yamaMomo：Sudachi 同義語辞書による 日本語分散表現の評価用データセットの作成

野口 夏希 [1] 勝田 哲弘 [2] 山村 崇 [2] 高岡 一馬 [2] 内田 佳孝 [2]
[1] 愛媛大学 [2] 株式会社ワークスアプリケーションズ・エンタープライズ
n_noguchi@ai.cs.ehime-u.ac.jp
{katsuta_a，yamamura_t，takaoka_k，uchida_yo}@worksap.co.jp

概要

分散表現の評価のうち内的評価では，単語間の類似度や意味に関係がある単語を正しく類推できるかを評価するタスクが特に注目され，日本語でもデータセットが作られている．しかし他の種類の内的評価タスクではデータセットが不足している．

本研究では表記揺れや略語・略称，カテゴリの類似度に関する内的評価タスクに注目し，Sudachi 同義語辞書を用いてデータセットを作成した．またこのデータセットで評価を行い，コーパスや形態素解析に起因する分散表現モデルの違いを確認した．

1 はじめに

分散表現の評価タスクには内的評価と外的評価がある．内的評価は単語分散表現の性能を直接評価し，外的評価は実際のタスクに単語分散表現を使って評価するものである．Bakarov [1] によると内的評価は，Word Semantic Similarity，Word Analogy，Thematic Fit，Concept Categorization，Synonym Detection，Outlier Word Detection の 6 つのタスクに分類表記ゆれ．

既存の日本語分散表現のデータセットには単語の意味がどれほど近いかを指す類似度や，どれほど関連するかを指す関連度の評価をするものがある [2] が、他の内的評価タスクのためのデータセットは不足している．

本研究では Outlier Word Detection と Concept Categorization のうち，略語・略称や表記揺れといった同義語間における類似度や，単語の属する分野に注目し，Sudachi 同義語辞書 [3] を用いて日本語分散表現の評価用データを作成した．

Sudachi 同義語辞書は Sudachi 形態素辞書に人手で同義語関係を付与した辞書である．詳細化された同義語関係として略語・略称であるかを示す略語・略称情報，異表記，翻字，誤字など表記揺れの種別を示す情報，単語がどのような概念に属するかを示す分野情報が付与されている．

本研究の貢献は以下の 3 つである．(1) 日本語のデータセットになかったタスクに着目したデータセット及び評価指標の作成　(2) 詳細な同義関係（代表表記と表記揺れ）や分野情報が評価できるデータセットの作成　(3) 本論文で作成したデータセットによる各モデルの評価．

2 先行研究

Bakarov [1] によると，内的評価のタスクは 6 種に分類される．

英語の Outlier Word Detection のためのデータセットである The 8-8-8 outlier detection dataset [4] では，単語集合の中から cos 類似度を用いて単語ベクトルの距離を測ることで outlier を選択している．英語の Concept Categorization [5] のためのデータセットである ESSLLI-2008 [6] では，単語集合をクラスタリングツールキットを用いてクラスタリングすることで単語集合をカテゴリごとに分類している．

しかし，日本語では 6 つのタスクのうち，Word Semantic Similarity [2, 7, 8, 9]，Word Analogy [8] のデータセットが公開されているに留まる．

3 データセットの作成と評価

本実験では，Outlier Word Detection と Concept Categorization のデータセットの作成に，Sudachi 同義語辞書[1)] [3] を使用した．またデータセットに含める単語には，評価対象となる複数の分散表現モデルで共通して出現する語を選択した．共通する語をできるだけ多くするため，分散表現モデルは大規模な chiVe [10], nwjc2vec [11], 朝日新聞単語ベクトル [12],

1） https://raw.githubusercontent.com/WorksApplications/SudachiDict/develop/src/main/text/synonyms.txt

WikiSudachiVec, 日本語 Wikipedia エンティティベクトル [13] の 5 つを使用した. WikiSudachiVec はコーパスに 日本語版 Wikipedia, 形態素解析に Sudachi（C 単位）を用いて学習された日本語分散表現である.

3.1 Outlier Word Detection

Outlier Word Detection は，単語集合 W が与えられたときに，単語集合内の残りのグループに属さない outlier の単語 w_o を推定するタスクである．本研究では，Sudachi 同義語辞書内で定義されている 2 単語の同義語ペア集合 S の各ペア $s = \{w_{s_1}, w_{s_2}\} \in S$ に，s に対して同義語ではないランダムな単語 w_o を加えた $W_s^o = \{w_{s_1}, w_{s_2}, w_o\}$ を作成する．

同義語ペア s の 2 単語は，Sudachi 同義語辞書において，代表語に対して表記揺れや略語・略称として定義されている同義語関係にあるものを選んだ. 表記揺れの中で，異表記（送り仮名の有無のような日本語の表記揺れ）と翻字関係（日本語の単語が英単語で書かれているような表記揺れ）を区別する. これにより，モデルがどの程度表記揺れに頑健であるかだけでなく，言語間での表記揺れはあまり区別できないなどの，詳細な評価もできると考えられる. これらの略称の表記揺れを評価することで，略称が出現しやすいようなドメインの文書に対するモデルの有効性などを評価できると考えられる.

また， outlier の単語の選び方によっては推定が容易であるような単語のみが選ばれる可能性がある. これを回避するために，各同義語ペア s に対して k 個の異なる outlier の単語 $\{w_{o_1}, ..., w_{o_k}\}$ をそれぞれ追加したとき，すべての組み合わせ $X_s = \{W_s^o, ..., W_s^{o_k}\}$ で outlier が推定できた割合で評価する．本研究では $k = 10$ とし，s に対して同義語ではないランダムな 10 単語を outlier として選択した. 表 1 に具体例を示す. 作成した同義語ペアの合計 $|S|$ は，異表記が 313 件，翻字関係が 859 件，略称が 182 件となった.

次に，このデータセットを使った分散表現の評価方法について述べる. 単語集合 W 内のある単語 w が w_o であるか推定する方法として，式 (1) から他のすべての単語 $W \setminus \{w\}$ との類似度 $score_W(w)$ を求め，最も類似度が低い単語 w を w_o と推定する.

$$score_W(w) = \frac{1}{|W| - 1} \sum_{w_j \in W \setminus \{w\}} sim(w, w_j) \quad (1)$$

ここで， $sim(\cdot)$ は単語間の類似度を示し，先行研究 [4] に倣ってコサイン類似度を用いた.

単語集合 W において w_o が正しく推定できたときに 1 (それ以外は 0) をとる関数 $TP(W)$ と，すべての outlier の組み合わせ X_s において w_o が正しく推定できたときに 1 (それ以外は 0) をとる関数 $TP_{all}(X)$ をそれぞれ式 (2, 3) のように定義すると，

$$TP(W) = \begin{cases} 1, & \text{if } \arg\min_{w \in W} score_W(w) = \{w_o\}, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

$$TP_{all}(X_s) = \begin{cases} 1, & \text{if } \sum_{W \in X_s} TP(W) = k \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

すべての同義語ペアに対する正解率 $Acc.$ は式 (4) のように定義される.

$$Acc. = \frac{\sum_{s \in S} TP_{all}(X_s)}{|S|} \quad (4)$$

3.2 Concept Categorization

Concept Categorization は，単語集合 W が与えられたときに，異なるカテゴリの部分集合ごとに分割するタスクである．本研究では Sudachi 同義語辞書の 31 種類の分野情報（IT，キャラなど）をカテゴリとし，2 つカテゴリ i, j を選び，各カテゴリの 2 単語ずつあわせた 4 単語のサンプル $W_{ij} = \{w_{i_1}, w_{i_2}, w_{j_1}, w_{j_2} \mid i \neq j\}$ を作成し，すべてのサンプルの集合を D と定義する. 具体例を表 2 に示す. 作成したすべてのサンプルの合計 $|D|$ は，104, 625 件となった.

次に，このデータセットを使った分散表現の評価方法について述べる. 単語集合 W に対して，各単語 w の埋め込み表現を素性としてクラスタリングを行い，カテゴリごとに分類できたかを評価する．クラスタリングのアルゴリズムには，AgglomerativeClustering[2)] を用い， パラメータとして affinity は cosine を，linkage は average を指定する. 先行研究 [5] では，クラスタリングの評価として各クラスタが単一のカテゴリから構成されている割合を表す purity を用いているが，本研究ではデータセット D に対してどのくらいのサンプルで正解できたかを評価する．正しくクラスタリングできたサンプルの集合を $D_{correct}$ とすると，正解率 $Acc.$ は式（5）のように定義される.

2) https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

表 1　Outlier Word Detection のデータの例

同義関係	代表語	同義語	outlier
異表記	入り口	入口	稽古，ジュース，茜，ローゼット，狭間 錘，誤る，アブ，ロー，グアテマラ
翻字関係	イエロー	yellow	ノベル，gang，cookie，coach，ジョイント bucket，mall，フラッグ，プリンセス，ステート
略称	アカウント	アカ	サポ，コンペティション，コンパ，メーキャップ，ナレーション ロケーション，メカ，ロボット，フェスティバル，ゼミナール

$$Acc. = \frac{|D_{correct}|}{|D|} \tag{5}$$

4　実験設定

3.1 項と 3.2 項で構築した Outlier Word Detection と Concept Categorization のデータセットを用いて分散表現の評価を行った．使用したモデルはコーパスや形態素解析器による差異を分析するために、以下の 13 個を用意した．

chiVe [10]：　**chiVe_mc5**, **chiVe_mc15**, **chiVe_mc90**
コーパスに NWJC [14], 形態素解析に Sudachi を用いて学習した分散表現であり，min-count が異なる 3 つのモデルを使用する．

nwjc2vec [11]：　**nwjc2vec**
コーパスに NWJC, 形態素解析に MeCab を用いて学習した分散表現を使用する．

日本語 Wikipedia エンティティベクトル [13]：
entityVec
コーパスに Wikipedia, 形態素解析に MeCab を用いて学習した分散表現を使用する．

朝日新聞単語ベクトル [12]：　**A-CBOW**，　**A-CBOW-R**, **A-GloVe**, **A-GloVe-R**, **A-Skip**, **A-SKip-R**
コーパスに新聞記事，形態素解析に MeCab を用いて学習した分散表現．3 つの学習アルゴリズム (CBOW, Skip-Gram, GloVe) によって学習したモデルと，Retrofitting によってファインチューニングされた計 6 つのモデルを使用する．

WikiSudachiVec：　**WikiSuda-A**, **WikiSuda-C**
コーパスに Wikipedia, 形態素解析に Sudachi（それぞれ A 単位, C 単位）を用いて学習した分散表現を使用する．

Outlier Word Detection や Concept Categorization タスクによって各モデルが表記揺れや分野情報をコーパスから学習できているかを評価する．

5　実験結果

表 3 に実験結果を示す．

5.1　実験 1：Outlier Word Detection

Outlier Word Detection の異表記と略称では全体的に chiVe が高い性能を示した．データセットのもとになった Sudachi 同義語辞書と訓練コーパスの分割単位が一致していることが影響したと考えられる．翻字関係は chiVe と entityVec の正解率が高かった．全モデルを通じて表記種別 (カタカナかアルファベット) だけをみて誤判別する傾向が目立った．chiVe は形態素解析に用いた Sudachi の正規化によって，文脈から翻字関係の表記揺れが学習できている可能性がある．また entityVec は，記事本文中のハイパーリンクを使用し，それぞれのリンクのアンカーテキストをリンク先の記事のタイトルに置換する方法を採っており，それが翻字の学習に有効だったためと考えられる．

朝日新聞単語ベクトルは学習に用いたアルゴリズムや Retrofitting の有無で正解率に差が出ていた．CBOW は周辺単語から単語を予測，GloVe は単語からコーパス全体から得た共起頻度を用いて周辺単語を予測，Skip-gram は単語から周辺単語を予測するアルゴリズムである．A-Glove の精度が A-CBOW や A-Skip よりも低いのは，GloVe が共起頻度を考慮することで表記揺れによる細かい差異を強調したからだと考えられる．Retrofitting は周辺単語が近いベクトルになるように fine-tuning する方法である．Retrofitting によって正解率が高くなったのは，Retrofitting に使用されている日本語 WordNet の中に表記揺れのペアも存在するため，ベクトルが近づけられたからだと考えられる．

各々のデータセットについてすべてのモデルでタスクを正しく解けなかった単語集合は，異表記が 19 件，翻字関係が 26 件，略称が 4 件あった．[5, 五, 7],

表 2 Concept Categorization のデータの例

分野 1 の単語		分野 2 の単語	
アップデート（IT）	ウェブサイト（IT）	配置（建築）	レイアウト（建築）
コピペ（IT）	コピーペースト（IT）	特会（ビジネス）	インターンシップ（ビジネス）

表 3 モデルによる評価
（太字：正解率が最も高かったもの上位 3 件）

モデル	Concept	異表記	翻字関係	略称
chiVe_mc5	**0.616**	**0.818**	**0.864**	**0.896**
chiVe_mc15	0.608	**0.815**	**0.843**	**0.901**
chiVe_mc90	**0.617**	**0.808**	**0.823**	**0.896**
nwjc2vec	**0.612**	0.748	0.156	0.835
entityVec	0.465	0.476	**0.823**	0.577
A-CBOW	0.498	0.594	0.164	0.731
A-CBOW-R	0.526	0.719	0.171	0.808
A-GloVe	0.336	0.562	0.012	0.621
A-GloVe-R	0.358	0.712	0.012	0.742
A-Skip	0.493	0.684	0.049	0.758
A-Skip-R	0.512	0.764	0.063	0.824
WikiSuda-A	0.536	0.700	0.540	0.731
WikiSuda-C	0.561	0.706	0.537	0.725

[バス, bass, light]，[コンパニー, コンパ, ラボ] など多義性があるものが含まれており，語義曖昧性によって，静的なベクトルでは学習が難しいデータもあることが考えられる．

5.2 実験 2：Concept Categorization

Concept Categorization の正解率は chiVe が最も良かった．これは，多くのドメインを持つ国語研日本語ウェブコーパスにより訓練しており，各カテゴリの事例が十分に存在したためだと考えられる．

各モデルで学習されている単語のカテゴリの違いを見るためにカテゴリ別の正解数による分散，単語集合の種類ごとの正解率を確認した．単語集合の種類ごとの正解率を表 4 に示す．

分散はモデル内でカテゴリごとに差があり，分類タスクの行いやすさはカテゴリで均一でないことがわかった．これにより，データセット内で解きやすい単語集合とそうでない単語集合が含まれていたことが考えられる．

単語集合の種類はモデル内でタスクの行いやすさが異なることがわかった．これにより，モデルによって分類しやすいカテゴリのペアに偏りがあることがわかった．

表 4 カテゴリペアによる正解率

モデル	IT-化学	キャラ-建築	商品-音楽
chiVe_mc5	0.902	0.408	0.298
chiVe_mc15	0.933	0.231	0.329
chiVe_mc90	0.933	0.351	0.284
nwjc2vec	0.942	0.511	0.311
entityVec	0.271	0.667	0.231
A-CBOW	0.307	0.840	0.271
A-CBOW-R	0.307	0.893	0.222
A-GloVe	0.053	0.378	0.102
A-GloVe-R	0.067	0.609	0.236
A-Skip	0.191	0.804	0.231
A-Skip−R	0.169	0.698	0.186
WikiSuda-A	0.916	0.458	0.396
WikiSuda-C	0.924	0.551	0.427

6 まとめ

本研究では Sudachi 同義語辞書を用いて，分散表現の内的評価タスクである Outlier Word Detection と Concept Categorization による日本語評価データセットを作成した．Sudachi 同義語辞書に掲載されている略語・略称情報，表記揺れ情報，分野情報によって，これまで評価されていなかった表記揺れやカテゴリを分散表現が学習できているかを評価し、各モデルで性能差があることを示した．

今後は，不足している他のタスクの日本語データセットの作成も検討したい．また，本論文で作成したデータセットは公開する予定である．

参考文献

[1] Amir Bakarov. A survey of word embeddings evaluation methods, 2018.

[2] 猪原敬介, 内海彰. 日本語類似度・関連度データセットの作成. 言語処理学会第 24 回年次大会 (NLP2018). 言語処理学会, 2018.

[3] 高岡一馬, 岡部裕子, 川原典子, 坂本美保, 内田佳孝. 詳細化した同義関係をもつ同義語辞書の作成. 言語処理学会第 26 回年次大会 (NLP2020). 言語処理学会, 2020.

[4] José Camacho-Collados and Roberto Navigli. Find the word that does not belong: A framework for an intrinsic evaluation of word vector representations. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Repre-*

sentations for NLP, pp. 43–50, Berlin, Germany, August 2016. Association for Computational Linguistics.

[5] Marco Baroni, Georgiana Dinu, and Germán Kruszewski. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 238–247, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[6] Stefan Evert Marco Baroni and Alessandro Lenci. Bridging the gap between semantic theory and computational simulations. In *Proceedings of the ESSLLI Workshop on Distributional Lexical Semantic*, FOLLI, Hamburg, 2008.

[7] Yuya Sakaizawa and Mamoru Komachi. Construction of a japanese word similarity dataset, 2017.

[8] Marzena Karpinska, Bofang Li, Anna Rogers, and Aleksandr Drozd. Subcharacter Information in Japanese Embeddings: When Is It Worth It? In *Proceedings of the Workshop on the Relevance of Linguistic Structure in Neural Architectures for NLP*, pp. 28–37, Melbourne, Australia, 2018. Association for Computational Linguistics.

[9] Tomonori Kodaira, Tomoyuki Kajiwara, and Mamoru Komachi. Controlled and balanced dataset for Japanese lexical simplification. In *Proceedings of the ACL 2016 Student Research Workshop*, pp. 1–7, Berlin, Germany, August 2016. Association for Computational Linguistics.

[10] 真鍋陽俊, 岡照晃, 海川祥毅, 高岡一馬, 内田佳孝, 浅原正幸. 複数粒度の分割結果に基づく日本語単語分散表現. 言語処理学会第 25 回年次大会 (NLP2019). 言語処理学会, 2019.

[11] Masayuki Asahara. 'nwjc2vec: Word embedding dataset from 'ninjal web japanese corpus''. *Terminology: International Journal of Theoretical and Applied Issues in Specialized Communication*, Vol. 24, No. 2, pp. 7–25, 2018.

[12] 田口雄哉, 田森秀明, 人見雄太, 西鳥羽二郎, 菊田洸. 同義語を考慮した日本語単語分散表現の学習. 情報処理学会研究報告, 第 2017-NL-233 巻, pp. 1–5. 情報処理学会第 233 回自然言語処理研究会, 2017.

[13] Masatoshi SUZUKI, Koji MATSUDA, Satoshi SEKINE, Naoaki OKAZAKI, and Kentaro INUI. A joint neural model for fine-grained named entity classification of wikipedia articles. *IEICE Transactions on Information and Systems*, Vol. E101.D, No. 1, pp. 73–81, 2018.

[14] Masayuki Asahara, Kikuo Maekawa, Mizuho Imada, Sachi Kato, and Hikari Konishi. Archiving and analysing techniques of the ultra-large-scale web-based corpus project of ninjal, japan. *Alexandria*, Vol. 26, No. 1-2, pp. 129–148, 2014.

A　分野情報による正解率の差

表 3 の Concept に対してより詳細に分野情報ごとの精度を出した結果を以下の表に示す.

表 5　分野情報ごとの正解率

分野情報	chiVe_mc90	nwjc2vec	entityVec	A-CBOW	A-GloVe	A-Skip
IT	0.716	0.747	0.618	0.642	0.397	0.633
キャラ	0.394	0.363	0.535	0.317	0.174	0.392
スポーツ	0.714	0.691	0.560	0.685	0.393	0.673
ビジネス	0.574	0.673	0.504	0.410	0.270	0.435
ファッション	0.512	0.566	0.337	0.444	0.149	0.422
交通	0.675	0.718	0.482	0.541	0.384	0.552
人	0.543	0.435	0.412	0.450	0.298	0.384
人名	0.619	0.623	0.595	0.519	0.350	0.519
企業名	0.582	0.592	0.528	0.529	0.397	0.495
動植物	0.686	0.664	0.571	0.540	0.377	0.576
化学	0.710	0.617	0.122	0.117	0.050	0.072
医療	0.639	0.655	0.513	0.534	0.379	0.468
単位	0.526	0.496	0.354	0.425	0.384	0.367
商品	0.228	0.243	0.135	0.260	0.251	0.215
国名	0.729	0.707	0.682	0.651	0.459	0.651
地名	0.663	0.707	0.557	0.570	0.365	0.560
地形	0.699	0.655	0.476	0.642	0.446	0.608
娯楽	0.673	0.701	0.602	0.586	0.407	0.566
店	0.566	0.549	0.258	0.413	0.054	0.319
店名	0.662	0.656	0.444	0.578	0.467	0.609
建築	0.552	0.723	0.483	0.643	0.381	0.602
政治	0.784	0.770	0.783	0.674	0.436	0.703
教育	0.800	0.800	0.712	0.687	0.436	0.728
料理	0.612	0.655	0.450	0.517	0.429	0.554
時間	0.455	0.209	0.332	0.197	0.293	0.201
法律	0.655	0.562	0.307	0.475	0.229	0.522
組織名	0.402	0.363	0.194	0.279	0.209	0.308
美容	0.691	0.729	0.381	0.520	0.421	0.539
色	0.705	0.666	0.551	0.544	0.399	0.566
観光	0.590	0.673	0.297	0.367	0.300	0.391
音楽	0.783	0.771	0.643	0.633	0.419	0.642