翻訳と BabelNet を利用した日本語の語義曖昧性解消

Ganbat Naranbuuvei
東京農工大学工学部

浅田宗磨
東京農工大学大学院

古宮嘉那子
東京農工大学大学院

{s203598x@st,s231157v@st,kkomiya@}.go.tuat.ac.jp

概要

本研究では、BabelNet の synset ID を語義ラベルに用いて日本語の語義曖昧性解消を行う。語義曖昧性解消では、あらかじめ正解ラベルとなる辞書の項目が与えられる。日本語では、分類語彙表の分類番号を利用することが多いが、多言語を扱うためには、BabelNet の多言語共通の synset ID を利用する必要がある。そこで、本研究では BabelNet の synsetID を語義ラベルに用いて、英語の語義つきコーパスから訓練した WSD モデルと、機械翻訳によって英語から和訳された語義つきデータで WSD モデルを学習し、それらを日本語のテストデータで評価して、比較と分析を行った。結果として、英語から学習したモデルが日本語のモデルより性能が高いことが分かった。また、日英のハイブリッドモデルを三つ提案した。

1 はじめに

語義曖昧性解消 (Word Sense Disambiguation, WSD) とは、自然言語処理 (NLP) の分野において、ある単語が持つ複数の意味の中から、文脈に応じて最も適切な意味を特定するタスクである。例えば、「きれい」という単語には「外観が愉快な」あるいは、「清潔な」などの意味があり、機械学習において、対象単語の周囲の単語の品詞や単語同士の共起関係などを文脈情報として語義を推定する。語義を正しく推定することは機械翻訳、質問応答などのタスクにも繋がる。

WSD は諸言語において研究されているが、英語では、一般的に WordNet[1] の synset ID が語義ラベルとして利用される。日本語 WordNet[2] もあるが、英語の WordNet を翻訳したものであり、誤りを含んでいる。そのため、日本語の WSD には、語義ラベルとして分類語彙表 [3] を利用することが多い。

多言語の WSD を行うには、多言語共通の語義ラベルが必要である。本研究では、多言語の WSD を目指し、まずは多言語共通の語義ラベルを利用した日本語の WSD を行う。多言語の WordNet 型の概念辞書 BabelNet[4] がある。BabelNet には WordNet のように、1 つの概念を表現する synset があり、それぞれ言語共通の ID を持っている。例えば、日本語の単語「銀行」と英語の単語「bank」は同じ概念を示しているため、同じ synset ID を持つ。

BabelNet の synset ID が付与されている日本語コーパスの量は限られている。そのため、本研究では、英語の語義ラベルつきコーパスから訓練した WSD モデルと、機械翻訳によって英語から和訳された語義ラベルつきデータで WSD モデルを学習し、それらを日本語のテストデータで評価して、比較と分析することを目的とする。また、これらの性質を利用した、日英のハイブリッドモデルを提案する。

2 関連研究

BabelNet の synset ID を利用した日本語の WSD の研究には Pasini ら [5] の研究がある。この研究では、英語の WSD で最もよく使われる SemCor コーパス[1)] と WNG コーパス (Princeton WordNet Gloss Corpus)[2)] を機械翻訳し、語義ラベルをつけたデータで東北大学が公開した日本語 BERT モデルを fine-tuning して、WSD を行っている。fine-tuning されたモデルは日本語 WordNet の例文から作成されたテストデータで評価している。彼らは、事前学習済みの多言語モデルを英語で学習し、日本語で評価する zero-shot 設定でも実験を行っている。

BERT を用いた英語の WSD に、Jiaju ら [6] の研究があり、WSD で BERT を encoder として利用することの有効性を示した。日本語の WSD には、Cao ら [7] の研究がある。この研究では、京都大学が公開した日本語 BERT の feature-based を用いて日本語の

1） https://sapienzanlp.github.io/xl-wsd/docs/data/
2） https://sapienzanlp.github.io/xl-wsd/docs/data/

50 個の単語に対して lexical sample task で WSD を行い、BERT が日本語 WSD にも有効であることを示した。また、日本語歴史コーパスの WSD の研究には Asada らの [8] がある。この研究では、現代文で訓練された BERT を古文で fine-tuning することで、古文の all-words WSD を行っている。

日本語 WordNet に関する平尾ら [9] の研究がある。日本語 WordNet には 5 %程の不整合が含まれているという報告があり、この研究では、日本語 WordNet のミスを分類し、それらを機械的に抽出する手法を提案し、実験を行っている。

3 データ

3.1 コーパス

本研究では、Pasini ら [5] が公開した言語横断的なコーパス、XL-WSD を用いた。XL-WSD は英語と日本語を含め、18 つの言語の WSD の gold テストデータと韓国語と中国語以外の言語の silver トレーニングデータからなる。BabelNet の多言語共通の語義ラベルを利用することで、言語横断の WSD が可能である。本研究では、公開されたコーパスの英語と日本語コーパスを用いて、日本語の WSD を行う。

3.2 日本語の実験データ

日本語の学習データ、検証データ、テストデータは、Pasini ら [5] による XL-WSD データを利用した。学習データは SemCor と WNG コーパスを英語から日本語に翻訳したコーパスである。検証データとテストデータは日本語 WordNet の例文から作成されており、1 文に 1 つの対象単語を含むため、対象単語数は例文数と等しい。検証データとテストデータは、対象単語の語義ラベルを日本語 WordNet から英語 WordNet へマッピングし、さらに BabelNet へマッピングすることで作成されている。データの詳細を表 1 に示す。

表 1 日本語の実験データの詳細

	学習データ	検証データ	テストデータ
対象単語数	23,217	1,901	7,602
語義種類数	1,141	1,755	5,964

3.3 英語の実験データ

Pasini ら [5] にならい、英語の学習データには SemCor と WNG コーパスを利用し、検証データには SemEval-07[10] を利用した。その詳細は表 2 の通りである。

表 2 英語の学習データと検証データの詳細

	学習データ	検証データ
対象単語数	840,471	455
語義種類数	117,653	361

本研究は日本語の WSD であるため、そのまま英語データで学習したモデルを利用することはできない。そのため、英語モデルのテストには日本語のテストデータを翻訳して利用した。翻訳には ChatGPT の GPT-4[11][3] と DeepL をそれぞれ利用して比較する。

4 語義曖昧性解消モデル

本研究では、ベースラインとなる英語 WSD モデルと日本語 WSD モデルを作成し、それらを合わせたハイブリッドモデルをいくつか提案する。WSD モデルの作成には Google 社の提供する英語 BERT[12] と東北大学が公開した日本語 BERT[4] を用いた。英語と日本語のモデルはそれぞれ fine-tuning して WSD を行った。BERT モデルは、WSD の対象単語ごとに語義を付与する系列ラベリングタスクとして設計した。fine-tuning を行う際に、対象単語の候補となる語義ラベルの集合に注目し、その中で、softmax 関数の出力が最も高いものを正解として学習及び推論を行った。

4.1 日本語の WSD モデル

日本語のモデルに関しては、東北大学が公開した日本語 BERT モデルの cl-tohoku/bert-base-japanese-v3 を事前学習済みモデルとして利用し、SemCor コーパスと WNG コーパスの語義ラベル付きの日本語翻訳コーパス [5] で fine-tuning をして WSD を行った。Tokenizer は、日本語 BERT に付属する MeCab 版の tokenizer を利用した。

4.2 英語の WSD モデル

英語の WSD モデルの訓練には、BERT モデルの bert-base-uncased を事前学習済みモデルとして利用し、語義ラベルつき SemCor コーパスと WNG コーパスで fine-tuning を行った。英語のモデルを日本語のデータで評価するために、日本語のテストデー

3) https://openai.com/research/gpt-4
4) https://huggingface.co/cl-tohoku/bert-base-japanese-v3

タを ChatGPT と DeepL で英語に翻訳した。この際、WSD の対象単語は二重引用符で囲むこと（“x”）で指定した。ChatGPT では、プロンプトとして「以下の文を英語に翻訳してください。その際に、二重引用符で囲まれている単語の翻訳語も二重引用符で囲んでください。」と与えた。しかし、翻訳した後に、テストデータとして利用できない例文があった。(1) 翻訳によって WSD の対象単語の特定が出来なかったことと (2) 対象単語を特定しても、翻訳の質により正解を出せないことに原因がある。

例えば、「“初演” は好評を博した.」という日本語のテスト事例では「初演」が WSD の対象単語である。この際 (1) 英訳では「初演」に対する英訳に二重引用符がつかないことがある。別の意味の単語についたり、そもそも二重引用符を含まない訳を返すことがあるためである。こういう場合には、翻訳によって WSD の対象単語の特定が出来ず、結果として英語モデルがうまく働かない。この処理で対象単語を特定できた例文は 7,602 文中、ChatGPT では 7,219 件と DeepL では 6,314 件であった。また、この例文を ChatGPT で翻訳すると「The “debut” was well-received.」になる。英語モデルは「debut」が持つ synset ID の集合の中から答えを探すが、実はこの中には、「初演」の正解の synset ID が含まれない。これが (2) 対象単語を特定しても、翻訳の質により正解を出せない例である。正解をモデルに与えることはできないが、「debut」が持つ synset ID の集合と、「初演」の持つ synset ID の集合に重なりがない場合には、英語モデルでは正解が出せないことが明らかである。こうして推定した、英語モデルで正解を出す可能性がある例文に絞ると、ChatGPT では 5,433 件で DeepL では 4,820 件であった。

4.3 単純日英ハイブリッドモデル

単純日英ハイブリッドモデルは、英語のモデルで解ける例文を英語のモデルを使用し、それ以外の例文は日本語のモデルを使用する手法である。具体的には、ChatGPT と DeepL で翻訳した際に、対象語の英訳（二重引用符で囲まれた単語）のもつ語義候補（synset ID の集合）と、もともとの日本語の WSD の対象語のもつ語義候補（synset ID の集合）に重複がない場合には日本語モデルを利用する。

4.4 Lemma 推定による日英ハイブリッドモデル

Lemma 推定による日英ハイブリッドモデルは英語のモデルで正解を出す可能性がない例文の対象単語を英語のモデルで評価できるように lemma を自動的に推定する手法である。日本語のテストデータを ChatGPT と DeepL で英語に翻訳した際に、対象単語の訳語の語義候補に日本語のその単語の語義候補と一致するものが存在しない場合、英語の単語リストから日本語の語義候補を持つ lemma を探して、対象単語をその lemma で書き換えている。こうすることで、WSD の対象単語を特定出来なかった例文以外の英語に翻訳された例文を英語のモデルで評価できるようになる。

例えば、先ほどの例文「“初演” は好評を博した.」の対象単語「初演」の語義候補を持つ英語を調べると、「premiere」という単語が該当する。翻訳例文の対象単語をこの単語で書き換える。この例では、書き換えた例文は「The “premiere” was well-received.」になり、英語モデルで正解できる可能性がある。正解ラベルを持つ単語が複数ある場合、最初に出現した単語に決める。ただし、語義候補を持つ単語なので、正解ラベルの語義を持つとは限らない。なお、文中に引用符がない文については日本語モデルを利用した。

4.5 対象単語修正による日英ハイブリッドモデル

ChatGPT で翻訳した際に、WSD の対象単語が変わってしまう問題があった。例えば、「彼は “出口” を閉鎖した.」の ChatGPT の翻訳は「He “closed” the exit.」であり、対象単語であるはずの「exit」ではなく「closed」に二重引用符がついた形で訳されてしまった。これを回避する手法として、対象単語修正による日英ハイブリッドモデルを提案する。本手法では二重引用符の代わりに、ChatGPT で直接対象単語を推定し、指定する。

例えば、上述の例では、ChatGPT に WSD の対象単語である「出口」の可能な翻訳を出力させ、[exit, way out] を得る。これらの単語を翻訳文から探し、見つかった場合、対象単語をその単語にする。この例の場合には訳文に「exit」を含むため、exit を対象語として英語の WSD を行う。なお、対象単語を直接推定するのは対象語の英訳のもつ語義候補と、もともとの日本語の WSD の対象語のもつ語義候補に重複がない場合のみとした。この処理を行った場

合、7,602 文中 5,803 件について英語モデルで答えを返すことができた。これ以外の文については、日本語モデルを利用した。

5 実験

本研究では、英語 BERT の bert-base-uncased と東北大学が公開した日本語 BERT をそれぞれ fine-tuning することで英語と日本語の WSD モデルを作成する。実験は、1) 英語モデル、2) 日本語モデル、3) 単純日英ハイブリッドモデル、4)Lemma 推定による日英ハイブリッドモデル、5) 対象単語修正による日英ハイブリッドモデルの五種類を行った。1) では、英語のモデルで評価できない例文の回答を不正解とみなして正解率を求めた。

5.1 実験設定

英語と日本語モデルは、エポック数は [5,10,15]、バッチサイズは [4,8,16]、学習率は [2e-6,2e-5,2e-4] の設定でグリッドサーチを行い、検証データにおいて最も正解率の高いものを採用した。学習時に学習データはをランダムにシャッフルし、最適化関数に Adam、損失関数にクロスエントロピー誤差を用いた。

5.2 評価手法

本研究では、正解率を評価手法として用いた。語義ラベルを正しく当てた回数を全対象語数に割って正解率を求める。ハイブリッドモデルの正解率は以下の式で求めた。

$$\text{hybrid モデルの正解率} = \frac{num_e + num_j}{\text{全対象単語数}}$$

num_e は英語モデルの正解数、num_j は日本語モデルの正解数を表す。

6 実験結果

日本語 WSD の正解率を表 3 に示す。また、ハイブリッドモデルの言語ごとの例文数と正解率を表 4 に示す。なお、表中における「単純」は単純日英ハイブリッドモデル、「Lemma 推定」は Lemma 推定による日英ハイブリッドモデル、「対象単語修正」は対象単語修正による日英ハイブリッドモデルを表す。

7 考察

表 3, 4 から英語モデルは日本語モデルより性能が高いことが分かる。原因として、日本語の学習データが英語から翻訳したものであるため、誤訳や不自然な語を含んでいる場合があること、また、日本語の学習データが英語より少ないことが考えられる。

表 3 日本語 WSD の正解率

モデル	ChatGPT	DeepL
英語モデル	50.89%	44.76%
日本語モデル	49.38%	
単純	66.82%	64.81%
Lemma 推定	64.77%	63.04%
対象単語修正	**67.60**%	-

表 4 ハイブリッドモデルの言語ごとの例文数と正解率

		ChatGPT	DeepL
単純	英語	5,433 (71.21%)	4,820 (70.59%)
	日本語	2,169 (55.83%)	2,782 (54.82%)
Lemma 推定	英語	7,219 (65.73%)	6,314 (65.85%)
	日本語	383 (46.74%)	1,288 (49.15%)
対象単語修正	英語	5,803 (70.52%)	
	日本語	1,799 (58.20%)	

また、ハイブリッドモデルは英語モデルよりも性能が高い。中でも、ChatGPT の翻訳は DeepL より正解率が高かった。これは、ChatGPT にはユーザがプロンプトを与えられるため、条件にあった出力を多く得られ、結果として、ChatGPT の方が DeepL よりも英語モデルで評価できる例文数が多くなったためであると考えられる。しかし、それでも名詞を動詞にして意訳する場合などがあり、WSD の対象単語の訳語が得られない場合もあった。そのため、対象単語修正による日英ハイブリッドモデルを利用した場合でも、英語モデルでテストできない例文があった。今後は、その点を含めより深く分析したい。

8 おわりに

本研究では、BabelNet の synset ID を用いた日本語の WSD を行った。英語と日本語の WSD モデルを比較したところ、英語モデルの方が日本語モデルよりも正解率が高かった。また、単純日英ハイブリッドモデル、Lemma 推定による日英ハイブリッドモデル、対象単語修正による日英ハイブリッドモデルを提案し比較したところ、対象単語修正によるモデルが最も良い正解率を示した。

謝辞

本研究は、科研費 22K12145 の助成を受けたものです。

参考文献

[1] George A. Miller. WordNet: A lexical database for English. In **Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992**, 1992.

[2] Francis Bond, Hitoshi Isahara, Sanae Fujita, Kiyotaka Uchimoto, Takayuki Kuribayashi, and Kyoko Kanzaki. Enhancing the Japanese WordNet. In Hammam Riza and Virach Sornlertlamvanich, editors, **Proceedings of the 7th Workshop on Asian Language Resources (ALR7)**, pp. 1–8, Suntec, Singapore, August 2009. Association for Computational Linguistics.

[3] 国立国語研究所. 『分類語彙表増補改訂版データベース』. https://clrd.ninjal.ac.jp/goihyo.html. 2024 年 1 月 3 日確認.

[4] Roberto Navigli and Simone Paolo Ponzetto. BabelNet: Building a very large multilingual semantic network. In Jan Hajič, Sandra Carberry, Stephen Clark, and Joakim Nivre, editors, **Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics**, pp. 216–225, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[5] Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. Xl-wsd: An extra-large and cross-lingual evaluation framework for word sense disambiguation. **Proceedings of the AAAI Conference on Artificial Intelligence**, Vol. 35, No. 15, pp. 13648–13656, May 2021.

[6] Jiaju Du, Fanchao Qi, and Maosong Sun. Using bert for word sense disambiguation, 2019.

[7] Cao Rui, Tanaka Hirotaka, Bai Jing, Ma Wen, and Shinnou Hiroyuki. Word sense disambiguation using supervised learning with bert. **Proceedings of Language Resources Workshop**, Vol. 4, pp. 273–279, 2019.

[8] Shoma Asada, Kanako Komiya, and Masayuki Asahara. All-words word sense disambiguation for historical japanese. In **Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation**, Hong Kong, December 2023. Association for Computational Linguistics.

[9] Takuya Hirao, Takahiko Suzuki, Kouki Miyata, Koki Miyata, and Sachio Hirokawa. Detection of inconsistency in japanese wordnet. **IPSJ SIG Technical Peport**, pp. 1–5, 2012.

[10] Roberto Navigli, Kenneth C. Litkowski, and Orin Hargraves. Semeval-2007 task 07: Coarse-grained english all-words task. In **Proceedings of the 4th International Workshop on Semantic Evaluations**, SemEval '07, p. 30–35, USA, 2007. Association for Computational Linguistics.

[11] OpenAI. Gpt-4 technical report. **ArXiv**, Vol. abs/2303.08774, , 2023.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.