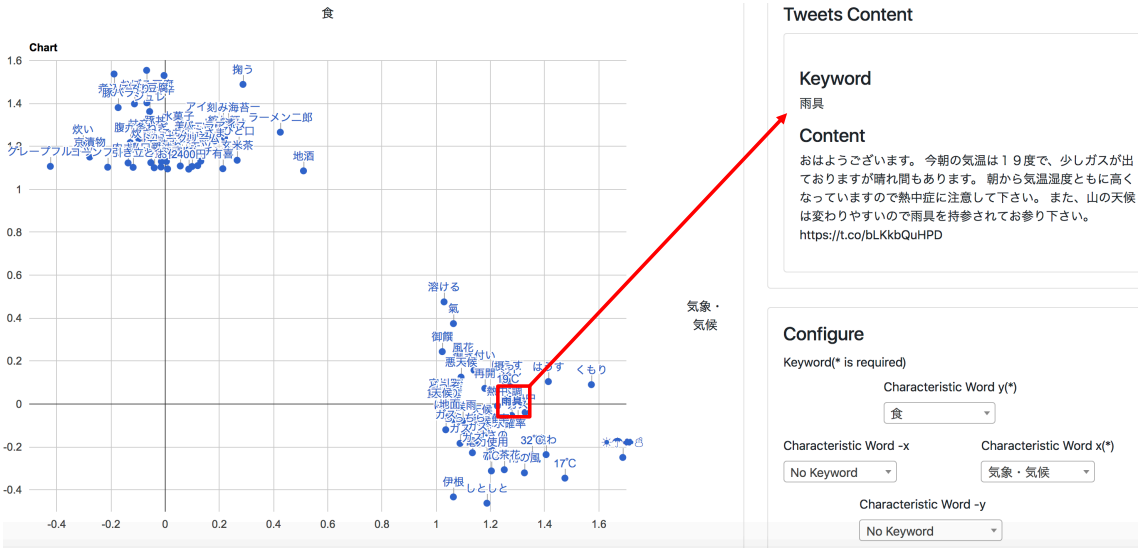

図 9　可視化例 3（y 軸：特徴単語「食」上位 50 ツイート，x 軸：特徴単語「気象・気候」上位 50 ツイート，代表語：ツイートとベクトルのコサイン類似度が最も 1 に近い単語）

ツイートに 8 種類のカテゴリ (食事，交通，天気，混雑，イベント，景観，観光地，その他) を付与したものである．カテゴリに対応する特徴単語が存在する「食事」,「交通」,「天気」カテゴリに関しては，これらのカテゴリに対応する特徴単語の重みの降順にツイートをランキングすることにより，再現率 10%（上位 161〜639）段階で，90.0%〜98.8%の適合率を示した．ベンチマークのツイート本文は単語 1 語のように添付された画像を見ないとカテゴリを類推できないツイートも多いが，特定の特徴単語の重みが大きくなるツイートには多くの単語が含まれており，再現率 10%段階では高い適合率を示したと考えられる．この特徴は場所やイベントに対するソーシャルセンサーとしてのツイートのリアルタイムフィルタリングに適したものである．

今後の課題は，トピックモデル（LDA）[17] を用いたカテゴリ分類との優位性比較，および特徴単語が存在しないカテゴリ（混雑，イベント，景観，観光地）について決定木などを用いて特徴単語の組み合わせによるカテゴリ表現手法の確立である．さらにエンコーダ・デコーダモデルを用いたカテゴリ予測（教師あり深層学習）[18] との優位性比較と融合方式を検討する．

また本稿では，解釈性の高い分散表現学習の有効なアプリケーションとして，非構造化テキストの可視化手法を提案した．カテゴリに対応する特徴単語を概念軸として x-y 平面にツイートを可視化した場合は，カテゴリ分析が可能となることを確認した．代表語の選択方法として，先頭語の選択，およびツイートと類似度が最も高い単語の選択の 2 種類の方法を試し

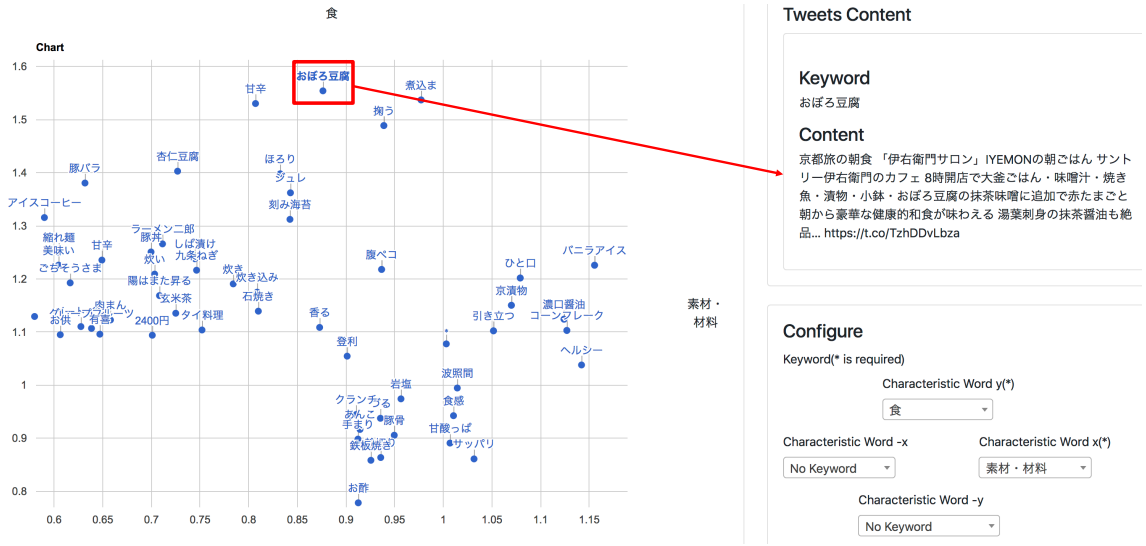

図 10　可視化例 4（y 軸：特徴単語「食」上位 50 ツイート，x 軸：特徴単語「素材 · 材料」上位 50 ツイート，代表語：ツイートとベクトルのコサイン類似度が最も 1 に近い単語）

た．今後の課題は，代表語の選択方法の改良と主観評価，および複数の特徴単語の組み合わせにより概念軸を表現する手法を MySQL と PHP スクリプトを用いて実現することである．

本研究は，ニューラルネットワークや深層学習のブラックボックス化の課題を人の知見（記号）との融合により，解決することを目指している．これにより，機械学習の利用において最も負荷が大きな教師データの作成を必要とせずに非構造化テキストのカテゴリ分析の可能性を示した．今後は，非構造化テキストのみならず添付画像に対しての 264 種類の特徴単語による分散表現学習の可能性を検討すると共にケーススタディにより実際に価値（ユーザ便益や収益など）が創出できるのか評価を進める．

謝　　辞

本研究成果の一部は，国立研究開発法人情報通信研究機構の委託研究，および NAIST ビッグデータプロジェクトの助成により得られたものです．

文　　献

[1] 榊 剛史, 松尾 豊, “ソーシャルセンサとしての Twitter ーソーシャルセンサは物理センサを凌駕するか？ー,” 人工知能学会誌, VOL. 27, No. 1, pp. 67-74, 2012.

[2] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” Proc. of Workshop at ICLR, 2013.

[3] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” Proc. of NIPS, pp.3111-3119, 2013.

[4] T. Mikolov, W. Yih, and G. Zweig, “Linguistic Regularities in Continuous Space Word Representations,” Proc. of NAACL HLT, pp.746-751, 2013.

[5] Q. Le, T. Mikolov, “Distributed Representations of Sentences and Documents,” Proc. of ICML, pp.1188-1196, 2014.

[6] 芥子 育雄, 池内 洋, 黒武者 健一, “百科事典の知識に基づく画像の連想検索,” 電子情報通信学会論文誌, Vol. J79-D-II, No. 4, pp. 484–491, 1996.

[7] 芥子 育雄, 鈴木 優, 吉野 幸一郎, グラム ニュービッグ, 大原 一人, 向井 理朗, 中村 哲, “単語意味ベクトル辞書を用いた Twitter からの評判情報抽出,” 電子情報通信学会論文誌, Vol. J100-D, No. 4, pp. 530–543, 2017.

[8] 芥子 育雄, 鈴木 優, 吉野 幸一郎, 大原 一人, 向井 理朗, 中村 哲, “分散的意味表現学習のための単語意味ベクトル辞書 Ver.2 と日本語 Twitter 極性分析ベンチマークについて,” 情報処理学会研究報告, Vol. 2017-NL-231, Vol. 2017-SLP-116, No. 8, pp. 1-7, 2017.

[9] I. Keshi, Y. Suzuki, K. Yoshino, and S. Nakamura, “Semantically Readable Distributed Representation Learning for Social Media Mining,” Proc. of IEEE/WIC/ACM International Conference on Web Intelligence (WI), pp. 716–722, 2017.

[10] I. Keshi, Y. Suzuki, K. Yoshino, and S. Nakamura, “Semantically Readable Distributed Representation Learning and Its Expandability Using a Word Semantic Vector Dictionary,” IEICE Transactions on Information and Systems, Vol. E101-D, No.4, April, 2018. DOI:10.1587/transinf.2017DAP0019

[11] B. Murphy, P. Talukdar, and T. Mitchell, “Learning Effective and Interpretable Semantic Models using Non-Negative Sparse Embedding,” In Proc. of COLING, 2012.

[12] H. Luo, Z. Liu, H. Luan, and M. Sun, “Online Learning of Interpretable Word Embeddings,” In Proc. of EMNLP, pp. 1687–1692, 2015.

[13] F. Sun, J. Guo, Y. Lan, J. Xu, and X. Cheng, “Sparse Word Embeddings Using l_1 Regularized Online Learning,” In Proc. of IJCAI, pp. 2915–2921, 2016.

[14] S. Suzuki, “Probabilistic Word Vector and Similarity Based on Dictionaries,” Proc. of CICLing, pp. 564–574, 2003.

[15] M. Faruqui, J. Dodge, S.K Jauhar, C. Dyer, E. Hovy, and N. A. Smith, “Retrofitting Word Vectors to Semantic Lexicons,” In Proc. of NAACL-HLT, pp. 1606–1615, 2015.

[16] Y. Sakaizawa and M. Komachi, “Construction of a Japanese Word Similarity Dataset,” CoRR, abs/1703.05916, 2017.

[17] D.M. Blei, A.Y. Ng, and M.I. Jordan, “Latent dirichlet allocation,” J. Mach. Learn. Res., vol.3, pp.993–1022, 2003.

[18] B. Dhingra, Z. Zhou, D. Fitzpatrick, M. Muehl, and W. Cohen, “Tweet2Vec: Character-Based Distributed Representations for Social Media,” In Proc. of ACL, vol.2, pp. 269–274, 2016.