特定のドメインに特化した日本語同義語獲得の検討

勝又智 [1]　飯田大貴 [1,2]
[1] 株式会社レトリバ　[2] 東京工業大学
{satoru.katsumata, hiroki.iida}@retrieva.jp

概要

自然言語処理を特定のドメインのテキストに応用する際，Entity に関する情報はドメイン特有の知識となることが多く，公開されている同義語辞書だけでなく，特定のドメインの同義語辞書の作成が求められる．しかし，この同義語辞書は人手で作成することが多く，非常に高コストである．本研究では同義語辞書作成支援を目的として，ベースライン及び評価用データの構築，人手評価を実施した．実験の結果，単語間類似度タスクで有効な分散表現作成手法が，本タスクでは有効ではなく，またその逆が起こりうることがわかった．

1　はじめに

自然言語処理技術の活用現場では，同義語辞書の作成が課題としてあげられることが多い．例えば，情報検索の分野では同義語辞書を使用することで精度が向上することが知られている [1]．WordNet などのオープンドメインにおける同義語辞書よりも，特定のドメインに沿った同義語辞書を使うことでさらなる精度向上が期待できる．特に，Entity の同義語情報はドメイン特有の知識となることが多く，特定のドメインに応じて作成する必要がある．しかし，特定のドメインに特化した同義語辞書作成は，人手で行われることが多く非常に高コストである．そのため，同義語辞書作成を支援する必要がある．

支援方法として有効な方法の一つに，対象の Entity に対する同義語候補をランキングで示す方法がある．このような形式のタスクに Zero-shot Entity Linking (EL) [2] がある．Zero-shot EL は訓練中に現れない Entity に対して，Entity 集合を適合度順にランキングするタスクである．しかしながら，この Zero-shot EL においては，Entity に関する説明文を利用可能な知識としているが，説明文を利用できないケースは応用上多い．

そこで，本研究では Entity の情報のみから，Entity 集合を適合度順に並べるタスク（**同義語獲得タスク**）を実施する．ところが，このような評価データセットは我々の知る限り存在しない．本研究では，特定のドメインのテキストに対して Entity 抽出器を用いて Entity 集合（**mention 集合**）を作成する．また，ユーザーが入力するクエリ Entity（**term**）を別途用意し，このクエリに対して mention 集合を適合度順に並べるタスクを行う．概要を図 1 に示す．ベースラインとして編集距離によるランキングと分散表現を用いた類似度によるランキングを行い，人手によって性能を評価した．

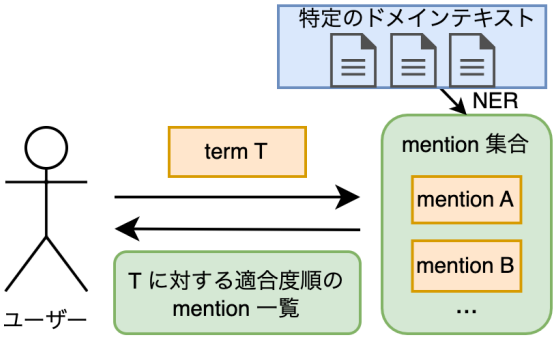

図 1　同義語獲得タスクの概要図

2　同義語獲得タスク

2.1　同義語獲得タスクの問題設定

同義語辞書を含む，様々な知識を活用した研究が盛んに行われている．日本語同義語辞書については，日本語 WordNet [3] や ConceptNet [4]，Sudachi 同義語辞書 [5] などの公開されている辞書が存在する．しかし，これらの同義語辞書は一般ドメインに対して構築されており，特定のドメインに関する Entity は含まれていないことが多い．そこで，本研究では特定のドメインのテキストから同義語辞書を作成する支援を行うタスクを検証する．

今回取り扱う問題設定は Zero-shot EL [2] と類似した設定としている．本問題設定では，図 1 のよう

表 1 分散表現作成手法一覧

手法	in-domain の使用	文脈	部分文字列のマッチ
fasttext	なし	なし	あり
BERT	なし	なし	あり
SimCSE	あり	なし	あり
BLINK	なし	あり	なし
LUKE	なし	あり	なし

に同義語の探索を行いたい特定のドメインのテキストに対して Named Entity Recognition (NER) を行い，取得した全ての種類の Entity を mention 集合とする．別途用意した term をクエリとして，mention 集合を適合度順に並べるタスクとして設定した．つまり，本問題設定は Zero-shot EL における Entity を，ユーザーが入力する term として定義した．また，先行研究では明示的な EL のデータを学習に使用しているが，実応用上ではそのようなデータを用意することは難しいため，本研究では EL データは使用しない．

本研究に類似した問題設定として，単語間類似度タスク [6, 7] が挙げられる．近年の単語間類似度タスクは，分散表現による単語間の類似度について，人間の判断との近さを評価するタスクである．本研究の問題設定との違いとして，入力の性質が異なることが挙げられる．具体的には，本研究では単語として連続した形態素を想定しており，この単語は全て Entity であるため，ドメイン特有の知識が分散表現作成に必要となる点が異なる．本研究ではこれらの差異による影響を確認するため，単語間類似度タスクと同義語獲得タスクとの相関を調査した．

2.2 適合基準

2.1 節の通り，同義語獲得タスクは同義語辞書作成支援を目的としている．そのため，本研究では評価として各システムの出力のうち，最も適合度が高い mention について，入力された term の同義語かどうかの人手評価を実施した．同義語の適合基準として，**互換性**と**同位語性**という概念を導入した．

互換性は，term と最も適合度が高い mention を入れ替えても意味が同じである性質としている．具体的には，term とその term が出現する文 $\text{Context}_{\text{term}}$ を用意し，この文が最も適合度が高い mention を term と入れ替えた文 $\text{Context}_{\text{mention}}$ と同一の意味であるかどうか確認した．

しかし，互換性のみでは，term が上位語，最も適合語が高い mention が下位語の場合といった上位下位概念の場合に対して評価が難しいものとなる．例えば term として ‘動物’ が入力され，最も適合度の高い mention が ‘犬’，term が出現する文が ‘動物を飼う．’ の場合，これらの関係が互換性を満たすかどうかは判断が分かれると考えられる．そこで本研究では，これらの場合を区別するため同位語性を導入した．同位語性は，term と最も適合度が高い mention の関係について，上位下位概念ではなく，共通の上位概念を持つ語（同位語）である性質としている．同位語性を導入することにより，上記の例の関係を，互換性は満たすが，同位語性は満たしていないとすることができる．本研究ではこのように，term と最も適合度が高い mention について，互換性があるかないか，同位語性があるかないかのアノテーションを行った．本論文では，互換性があり同位語性もある場合を**強い同義語**と呼称する．

表 2 特定のドメインのテキストコーパス一覧

データ	文書数	mention 集合サイズ
Aamazon Review		
book	5,836	3,459
electronics	10,362	3,905
リコール文書		
国交省	8,261	36,380
消費者庁		
車両・乗り物	3,156	9,879
家電製品	866	5,251
Twitter	21,265	17,935

3 実験

3.1 分散表現作成手法

本研究は term と mention の適合度推定に分散表現のコサイン類似度計算を用いた．検証した分散表現作成手法を表 1 に記載する．また，分散表現ではなく，編集距離を用いた場合も検証した．以下に各分散表現の概要を記載し，詳細は付録 A に記載する．

fasttext 単語ベクトルを元にした手法として，日本語 Wikipedia[1] で学習した fasttext [8] を用いた．term 及び mention の分散表現作成の際は，各形態素をベクトル化し，それらの平均を計算した．

BERT 大規模モデルを用いた手法として，学習済み BERT[2] [9] を用いた．term 及び mention の分散表現は BERT の各サブワードごとの出力を平均したベクトルを使用した．

1） 本研究では 2022 年 6 月 20 日時点でのファイルを利用した．
2） 特に明記しない限り BERT 関連のモデルは全て cl-tohoku/bert-base-japanese-v2 を使用．

表 3 同義語獲得検証結果．左の数値が強い同義語を正解とした時の精度で，右の数値が互換性または同位語性を満たした場合を正解とした時の精度．

	Amazon Review		リコール文書				
手法	book	electronics	国交省	車両・乗り物	家電製品	Twitter	平均
編集距離	0.04/0.14	0.07/0.11	0.07/0.10	0.12/0.16	0.22/0.26	0.13/0.17	0.10/0.16
fasttext	0.07/0.15	0.06/0.13	0.04/0.08	0.07/0.10	0.14/0.17	0.05/0.05	0.07/0.11
BERT	0.10/0.17	0.08/0.13	0.10/0.17	**0.14/0.18**	**0.26/0.34**	0.15/**0.24**	0.14/**0.21**
SimCSE	**0.13/0.21**	**0.10/0.15**	0.11/0.13	**0.14**/0.15	**0.26**/0.30	**0.16**/0.20	**0.15**/0.19
BLINK	0.12/0.20	**0.10**/0.12	**0.15/0.19**	**0.14**/0.15	**0.26**/0.31	0.11/0.16	**0.15**/0.19
LUKE	0.04/0.08	0.05/0.06	0.09/0.09	0.09/0.11	0.15/0.16	0.07/0.07	0.08/0.10

SimCSE 近年，文間類似度などのタスクで研究されている SimCSE [10] についても検証を行った．公開されている BERT モデルに対して， Unsupervised SimCSE を用いて特定のドメインのデータで学習を行ったモデルを使用する．term 及び mention の分散表現作成方法については BERT と同様である．

BLINK Wu ら [2] が提案したモデル（BLINK）についても検証を行った．本研究では他の手法との公平な比較のため，BLINK モデルのうち， Bi-Encoder の構造のみ用いた．Bi-Encoder の学習は，公開されている BERT に対して日本語 Wikipedia を用いて行った．term ベクトル作成時に必要となる説明文は空白としている．mention ベクトル作成時に必要となる mention の前後文脈は，特定のドメインのコーパス中で出現したものを使用した．また，彼らは BERT の出力として [CLS] に対応するベクトルを使用しているため，本研究でも同様の処理を行った．

LUKE Entity の分散表現作成手法として研究されている LUKE[3] [11] についても検証を行った．LUKE は Entity が出現する文脈を使用するため，mention については BLINK と同様に，mention が出現した文脈を LUKE に入力してベクトル化を行った．term については出現する文脈は想定していないため，その語のみを入力しベクトル化を行った．

3.2 データ

本研究では特定のドメインのテキストコーパスとして表 2 のデータを使用した．各データの詳細については付録 B に記載する．

ユーザーが入力する term の模擬として，本研究では Wikipedia 記事のタイトルを使用した．各特定のドメインのテキストと Wikipedia 記事の対応関係を，Piratla ら [12] が提案した情報検索システムを利用した関連文書獲得手法で取得し，その中で関連度が高い 100 記事のタイトルを term として用いた．

特定のドメインのコーパスから mention 集合を作成する際の NER モデルとして，本研究では GiNZA モデル[4] を元に Wikipedia のアンカータグを利用して学習したモデルを用いた．

3.3 評価

本研究では 2.2 節で述べた適合基準を用いて評価を行う．具体的には，評価者に term と term に対して最も適合度が高いと推定された mention， term を含む文書を提示する．term は Wikipedia の記事タイトルであるため，対応する記事タイトルの定義文書が存在しており，本研究ではこの文書を提示している．評価者はこれらから，term と mention 間について互換性と同位語性についてそれぞれ満たすかどうかアノテーションを行った．なお，このアノテーションは 1 人で行っている．

評価尺度としては，強い同義語を満たす場合のみを正解とした場合の accuracy 及び，term と mention 間で互換性または同位語性を満たす場合を正解とした場合の accuracy を計測した．

3.4 実験結果

各特定のドメインのテキストに対する評価結果を表 3 に示す．表 3 の右端列に各ドメインの精度を平均した結果を示している．この結果から，BERT, SimCSE，BLINK は同程度の結果となっていることがわかる．これらの場合について，強い同義語の場合で 0.14 から 0.15，互換性または同位語性を満たす場合でも BERT の 0.21 が最大という結果となり，同義語辞書作成支援に本システムを使用した場合，およそ 1/7 から 1/5 のクエリに対して適切な同義語候補が出力されることが想定される．

本検証ではさまざまなデータで検証を行ったが，

3） studio-ousia/luke-japanese-base を使用．

4） https://megagonlabs.github.io/ginza/ の ja_ginza v5 を使用．

表 4 単語間類似度実験．上段は各手法の結果を，下段は同義語獲得検証（強い同義語）との相関を示している．

手法	spearman	pearson	MRR	Hits@1	Hits@3
fasttext	0.243	0.244	0.208	0.109	0.237
BERT	0.060	0.019	0.177	0.120	0.204
SimCSE	0.231	0.233	0.187	0.091	0.233
BLINK	0.274	0.285	0.271	0.156	0.313
LUKE	0.163	0.179	0.217	0.135	0.236
spearman	0.100	0.100	-0.100	-0.100	-0.200
pearson	-0.019	-0.085	-0.037	0.015	0.238

どのデータでも BERT，SimCSE，BLINK が他の手法と比べて高い精度となっている．表 1 の通り，BERT，SimCSE，BLINK といった手法は学習に特定のドメインのデータの使用や，文脈の使用，類似度計算の際に部分文字列を使うかどうかといった違いはあるものの，これらの違いが本タスクに対して大きく影響することはないと思われる．同じく BERT の構造を元にした手法で LUKE のみ精度が低い結果となったが，これはおそらく LUKE のベクトル作成の構造が本タスクに向いていないためと思われる．LUKE は Entity ベクトル作成の際に Entity Vocabulary を使用する．本研究での mention ベクトル作成には NER の抽出結果を入力しているため，mention ベクトル作成時の多くの場合で未知 Entity に対するベクトル作成となり，効果的なベクトル作成が行われないと思われる．

3.5 単語間類似度タスクとの比較

同義語獲得タスクと単語間類似度タスクとの相関を確認するため，各分散表現作成手法について単語間類似度タスクを実施した．評価データは Sakaizawa and Komachi [13] の名詞に関するデータを用いた．評価尺度として Wang ら [14] が提案したランキング形式の尺度も用いた．本検証の詳細は付録 C に記載する．実験結果を表 4 に示す．

同義語獲得検証と単語間類似度実験の結果を比較すると，相関がほとんどないことがわかる．例えば同義語獲得タスクで比較的精度が低かった fasttext が単語間類似度タスクでは高い精度となっている．また，BERT についても同義語獲得タスクでは精度が高い結果であったが，単語間類似度タスクでは Hits@1 を除く指標で精度が低い結果となった．これらの結果はおそらく，同義語獲得タスクと単語間類似度タスクの入力単語の単位の違いによるものだと思われる．表 4 の下段の各指標ごとの相関から

表 5 同義語獲得タスクの出力例及びアノテーション例

term	コンパクトカセット	評価
編集距離	コンパクトカメラ	関連なし
fasttext	コンパクト	関連なし
BERT	カセットテープ	強い同義語
SimCSE	カセットテープ	強い同義語
BLINK	カセット	強い同義語
LUKE	意味シングルドライバ	関連なし

も，このタスク間で相関はほとんどないことが確認できる．このように，同義語獲得タスクは単語間類似度タスクと異なる性質を持つタスクと考えられる．

3.6 同義語獲得検証の具体例

表 5 に Amazon Review データの electronics カテゴリに含まれる例を示す．term として ‘コンパクトカセット’ を入力した場合，BERT と SimCSE は ‘カセットテープ’ を最も類似した mention として出力し，BLINK は ‘カセット’ を出力した．‘カセットテープ’，‘カセット’ のどちらについても，‘コンパクトカセット’ の定義文[5] $\text{Context}_{\text{コンパクトカセット}}$ と入れ替えた文 $\text{Context}_{\text{カセットテープ}}$ や $\text{Context}_{\text{カセット}}$ の意味が変わらないとして，互換性は満たすとアノテーションしている．これらの単語間では，上位概念として ‘オーディオ用磁気記録テープ’ を置くことができると考え，同位語性も満たしているとした．

一方で，編集距離と fasttext と LUKE の出力に関しては全て関連なしと評価した．この例から，編集距離や fasttext は表層的な類似は反映されるが，意味的な類似がうまく反映されていないと考えられる．

4 おわりに

本研究では，特定のドメインに依存した同義語辞書作成支援を目的として，ベースラインおよび評価用データセットの構築，人手評価を実施した．ベースラインとして，表層の特徴のみを使用する手法，単語単位の分散表現を拡張した手法，BERT やそれを派生した手法，Entity の分散表現を作成する手法を比較し，BERT 及びそれを派生した手法が優れた結果となった．また，実験の結果から単語間類似度タスクと同義語獲得タスクでは異なる性質を持つタスクであることがわかった．今後は，この単語間類似度タスクと同義語獲得タスクの違いを調査し，これらのタスクの異なる性質を明らかにしていく．

5） この定義文の詳細は付録 D に記載．

謝辞

本研究実施に当たって，株式会社レトリバの西鳥羽二郎様，木村大翼様には有益な助言をいただきました．この場を借りて深く御礼申し上げます．この成果は，国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務（JPNP18002）の結果得られたものです．

参考文献

[1] Hui Fang. A re-examination of query expansion using lexical resources. In **Proceedings of Annual Meeting of the Association for Computational Linguistics**, 2008.

[2] Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. Scalable zero-shot entity linking with dense entity retrieval. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing**, 2020.

[3] Francis Bond, Hitoshi Isahara, Kyoko Kanzaki, and Kiyotaka Uchimoto. Boot-strapping a wordnet using multiple existing wordnets. In **International Conference on Language Resources and Evaluation**, 2008.

[4] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In **Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence**, 2017.

[5] 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), 2020.

[6] Herbert Rubenstein and J. Goodenough. Contextual correlates of synonymy. **Communications of the ACM**, Vol. 8, pp. 627–633, 1965.

[7] Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. A study on similarity and relatedness using distributional and WordNet-based approaches. In **Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics**, 2009.

[8] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. **Transactions of the Association for Computational Linguistics**, Vol. 5, pp. 135–146, 2016.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, 2019.

[10] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, 2021.

[11] Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. LUKE: Deep contextualized entity representations with entity-aware self-attention. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing**, 2020.

[12] Vihari Piratla, Sunita Sarawagi, and Soumen Chakrabarti. Topic sensitive attention on generic corpora corrects sense bias in pretrained embeddings. In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**, 2019.

[13] Yuya Sakaizawa and Mamoru Komachi. Construction of a Japanese word similarity dataset. In **Proceedings of the Eleventh International Conference on Language Resources and Evaluation**, 2018.

[14] Bin Wang, C.-C. Jay Kuo, and Haizhou Li. Just rank: Rethinking evaluation with word and sentence similarities. In **Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics**, 2022.

[15] Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. The multilingual Amazon reviews corpus. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing**, 2020.

A　分散表現作成手法の詳細

fasttext　本検証で使用したモデルは，Wikipedia に対して MeCab[6] で分かち書きしたテキストで学習を行い作成した．term 及び mention の分散表現は，各形態素の分散表現について平均を取ることで作成した．

BERT　本研究では学習済み BERT[7] を用いた．term 及び mention の分散表現は BERT の各サブワードごとの出力を平均したベクトルを使用した．

SimCSE　本研究で扱う SimCSE は Unsupervised SimCSE で学習したモデルを使用した．具体的には，同義語候補である各 mention について，異なる Dropout を適用して BERT でエンコードしたベクトルを正例，バッチ中の異なる mention を負例として学習を行った．term 及び mention の分散表現作成方法は BERT と同様である．

BLINK　本検証では，Wu ら [2] で提案された手法のうち，Bi-Encoder 構造のみ用いている．この Bi-Encoder の学習には，日本語 BERT に対して fasttext と同じ Wikipedia を使用して行っている．また，彼らは Entity のベクトル表現作成の際に，‘[CLS] EntityTitle [ENT] 説明文 [SEP]’ といったように Entity の説明文を付与している．一方で，本研究で Entity に相当する term には説明文を与えていないので，term のベクトル作成時にはこの説明文は空白としている．また，mention のベクトル表現作成の際には，‘[CLS] 左側文脈 [Ms] mention [Me] 右側文脈 [SEP]’ といったように，文脈付きで行っている．こちらについては，本研究でも特定のドメインのコーパス中における mention の前後文脈を使用した．

LUKE　本研究では LUKE モデルとして，公開されている日本語モデル[8] を使用した．LUKE は Entity のベクトル作成を目的に BERT を拡張したモデルであり，Entity が出現する文脈も使用するため，mention については BLINK と同様に，mention が出現した文脈も LUKE に入力してベクトル化を行った．term については他の手法と同様に出現する文脈は想定していないため，その用語のみを入力しベクトル化を行った．

B　同義語獲得検証で使用したデータの詳細

本研究では特定のドメインのテキストコーパスとして Amazon Review データ，リコール文書，Twitter データを使用した．Amazon Review データは MARC [15] について，日本語データに対してカテゴリ単位で抽出したものを使用している．リコール文書はリコール情報公開サイト[9] から抽出し，登録されているカテゴリで区分けを行った．Twitter に関しては，COVID-19 日本語データセット[10] で意見・感想とアノテーションされている文書を使用した．

C　単語間類似度タスクの実験設定

本研究の単語間類似度タスクでは，Sakaizawa and Komachi [13] の名詞に関するデータを用いて行った．評価として，アノテーションされている人手評価との相関及び，Wang ら [14] によって提案されているランキング形式の尺度も用いた．Wang らは単語間類似度データセットの中で，類似度が高い上位 25%（5,514 ペア）を正例として定義している．彼らは単語間類似度データセットに含まれる全ての単語について，分散表現などを利用して類似度順に並べ，上位に正例を推薦できているか検証している．本研究でも同様の処理を行い，日本語名詞単語間類似度データセットのうち，276 ペアを正例として使用した．また，Wang らは単語間類似度に含まれる単語だけでなく，ランキングの候補単語として Wikipedia の頻度上位 20,000 単語を使用している．一方で，本研究では正例の数に合わせて日本語 Wikipedia の頻度上位 1,000 単語の名詞を使用した．なお，この名詞かどうかの判断に関しては UniDic （v3.1.0）[11] を用いた．

また，この検証での SimCSE はデータセット中に含まれる単語を利用して Unsupervised SimCSE の学習を行った．具体的には各単語について異なる Dropout を適用して BERT でエンコードしたベクトルを正例，バッチ中の異なる単語を負例として学習を行った．

D　同義語獲得検証の具体例の詳細

入力となる term の定義文を含む具体例を表 6 に示す．

表 6　定義文を含む同義語獲得タスクの出力例及びアノテーション例

term	コンパクトカセット	
定義文	コンパクトカセットは，オランダの 電機メーカーであるフィリップス社が、 フェライトを素に 1962 年に開発したオーディオ用磁気記録テープ媒体の規格である。	評価
編集距離	コンパクトカメラ	関連なし
fasttext	コンパクト	関連なし
BERT	カセットテープ	強い同義語
SimCSE	カセットテープ	強い同義語
BLINK	カセット	強い同義語
LUKE	意味シングルドライバ	関連なし

6）IPADIC を使用．
7）cl-tohoku/bert-base-japanese-v2 を使用．
8）studio-ousia/luke-japanese-base を使用．
9）国交省: https://www.mlit.go.jp/jidosha/carinf/rcl/index.html; 消費者庁: https://www.recall.caa.go.jp
10）https://www.db.info.gifu-u.ac.jp/covid-19-twitter-dataset/
11）https://github.com/polm/unidic-py