ウェブ上の言語資源を用いた単語のベクトル表現の翻訳

石渡 祥之佑[1,a)]　鍛治 伸裕[2,3,b)]　吉永 直樹[2,3,c)]　豊田 正史[2,d)]　喜連川 優[2,4,e)]

概要：言語処理では，単語間の意味を比較するために，それらの共起語のベクトルを用いて語の意味を表現する．しかし，そうした共起語に基づくベクトル表現では，異なる言語の単語対の意味を比較することはできない．そのため，言語横断検索や対訳辞書の自動構築といった多言語処理には利用できない．そこで本稿では，ある言語におけるベクトル表現を他の言語のベクトル表現に写像することで，言語の壁を越えて透過的に意味を扱うことを目指す．提案手法では，既存手法と同様に，写像に用いる翻訳行列の学習を最適化問題として定式化する．その際，ウェブ上の言語資源を利用して目的関数を補正することで，より正確な翻訳行列の学習を可能にする．実験では日本語から英語へのベクトル表現の翻訳を行い，提案手法の有用性を評価する．

1. はじめに

単語の意味を計算処理可能な形式で表現することは，情報検索やウェブマイニングなど，自然言語テキストを対象としたアプリケーションの高度化に必要となる要素技術であり，これまでに多くの研究が行われてきた [1][2][3]．現在，単語の意味表現として広く用いられているのは，「ある単語の意味は，その単語と共に出現している単語群によって特徴づけられる」という分布仮説 [4] に基づくものである．すなわち，ある単語の意味を，コーパスにおける共起語の頻度に基づくベクトルとして表現するという方法が一般的となっている．単語をそうしたベクトル形式で表現することによって，単語間の意味的な類似性の計算を，ベクトル間の類似度計算 (コサイン類似度など) に帰着させることが可能になる．

しかし，従来の意味表現は，異なる言語の単語間の類似性判定，すなわち単語の対訳関係の判定に用いることが難しい．なぜなら，異なる言語のコーパスから上記のようなベクトル表現を学習した場合，共通の共起語がほとんど出現しないためである．このため，言語横断検索や対訳辞書の自動拡張など，多言語処理アプリケーションにおいて利用することができない．このような問題意識から，近年では，単語のベクトル表現を別の言語に翻訳する (例: 日本語 “犬” のベクトル表現を英単語 *dog* のベクトル表現に変換する) ための方法に関する研究が進められている [5]．

本研究では，単語のベクトル表現の翻訳において，ウェブ上の対訳資源を活用する新しい方法を提案する．分布仮説に基づいて単語をベクトルで表現した場合，ベクトルの各次元は共起単語に対応していることになる．そのため，対訳辞書などの言語資源を使えば，ある言語のコーパスから学習したベクトル表現のどの次元が，別言語のコーパスから学習したベクトル表現のどの次元に対応しているかに関して，部分的にではあるが，情報を得ることができる．そこで，ベクトル間の翻訳行列を学習する際，対訳辞書から得られる情報を報酬項として利用することにより，より精度の高い翻訳を実現させる．

提案手法の有効性を検証するため，ベクトル表現の翻訳によって訳語選択を行うというタスクを設定し，その精度を評価した．現在までに，対訳辞書の利用による精度の向上を確認することができたため，その結果について報告を行う．また，今後の改良点に関しても議論を行う．

以降，2 節では単語の意味をベクトルで表現するための方法を述べる．3 節ではある言語の単語のベクトルを他の言語のベクトルに翻訳する手法を述べる．続く 4 節で翻訳の精度を評価するための実験について述べ，5 節，6 節，7 節ではそれぞれ関連研究，まとめ，今後の課題を述べる．

1 東京大学大学院 情報理工学系研究科
Graduate School of Information Science and Technology, The University of Tokyo
2 東京大学 生産技術研究所
Institute of Industrial Science, The University of Tokyo
3 独立行政法人 情報通信研究機構
National Institute of Information and Communications Technology
4 国立情報学研究所
National Institute of Informatics
a) ishiwatari@tkl.iis.u-tokyo.ac.jp
b) kaji@tkl.iis.u-tokyo.ac.jp
c) ynaga@tkl.iis.u-tokyo.ac.jp
d) toyoda@tkl.iis.u-tokyo.ac.jp
e) kitsure@tkl.iis.u-tokyo.ac.jp

2. 単語のベクトル表現

本節では，単語の意味をベクトルで表現するための方法について説明を行う．ここで基本となるのは，「ある単語の意味は，その単語と共に出現している単語群によって特徴づけられる」[4][6] という考え方である．この考えによれば，単語の意味は共起単語の分布のベクトルで表現できる．

例として，以下のような文 [7] で構成されるコーパスが与えられたとき，これを用いて *valley* という単語の意味をベクトルで表現することを考える．

> *I have a dream that one day every valley shall be exalted.*

単語 *valley* のベクトル表現は，前後に出現している単語の出現頻度に基づいて構成される．このとき，*valley* の前後何語まで考慮するかを定める必要があるが，ここでは前後 3 語を考えるものとする．そうすると，*valley* の周囲には *one*，*day*，*every*，*shall*，*be*，*exalted* が 1 度ずつ出現していることになる．そのため，単純な共起頻度を使って得られる *valley* のベクトルは各次元をそれぞれ共起語 *I*，*have*，*a*，*dream*，*that*，*one*，*day*，*every*，*valley*，*shall*，*be*，*exalted* に対応させることで，以下のようになる．

$$(0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 1, 1)$$

例えば，上記のベクトルの 6 番目の要素の値が 1 であるが，これは *valley* が *one* と 1 回共起したことを表している．

この例では，簡単のため共起頻度の値をそのまま各次元の値としているが，一般的には，式 (1) に示す正の自己相互情報量 (PPMI)[8][9] など共起度を表す尺度に変換したものが用いられる．

$$ppmi(w, f) = \max\left(\log_2 \frac{P(w, f)}{P(w)P(f)}, 0\right) \qquad (1)$$

ただし，ここで $P(w)$ は見出し語 w の出現確率，$P(f)$ はベクトルのある次元に対応する語 f の出現確率，$P(w, f)$ は w と f がコーパス内で同時に出現する確率をそれぞれ表す．

上記の例では，ある単語の前後 3 語をベクトルの要素として利用したが，このように対象の前後何語までを共起しているとみなすかを文脈窓と呼ぶ．文脈窓の長さは，短くするほど結果のベクトルが単語の統語的な性質を捉え，逆に長くするほど，単語のトピックの性質を捉えることが知られている [10]．

3. 提案手法

前節で述べたように，ある単語のベクトルは同じ言語のテキストコーパスから作られ，ベクトルの各次元はその言語における他の単語となっている．そのため，異なる言語に属する 2 単語のベクトルをそのまま比較することはできない．そこで本節ではこの問題に対処するため，ある言語における単語のベクトルを他の言語におけるベクトルに翻訳する方法を提案する．

3.1 ベクトル表現の翻訳

まずはじめに，提案手法の基礎となる Mikolov らの手法 [5] について説明する．

Mikolov らは行列変換に基づくベクトル表現の翻訳手法を提案している．これは，ある言語の単語 $\boldsymbol{x}$[*1] に翻訳行列 $\boldsymbol{W}$ を乗じることによって，その訳語のベクトル表現の近似を得るというものである．翻訳行列 $\boldsymbol{W}$ は，訳語ペアの集合 $\{\boldsymbol{x_i}, \boldsymbol{z_i}\}_{i=1}^{n}$ を用いて，以下の最適化問題を解くことによって得られる．

$$\boldsymbol{W}^{\star} = \arg\min_{\boldsymbol{W}} \sum_{i=1}^{n} ||\boldsymbol{W}\boldsymbol{x}_i - \boldsymbol{z_i}||^2 + \frac{\lambda}{2}||\boldsymbol{W}||^2 \qquad (2)$$

ここで 2 つめの項は正則化項である．この正則化項は，Mikolov らの元論文では用いられていない．しかし，正則化項には過学習を防ぐ効果があり，一般的にモデル学習の精度を高めるために有効であることから，本論文では上記のような学習方法に基づいて議論を進める．

3.2 報酬項の導入

前節で述べた翻訳行列の学習に用いるある言語のベクトルと他の言語のベクトルの次元は，互いに異なるコーパスにおける共起単語に対応している．この 2 つの言語それぞれのベクトルは，たとえ異なる言語と言えども，対訳や翻字，発音の類似など，何らかの関係性が存在する可能性がある．本来，こうした次元同士の関係性は翻訳の際の有用な手がかりとして用いることができるはずだが，前節の式 (2) ではこうした情報を考慮できていない．そこで，本研究では異なる 2 つの言語のベクトルの次元を恣意的に対応付ける方法を提案する．

続いて，提案手法の基本的なアイデアを述べる．例として，$\boldsymbol{x}$ が次元数 M の日本語のベクトル，$\boldsymbol{z}$ が次元数 N の英語のベクトルだとすると，$\boldsymbol{W}$ は N 行 M 列の行列となる．仮に $\boldsymbol{x}$ の p 次元目が「犬」という語との共起頻度を表し，$\boldsymbol{z}$ の q 次元目が *dog* という語との共起頻度を表すとする．「犬」と *dog* は明らかに正しい対訳ペアなので，$\boldsymbol{W}$ の q 行 p 列目の値は相対的に大きくなるべきである．

このように対訳関係になっている次元のペアにかかる重みを強くするため，式 (2) に報酬項を加えた下記の最適化問題を解くことによって，翻訳行列を学習することを提案する．

[*1] 正確には単語ではなく単語のベクトル表現であるが，文脈から明らかな場合には，単語とそのベクトル表現のことは区別しないものとする．

$$\boldsymbol{W}^{\star} = \arg\min_{\boldsymbol{W}} \sum_{i=1}^{n} ||\boldsymbol{W}\boldsymbol{x_i} - \boldsymbol{z_i}||^2 + \frac{\lambda}{2}||\boldsymbol{W}||^2 - \beta \sum_{(p,q)\in dict} w_{pq} \tag{3}$$

ただし $dict$ は対訳関係にある次元の組の集合である．係数 β (> 0) は報酬項の強さを制御するパラメータであり，これを大きくするほど，対訳ペア (p, q) に対応する $\boldsymbol{W}$ の q 行 p 列目の成分が大きくなることが期待できる．

本研究では，上式における対訳関係にある次元の組の集合 $dict$ はウェブ上の言語資源から得られることに着目した．今回は特に，Web から利用可能な英-日の対訳辞書である EDICT[*2] を用いて実験を行う．ただし，多義語については EDICT における 1 つめの訳語のみを使用し，日本語と英語が 1:1 に対応する形式にして用いた．

3.3 確率的勾配降下法の適用

前節の最適化問題を確率的勾配降下法で解く．このとき，τ 番目の学習サンプル $(\boldsymbol{x}, \boldsymbol{z})$ が与えられたとき，翻訳行列 $\boldsymbol{W}$ は以下のように更新される．

$$\boldsymbol{W} \leftarrow \boldsymbol{W} - \eta_\tau \nabla E_\tau(\boldsymbol{W}) \tag{4}$$

ここで η_τ は学習率である．本研究ではオンライン凸最適化アルゴリズムである Pegasos [11] に基づき，$\eta_\tau = \frac{1}{\lambda\tau}$ とする．また，$\nabla E_\tau(\boldsymbol{W})$ は，τ 番目の学習サンプル $(\boldsymbol{x}, \boldsymbol{z})$ に基づいて求められた勾配であり，以下の式で表すことができる．

$$\nabla E_\tau(\boldsymbol{W}) = 2(\boldsymbol{W}\boldsymbol{x} - \boldsymbol{z})\boldsymbol{x}^{\mathrm{T}} + \lambda\boldsymbol{W} - \beta\boldsymbol{D}$$

ただし $\boldsymbol{D}$ は，ウェブ上の対訳資源から得られた対訳ペアに対応する次元の組 (p, q) に対しては $D_{pq} = 1$，それ以外の (p, q) に対しては $D_{pq} = 0$ の行列である．

4. 実験

本節では，日本語の単語のベクトル表現を英語のベクトル表現に翻訳することによって，適切な訳語選択が可能かどうかを評価する実験について述べる．その結果により，提案手法の有効性を検証する．

4.1 データセット

データセットとして，翻訳に用いるための日本語のベクトル，翻訳の正解データとしての英語のベクトル，そして日英のベクトルの訓練データを作成するための日英間の単語対応表の 3 つを用意する．

まず，英語の単語のベクトルとして Turney ら [12] が作成したデータセットを用意した．Turney らはまず，ウェブページをクロールして得られたおよそ 280GB の英語のテキストデータのうち，Wordnet[*3] に出現している語を見出し語化し，コーパスとした．続いて彼らはこのコーパスから，文脈窓を 4 に設定して (すなわち，前後 4 単語以内に出現している語を共起語として) 単語のベクトルを作成した．この際，ベクトルの次元は全て Wordnet に存在する語から選ばれており，接続詞や助詞と言った機能語は含まれない．また，このデータセットには *big tree* や *at once* といった短いフレーズも含まれており，これらも 1 単語として扱われている．

続いて，日本語の単語のベクトルを作成するため，2012 年 1 月 1 日〜2012 年 12 月 31 日の期間，Twitter[*4] からクロールして得られた日本語の投稿およそ 24GB を得た．Kaji ら [13] の提案した形態素解析ソフトウェアを用いて全ての投稿を単語に分割した後，これらを見出し語化してコーパスとし，単語のベクトルを作成した．この際，Turney らの作成した英語のベクトルでは文脈窓を 4 としていたことを参考にしつつ，日本語では助詞や助動詞といった機能語が多用されることを考慮し，日本語の単語ベクトルでは文脈窓を 5 に設定した．英語の場合と同様に，作成されたベクトルの次元は全て日本語 Wordnet[*5] に存在する語から選ばれており，接続詞や助詞といった機能語は含まれない．

こうして作られた日本語と英語のベクトルは，いずれも非常に高次元である．このままでは翻訳行列 W の学習は計算資源を大量に消費してしまうため，日英ともに頻出の次元 1000 次元のみを残した．また，単語の出現頻度の差異を吸収するため，全てのベクトル各次元の値を式 (1) に基いて PPMI に変換した後，正規化を行ってスケールを統一した．

日英間の単語対応表は，日本語 Wordnet から得られた日英間の単語翻訳ペア 230,653 対 [*6] を使用した．

実験では以上の日本語ベクトル，英語ベクトル，単語対応表の 3 つのデータセット全てに含まれている語 (日本語，英語それぞれ 20,473 語) のみを残し，残りを破棄した．こうして得られた日本語，英語の各ベクトルは，上記で述べた日英間の単語対応表で対応を取り，ペアとして扱えるようにした．この 20,473 ペアのうち 5000 ペアをテストセット，2000 ペアを開発セット，残りの 15,083 ペアを訓練セットとして使用した．

4.2 評価手法

3.1 節で述べたように，学習した翻訳行列がどれほど正確に言語間の翻訳を行うことができるかは，ある単語のベクトル $\boldsymbol{x}$ を翻訳した結果 $\boldsymbol{W}\boldsymbol{x}$ が，どれほど対訳語のベクトル $\boldsymbol{z}$ に近くなるかで評価すればよい．そこで，表 1 のような日本語から英語の対訳を選択する 4 択問題を用いて評

[*2] http://www.edrdg.org/jmdict/edict.html
[*3] http://wordnet.princeton.edu/
[*4] https://twitter.com/
[*5] http://nlpwww.nict.go.jp/wn-ja/
[*6] この際，多義語の場合でも 1 つめの訳語のみを使用し，2 つめ以降の訳語は破棄した．

表 1 テストデータの 1 例

日本語:		「牛飼い」
翻訳候補:	(1)	cattleman
	(2)	course of action
	(3)	remaining
	(4)	big tree
正解:	(1)	cattleman

価する．ある日本語の単語と，その翻訳候補として 4 つの英単語が与えられる．翻訳候補 (1)〜(4) のうち，正しい選択肢は「牛飼い」と同義の (1) *cattleman* であり，(2)〜(4) はそれ以外のランダムな英単語を用いて作成した誤った選択肢である．ここで与えられる正解の対訳ペアは 4.1 節末尾で述べた日本語，英語のベクトル 5000 ペアの中からランダムに選ばれた 1,250 ペアであり，誤った翻訳候補はそれ以外の 3,750 語からランダムに選び，構成される．

翻訳行列を用いて変換した日本語のベクトル x と，英語の候補 c のベクトル z_c との距離 E_c は式 (5) で算出する．式 (5) により求めた E_1 〜 E_4 のうち最小となる場合の c を翻訳候補の中から選択する．

$$E_c = ||\boldsymbol{W}\boldsymbol{x} - \boldsymbol{z_c}||^2 \tag{5}$$

表 1 のような評価セットをパラメータ調整用に 500 題，テスト用に 1,250 題用意した．評価では表 1 の (1) を選ぶことができた確率 (正答率) を確認する．以下 4.3 節〜4.4 節では，それぞれ異なる条件でこの評価を行った．

4.3 2 つの比較手法によるベクトルの翻訳

3.2 節で述べた提案手法の性能を把握するため，以下に述べる 2 つの比較手法を実装した．

比較手法 1

翻訳行列による翻訳が有用であることを示すため，翻訳行列を用いない手法を実装した．この手法では，ウェブ上から得られた日本語と英語の対訳辞書のみをベクトルの翻訳に用いる．対訳関係にあるペア同士は同じ意味を表しているため，互いに対応付けることで日本語のベクトルから英語のベクトルへの翻訳を行うことができる．具体的な手順は下記に従う．まず 4.1 節で述べた日本語のベクトルの 1000 次元それぞれに対応する 1000 語と，英語のベクトルの 1000 次元それぞれに対応する 1000 語の組み合わせのうち，EDICT[*7] から日英対訳ペアを 231 対を得た．この対訳ペアで対応付けられた 231 次元のみを残し，日本語，英語ともに 231 次元のベクトルを得た．最後に日本語の各次元から英語の各次元へ 1:1 の変換を行うことで，翻訳後のベクトルを得た．

比較手法 2

報酬項が翻訳精度向上に寄与することを示すため，3.1 節で述べた報酬項を導入しない手法を実装した．まず式 (2) に基づき，15,083 対の日本語-英語ペアのベクトルを用いて，翻訳行列 $\boldsymbol{W}$ の学習を行った．こうして得られた $\boldsymbol{W}$ を日本語のベクトルに乗じることで，翻訳後のベクトルを得た．

[*7] http://www.edrdg.org/jmdict/edict.html

表 2 実験結果

手法:	λ	β	正答率 [%]
比較手法 1 (対訳辞書のみ用いた翻訳)	-	-	50.2
比較手法 2 (翻訳行列のみによる学習)	0.03	0.0	50.0
提案手法 (翻訳行列+対訳辞書)	0.04	0.05	55.4

4.4 実験結果

提案手法では，翻訳行列 $\boldsymbol{W}$ を学習する際，4.3 節で得られた 231 対の日英対訳辞書を用いた，この際，式 (3) に基づき，$\boldsymbol{W}$ の学習を行った．こうして得られた $\boldsymbol{W}$ を日本語のベクトルに乗じることで，翻訳後のベクトルを得た．

上記の提案手法及び 4.3 節で述べた 2 つの比較手法を用いて，4.2 節の実験を行った際の結果を表 2 に示す．正則化項の係数 λ と，報酬項の係数 β はともに開発セットを用いて最適な値を求めた．λ は 0.05 〜 10.0 の範囲から 0.01 刻みで最適なものを選択し，β は β は {0, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0, 5.0, 10.0} の中から最適なものを選択した．ただし，比較手法 2 では報酬項は存在しないため，β を 0.0 に固定して実験を行った．

4.3 節に述べた比較手法 1(対訳辞書によって直接ベクトルを変換する手法) を用いた場合では正答率 50.2%となっており，比較手法 2(報酬項無しで翻訳行列を学習する手法) とほぼ同程度の精度が得られた．提案手法 (報酬項を導入して翻訳行列を学習する手法) では，比較手法 2 と比較しておよそ 10%の精度向上が得られたことから，対訳関係にある次元の重みを調整することで，より良い翻訳行列 $\boldsymbol{W}$ を得ることができると言うことができる．

図 1，図 2 にそれぞれ日本語の単語のベクトルと英語の単語のベクトルの例をそれぞれ図示する．ただし，これらは全て主成分分析 (PCA) によって 1000 次元から 2 次元にベクトルの次元数を削減したものである．各図における各単語の相対位置が，その言語における単語同士の関係性を表している．単語のベクトルを翻訳することは，すなわち図 1 に示した日本語のベクトル空間から，図 2 に示した英語のベクトル空間への写像を行うことと同義である．図 2 には，図 1 における日本語の単語「牛飼い」のベクトルを英語のベクトルに翻訳した結果も載せている．ここからは，提案手法を用いて翻訳した場合の方が，比較手法 2 の手法を用いた場合よりも正解データである英語の *cattleman* に近いベクトルが得られていることが読み取れる．

表 3 に比較手法と提案手法で異なる出力結果が得られたサンプルを示す．日本語の各単語について，ベースラインと提案手法のそれぞれを用いた場合，式 (5) で定義した日

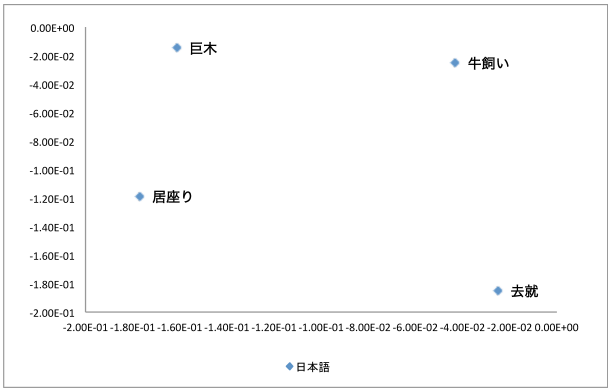

図 1　日本語のベクトル

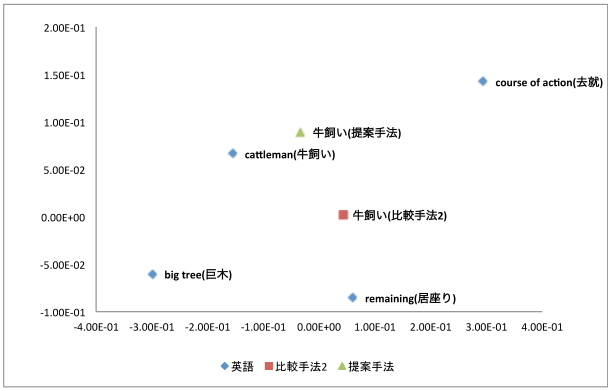

図 2　英語のベクトルと翻訳後のベクトル

本語とその翻訳候補 c の距離 E_c が小さい順に並べてある．表 3 では各問題で (1) が常に正しい対訳語であり，1 番目に (1) が来ていれば「日本語から英語へ正しく翻訳出来た」ことを表す．

4.5　考察

表 2 から，比較手法 1 と比較手法 2 はほぼ同程度の精度を達成していることがわかる．しかし，比較手法 1 では「ちょっかい」「陰暦」「露点」は翻訳できなかった．これは，元々スパースなベクトルの次元が 1000 次元から 231 次元に減ることで，ゼロベクトルとなってしまったことに起因する．日本語のゼロベクトルは翻訳しても英語のゼロベクトルとなってしまうので，他の英単語との類似度を計算することは不可能である．このように，比較手法 1 には辞書にまだ存在していない次元同士の対訳関係を全く用いることができないという限界が存在する．一方で，比較手法 1 と提案手法はいずれも 1000 次元全てを用いているため，ゼロベクトルになる可能性は比較手法 2 よりも少ない．

提案手法は比較手法 1 のデータ (対訳辞書) と，比較手法 2 のアイデア (翻訳行列の学習) を組み合わせた手法であると言える．以下では表 3 の結果を元に，比較手法 2 と提案手法の差異を議論する．

表 3 のうち，日本語「病理」や「ちょっかい」，「のけ者」などはいずれも比較手法 2 では正解候補 (1) が 2 位以降に

表 3　出力結果 (一部)

日本語:	「牛飼い」	
翻訳候補:	**(1) cattleman**	(2) course of action
	(3) remaining	(4) big tree
比較手法 1:	**(1)**, (4), (3), (2)	
比較手法 2:	(4), (3), (2), **(1)**	
提案手法:	(3), **(1)**, (4), (2)	
日本語:	「病理」	
翻訳候補:	**(1) pathology**	(2) part of speech
	(3) incessantly	(4) tachycardia
比較手法 1:	(2), **(1)**, (3), (4)	
比較手法 2:	(4), (2), (3), **(1)**	
提案手法:	**(1)**, (4), (2), (3)	
日本語:	「ちょっかい」	
翻訳候補:	**(1) meddle**	(2) haltingly
	(3) hairpiece	(4) complaining
比較手法 1:	N/A (設問がゼロベクトル)	
比較手法 2:	(2), (4), **(1)**, (3)	
提案手法:	**(1)**, (2), (4), (3)	
日本語:	「のけ者」	
翻訳候補:	**(1) outcast**	(2) unconcernedly
	(3) stripping	(4) mead
比較手法 1:	**(1)**, (4), (3), (2)	
比較手法 2:	(3), **(1)**, (2), (4)	
提案手法:	**(1)**, (3), (2), (4)	
日本語:	「陰暦」	
翻訳候補:	**(1) lunar calendar**	(2) secrecy
	(3) hermit	(4) destruction
比較手法 1:	N/A (設問がゼロベクトル)	
比較手法 2:	(3), **(1)**, (4), (2)	
提案手法:	**(1)**, (3), (4), (2)	
日本語:	「黒帯」	
翻訳候補:	**(1) black belt**	(2) spicebush
	(3) obsidian	(4) at once
比較手法 1:	(2), (3), (4), **(1)**	
比較手法 2:	(4), (3), (2), **(1)**	
提案手法:	(4), **(1)**, (3), (2)	
日本語:	「露点」	
翻訳候補:	**(1) dew point**	(2) leakage
	(3) experienced	(4) the universe
比較手法 1:	N/A (設問がゼロベクトル)	
比較手法 2:	(2), (4), (3), **(1)**	
提案手法:	(2), **(1)**, (3), (4)	
日本語:	「どんけつ」	
翻訳候補:	**(1) tail end**	(2) right on
	(3) head	(4) recrimination
比較手法 1:	(2), **(1)**, (3), (4)	
比較手法 2:	**(1)**, (4), (3), (2)	
提案手法:	(4), (3), **(1)**, (2)	

出力されているが，提案手法を用いた場合は 1 位に順位を挙げており，正しい翻訳として (1) を選ぶことができるようになったことがわかる．また，「牛飼い」や「黒帯」「露店」といった単語の場合では，提案手法を用いた場合でも (1) を選ぶことができていない．しかし，いずれも (1) の順位は向上しており，比較手法 2 よりは正解に近づいている．このことからも，提案手法の翻訳行列 $\boldsymbol{W}$ は比較手法 2 のものよりも正確な翻訳に寄与していることがわかる．比較手法 2 では正しく翻訳できていた単語が，提案手法ではうまく翻訳できなくなるサンプルも少ないながら存在した．単語「どんけつ」は比較手法 2 では正しく (1)*tail end* と翻訳できているが，提案手法では (4)*recrimination* が翻訳として選択されている．提案手法により翻訳精度が下がる理由は明らかでないが，今後詳細に分析を行う必要がある．

5. 関連研究

5.1 翻訳行列の学習

Mikolov ら [5] は Continuous Bag-of-Words (CBOW)[14] モデルを用いて学習した単語のベクトルを用いて翻訳行列を学習する手法を提案している．CBOW モデルで表された単語のベクトルは，各次元が明確に他の語との共起頻度を表していない．そのため，本研究の提案手法のように，次元同士の対訳関係といった知識を学習の際の報酬項として導入することはできない．そうした点では，本研究は Mikolov らの研究とは本質的に異なっている．

5.2 異言語に共通のベクトル表現の学習

ベクトル表現の翻訳とは異なるタスクであるが，Klementiev ら [15]，Zhang ら [16]，Hermann ら [17] は，異言語に共通のベクトル表現を学習する方法を提案している．特に Hermann ら [17] は「複数の言語から単語の表現を学習することで，言語非依存なもの (すなわち，意味) を精度よく捉えられる」と主張し，翻訳の必要ない言語非依存なベクトル表現を提案している．しかし，こうした単語の表現を学習するためには，莫大な対訳コーパスが必要となるため，対訳コーパスの存在しない言語対には適用できない．また，まだ対訳の出現していない新語にも適用できないという欠点も存在する．

6. まとめ

本研究では，単語のベクトル表現の翻訳において，ウェブ上の対訳資源を活用する新しい方法を提案した．単語を共起頻度のベクトルで表現した場合，ベクトルの各次元が共起単語に対応している．このことを利用し，ウェブ上から得られる対訳辞書を用いて，ある言語のコーパスから学習したベクトル表現の次元と，別言語のコーパスから学習したベクトル表現の次元に部分的な対応付けを行った．

提案手法では，こうした異言語における単語間の対応関係の情報を，ベクトル間の翻訳行列を学習する際の報酬項として利用することで，より精度の高い翻訳を実現した．日本語のベクトルから英語のベクトルへの翻訳を行った実験では，比較手法と比較しておよそ 10%程度，提案手法の翻訳精度が高いことを示した．

7. 今後の課題

7.1 関連研究との比較

単語のベクトル表現の翻訳に関しては，数は少ないものの，いくつかの関連研究が存在する．特に，5.1 節で述べた Mikolov らの研究は本研究と翻訳行列を学習するというアイデアは共通している．そのため，こうした手法とデータセット，コーパス等の条件を揃えた上で翻訳精度の比較を行う必要がある．また，5.2 節で述べた，異言語に共通のベクトル表現を学習する手法との詳しい比較も今後の課題としたい．

7.2 さまざまな言語知識の利用

本研究では日本語のベクトルから英語のベクトルへの翻訳を行っており，翻訳行列 $\boldsymbol{W}$ の調整にはウェブから獲得した対訳辞書のみを用いている．しかし，対象とする言語対によっては対訳辞書以外にも翻訳に用いることができる言語特性があると考えている．下記に翻訳に利用できるであろう言語特性と，実際の言語対における対訳例を挙げる．

表記の類似

例として，*university*(英語) と *universidad*(スペイン語) はいずれも「大学」という意味を表す．また，中国語と日本語における漢字など，多くの文字を共有している言語対も存在する．こうした言語対は異なる言語であるにも関わらず，単語レベルでは綴りが類似している場合が少なくない．こうした表記の類似，すなわち「編集距離の近さ」の情報を翻訳行列 $\boldsymbol{W}$ の調整に使用することで，表層が類似している言語間のベクトルの翻訳の精度を向上させることができるであろう．

借用語

日本語には「ウェブブラウザ」(*web browser*) や「ビジュアルデザイン」(*visual design*) など英語圏からの借用語が多く存在し，こうした借用語はカタカナで表記されることが一般的である．こうした「カタカナ語」と英語の対応関係を翻訳行列 $\boldsymbol{W}$ の調整に用いることで，「モンスター・ペアレンツ」「アラウンド・サーティ」といった和製英語もそれぞれ *“monster parents”*，*“around thirty”* というように英語の単語に対応付けることができる．こうした情報を用いることで，借用-被借用の関係を多く持つ言語間の翻訳精度を向上させることができる可能性がある．

7.3 ウェブから獲得する言語資源とコーパスの多様化・大規模化

7.2 節で述べたさまざまな言語知識を翻訳行列の学習に利用するため，こうした言語知識をウェブ上から獲得する必要がある．また，実際に使われる語は新語・未知語など辞書に存在しないものも多い．そのため，ベクトルの作成に用いる言語コーパスも，マイクロブログやソーシャルネットワーキングサービスといった，新しい表現が用いられる頻度の高い情報源から作成するのが精度の向上に有効であると考えている．

謝辞

本研究の一部は JSPS 科研費 25280111 の助成を受けたものです．

参考文献

[1] Jeff Mitchell and Mirella Lapata. Vector-based models of semantic composition. In *Proceedings of ACL*, pp. 236–244, 2008.

[2] Reinhard Rapp. Word sense discovery based on sense descriptor dissimilarity. In *Proceedings of the ninth MT Summit*, pp. 315–322, 2003.

[3] Peter D. Turney. Domain and function: A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research*, Vol. 44, pp. 533–585, 2012.

[4] Zellig S. Harris. Distributional structure. *Word*, Vol. 10, pp. 146–162, 1954.

[5] Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint*, 2013.

[6] John R. Firth. A synopsis of linguistic theory. *Studies in Linguistic Analysis*, pp. 1–32, 1957.

[7] Martin Luther King and Martin Luther King Jr. *I have a dream*. MPI Home Video, 1986.

[8] Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Computational linguistics*, Vol. 16, No. 1, pp. 22–29, 1990.

[9] John A. Bullinaria and Joseph P. Levy. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior research methods*, Vol. 39, No. 3, pp. 510–526, 2007.

[10] Dekang Lin and Xiaoyun Wu. Phrase clustering for discriminative learning. In *Proceedings of ACL-IJCNLP*, pp. 1030–1038, 2009.

[11] Shai Shalev-Shwartz, Yoram Singer, Nathan Srebro, and Andrew Cotter. Pegasos: Primal estimated sub-gradient solver for SVM. *Mathematical programming*, Vol. 127, No. 1, pp. 3–30, 2011.

[12] Peter D. Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. Literal and metaphorical sense identification through concrete and abstract context. In *Proceedings of EMNLP*, pp. 680–690, 2011.

[13] Nobuhiro Kaji and Masaru Kitsuregawa. Efficient word lattice generation for joint word segmentation and POS tagging in Japanese. In *Proceedings of IJCNLP*, pp. 153–161, 2013.

[14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint*, 2013.

[15] Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. Inducing crosslingual distributed representations of words. In *Proceedings of COLING*, pp. 1459–1474, 2012.

[16] Jiajun Zhang, Shujie Liu, Mu Li, Ming Zhou, and Chengqing Zong. Bilingually-constrained phrase embeddings for machine translation. In *Proceedings of ACL*, pp. 111–121, 2014.

[17] Karl Moritz Hermann and Phil Blunsom. Multilingual models for compositional distributed semantics. In *Proceedings of ACL*, pp. 58–68, 2014.