E-019

分散表現に基づく日本語の意味的逆引きの検討
A Study on Japanese Reverse Dictionary Based on Distributed Representations

髙橋 寿聡† 梅澤 猛‡ 大澤 範高‡
Toshiaki Takahashi Takeshi Umezawa Noritaka Osawa

1. はじめに

意味は説明できるが言葉が思い浮かばないとき，意味記述文を入力として対応する見出し語を検索する意味的逆引きが役立つ．Hill らは，離散的なトークンである単語を固定長の実数ベクトルで表す分散意味表現（以下，分散表現）を用いた意味的逆引きの手法を提案し，その有効性を示した [1]．Hill らの手法に基づいた既存の意味的逆引き手法の多くは英語を対象としており [2] [3]，言語体系の異なる日本語へ適用した場合の有効性は明らかでない．そこで本研究では，Hill らの手法を日本語に適用した場合の評価実験を行い，分散表現学習に基づく日本語の意味的逆引き手法を検討した．

2. 関連研究

Hill らは，word2vec を用いて単語の分散表現を事前学習したあと，見出し語と意味記述文のペア（以降，これを辞書という）を教師として Recurrent Neural Network（RNN）の教師あり学習を行なった．そして入力される意味記述文を RNN により単語の分散表現空間にマッピングすることで単語の検索を行った．比較として，RNN（LSTM: Long short-term memory）を用いるかわりに分散表現列の総和の線形変換の重みを学習することでマッピングを行う方法の評価も行なっているが，RNN を用いた場合と性能に有意な差は見られないことが示されている [1].

Hill らの手法をベースとして，RNN による入力文の分散表現空間へのマッピングを行うだけでなく，見出し語と意味記述文の対応関係以外の情報を用いる手法も提案されている．見出し語が属するカテゴリの情報や品詞の情報などを用いて，検索対象を絞り込んだり複数のスコア付けを合わせたりすることで性能が向上することが示されている [2] [3]．しかしこれらは英語を対象として評価実験が行われており，日本語へ適用した場合の有効性は明らかでない．

本研究では，単語の分散表現と Neural Network を用いて辞書から意味記述文と見出し語の対応を学習する意味的逆引きの手法を日本語に適用した場合の性能を評価するため，まず Hill らの研究 [1]に基づいた日本語の意味的逆引き手法を検討する．

3. 分散表現に基づく日本語の意味的逆引き

分散表現に基づく意味的逆引きを行うために，単語を d 次元実数ベクトル空間である分散表現空間 $\mathbb{W} = \mathbb{R}^d$ にマッピングするモデル M_W および，$\mathbb{W}$ 上の点の列 $Q' = (\boldsymbol{q}_1, \cdots, \boldsymbol{q}_d)$ を $\mathbb{W}$ 上の点 $\boldsymbol{p}$ にマッピングするモデル M_T の構築を行う．単語を検索するには，まず入力された意味記述文を構成する単語 w_i を M_W を用いて $\mathbb{W}$ 上の点 $\boldsymbol{v}_i$ に変換し，長さ l の単語の列 $Q = (w_1, \cdots, w_l)$ を行列 $Q' \in \mathbb{R}^{d \times l}$ に変換する．つぎに M_T でこれを $\mathbb{W}$ 上の点 $\boldsymbol{p}$ に変換し，検索対象語彙 V の各要素に対応する $\mathbb{W}$ 上の点との距離に基づいて順位付けを行う（図 1）．

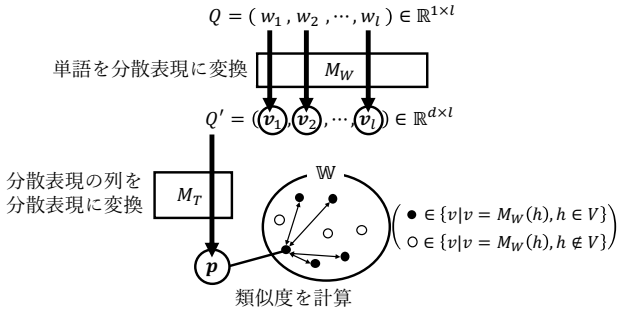

図 1　意味的逆引きの実行

つづいて M_W および M_T について説明する．M_W は，テキストコーパス C 中のテキストを用いた教師なし学習により構築する．本稿では，C から構築した単語 w を d 次元の分散表現 $\boldsymbol{v}$ に変換するモデル $M_W[C]$ を単に M_W と表し，w から $\boldsymbol{v}$ への変換を $\boldsymbol{v} = M_W(w)$ と表す．ただし，単語の列である意味記述文を表す $(1 \times l)$ 行列 $Q = (w_1, \cdots, w_l)$ に対して，$(d \times l)$ 行列 $Q' = M_W(Q) = (\boldsymbol{v}_1, \cdots, \boldsymbol{v}_l)$ は分散表現の列を表す．

M_T は，見出し語 h と意味記述文 S の組 (h, S) を要素とする辞書 D に基づいてモデルを構築する．$(h, S) \in D$を M_W に基づいて分散表現および分散表現列に変換した$(\boldsymbol{h}', S')$ を要素とする辞書 E を作成し（$\boldsymbol{h}' = M_W(h), S' = M_W(S)$），辞書 E に基づいて，S' から $\boldsymbol{h}' \in \mathbb{W}$ に変換するモデル $M_T[E]$ を構築する．本稿では，$M_T[E]$ を単に M_T と表す．

4. 実験

4.1 M_W と M_T の構築

テキストコーパス C は，日本語 Wikipedia のダンプファイルからプレーンテキストを取り出すツール WikiExtractor を用いて記事本文データから作成した．コーパス C の総単語数は 179,915,547 であり，語彙数は 962,571 であった．M_W の構築には word2vec を用い，分散表現の次元数は 500 とした．

辞書 D として，日本語 WordNet [4]のデータから作成した辞書 120,461 件からランダムに 10,000 件，20,000 件，40,000 件を抽出したものを用意し（$|D| = \{10\text{k}, 20\text{k}, 40\text{k}\}$），それぞれを 9:1 に分割した 9 割を D_t，残り 1 割を D_v として，M_T の訓練には D_t を用いた（$D = D_t \cup D_v, D_t \cap D_v = \emptyset, |D_t| : |D_v| = 9:1$）．$M_T$ としては，「RNN（LSTM）」と「分散表現列の要素の総和の線形変換（線形）」の 2 種類を検討した．日本語テキストを単語へ分割する際の形態素解析は，MeCab および NEologd 辞書を用いて行った．

† 千葉大学大学院融合理工学府数学情報科学専攻情報科学コース　Division of Mathematics and Informatics, Graduate School of Science and Engineering, Chiba University
‡ 千葉大学大学院工学研究院　Graduate School of Engineering, Chiba University

Copyright © 2020 by
The Institute of Electronics, Information and Communication Engineers and
Information Processing Society of Japan All rights reserved.

4.2 評価方法

評価に用いる辞書 D_e の要素の意味記述文 S との類似度を基に検索対象語彙 V 中の単語に順位を付与した．評価指標は，D_e 全要素のうち，S に対応する見出し語 $h \in V$ の類似度順位が N 位以内である割合（累積正答率）とした．類似度計算には 2 点間のコサイン距離を用いた．

D_e としては，M_T の構築に用いた辞書 D_t をそのまま用いた場合と D_v を用いた場合の 2 通りを考えた（$D_e = \{D_t, D_v\}$）．V としては，辞書 D_t の見出し語の集合 V_{D_t} とコーパス C に含まれる単語の集合 V_C の 2 通りを考えた（$V = \{V_{D_t}, V_C\}$）．ただし $|V_C| = 962571$ である．M_W, M_T の構築および評価の手順を図 2 に示す．

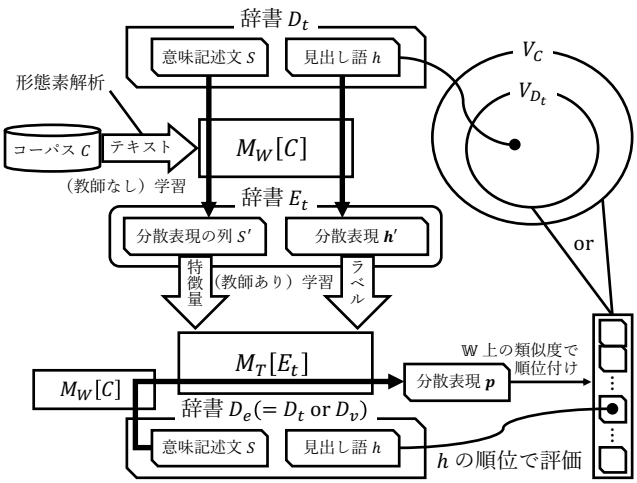

図 2　M_W と M_T の構築およびその評価

4.3 結果

$D_e = D_t, V = V_{D_t}$ の場合の，M_T および $|D_t|$ の組み合わせに対する累積正答率を図 3 に示す．比較として，先行研究である Hill ら [1]の実験結果のうち，条件が $D_e \subset D_t, V = V_{D_t}$ （$|D_e| = 500, |D_t| \approx 900\text{k}, |V_{D_t}| \approx 100\text{k}$）のものも示している．図 3 に示したように，総和の線形変換に比べて LSTM の方が高い結果となった．これは，英語を対象とした Hill らの実験結果 [1]とは異なっている．また本実験の結果では，M_T として総和の線形変換と LSTM の両方において，$|D_t|$ が大きくなるにつれて累積正答率は低くなっている．また，見やすさのため図 3 には示していないが，$D_e = D_t, M_T = \text{LSTM}$ の場合において検索対象語彙を V_{D_t} から V_C に拡大した場合の累積正答率は，$|D_t|$,しきい値の組み合わせすべてを通して最大 3.19 パーセントポイントの低下にとどまった．

次に，$D_e = D_v, V = V_C$ で，M_T として LSTM を用いた場合の，異なる $|D_t|$ に対する累積正答率を図 4 に示す．比較として，M_T の構築を行うかわりに，doc2vec ソフトフェア（PV-DBOW アルゴリズム）を用いて意味記述文 $\boldsymbol{s}$ を 500 次元ベクトルに変換し，そのベクトルとの類似度により単語の検索を行なった（$V = V_C$）場合の結果も示している．図 4 に示したように，$|D_t|$ が大きくなるにつれて累積正答率も高くなっているが，$D_e = D_t, V = V_C$ の場合に比べると，最大で（$|D_t| = 9\text{k}$, しきい値 10のとき）93.33 パーセントポイント低下し著しく低い結果となった．

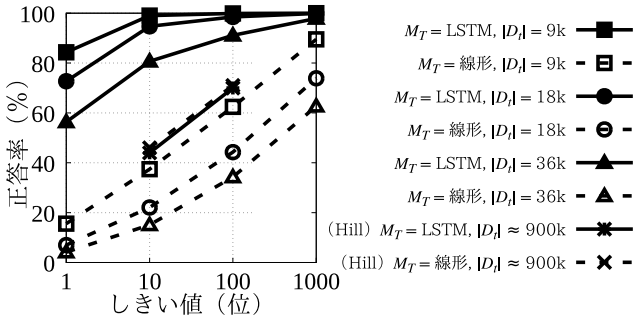

図 3　$D_e = D_t, V = V_{D_t}$ の場合の累積正答率の比較

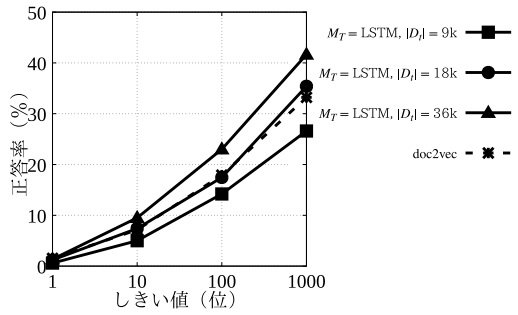

図 4　$D_e = D_v, V = V_C$ の場合の累積正答率の比較

4.4 考察

図 3 に示した結果から，日本語の意味的逆引きにおいては，M_T として総和の線形変換よりも LSTM を用いる方が適していることが示唆された．また，総和の線形変換よりも LSTM による変換の方が性能が高かったことから，M_T として異なる変換モデルを用いることでさらに性能が向上する可能性がある．汎化性能に関して，図 4 に示した結果から，試した範囲では辞書サイズが大きくなるにつれて累積正答率が向上しており，入力される意味記述文および対応する見出し語がともに D_t に含まれていない場合の性能は，D_t の件数を増やすことで向上することが示唆された．

5. おわりに

分散表現に基づく日本語の意味的逆引きでは，意味記述文の分散表現列を単語の分散表現空間にマッピングする変換モデルとして，英語の場合と異なり総和の線形変換よりも LSTM による変換の方が適していることが示唆された．

参考文献

[1] Hill Felix, Cho Kyunghun, Korhonen Anna, Bengio Yoshua, “Learning to understand phrases by embedding the dictionary”, Transactions of the Association for Computational Linguistics, vol.4, pp.17-30 (2016).

[2] 森永 雄也, 山口 和紀, “カテゴリ情報を付与した文の分散表現による逆引き辞書の精度向上”, SIG-AM, Vol.16, No.01, pp.1-8 (2017).

[3] Zhang Lei, Qi Fanchao, Liu Zhiyuan, Wang Yasheng, Liu Qun, Sun Maosong, “Multi-channel Reverse Dictionary Model”, arXiv preprint arXiv:1912.08441 (2019).

[4] Bond Francis, Isahara Hitoshi, Fujita Sanae, Uchimoto Kiyotaka, Kanzaki Kyoko, “Enhancing the Japanese WordNet”, Proceedings of the 7th workshop on Asian language resources, pp.1-8, Association for Computational Linguistics (2009).

Copyright © 2020 by
The Institute of Electronics, Information and Communication Engineers and
Information Processing Society of Japan All rights reserved.