

半教師あり学習を用いた単語アライメントの改善

苗 中濤¹ 永田 昌明² 鶴岡 慶雅¹

¹ 東京大学大学院情報理工学系研究科電子情報学専攻

² NTT コミュニケーション科学基礎研究所

{miao,tsuruoka}@logos.t.u-tokyo.ac.jp masaaki.nagata@ntt.com

概要

単語アライメントは自然言語処理の重要な基礎タスクである。既存の手法はほとんど Transformer エンコーダモデルに基づいている。本研究では、Transformer デコーダモデルに基づく単語アライナーを提案する。また、翻訳文ペアをラベルなしデータとして活用し、Transformer エンコーダベースおよび Transformer デコーダベースの単語アライナーの両方に適用可能な単語アライメントのための半教師あり学習手法を提案する。実験結果は、数万の翻訳文ペアを用いた提案手法が、単語アライメントデータセットにおいて、現在の最先端手法を上回ることを示している。

1 はじめに

単語アライメントとは、翻訳文のペアにおいて、ソース文とターゲット文の間で互いに翻訳になっている単語の対を見つけるタスクである。このタスクは、当初、統計的機械翻訳を改善するために提案された [1]。単語アライメントや深層学習技術の発展に伴い、単語アライメントは自然言語処理 (NLP) のさまざまな分野に応用されている [2, 3, 4]。

単語アライメントを改善する代表的な研究に、SpanAlign [5]、WSPAlign [6]、BinaryAlign [7] がある。SpanAlign [5] は、単語アライメントを SQuAD 形式のスパン予測質問応答タスクとして訓練する。WSPAlign [6] は、SpanAlign における手動でラベル付けされたデータという要件を緩和し、大規模な弱教師あり事前学習データセット (Wikipedia のエンティティリンクと多言語 Transformer エンコーダ言語モデルによる文脈化単語埋め込みに基づく約 200 万のノイズを含む段落ペア) を活用する。BinaryAlign [7] は、単語アライメントをバイナリ系列ラベリングタスクとして再定義し、現時点での最高精度を達成している。これらの単語アライメント手法は、(1) 手

動でラベル付けされたデータセット、またはラベル付けされたデータの厳格な要件を緩和するための大規模な弱い教師情報付き事前訓練データ、および (2) 多言語 Transformer エンコーダモデルに大きく依存している。

そこで本研究では、Transformer デコーダに基づく多言語大規模言語モデル (LLM) が単語アライメントにおいてどの程度効果的であるかを探求する。また翻訳文ペアをラベルなしデータとして活用し、Transformer エンコーダベースおよび Transformer デコーダベースのモデル両方に適用可能な単語アライメントのための半教師あり学習手法を提案する。具体的には、少量の翻訳データ (1 言語ペアあたり平均約 35,000 の翻訳文ペア) を用いた半教師あり学習手法を利用することで、BinaryAlign と多言語 LLM ベースの単語アライナーの性能をさらに向上させ、最先端の性能を達成することができた。

2 関連研究

多言語 Transformer エンコーダモデルに基づいて、単語アライメントを改善するための様々な方法が提案されている。多言語 Transformer エンコーダベースの言語モデルにおける単語アライメントの性能向上に焦点を当てた研究は、大きく 2 つにわけることができる。ひとつは、エンコーダモデルから抽出された文脈化単語埋め込みに基づく手法である [8, 9, 10]。代表的な手法としては、SimAlign [8]、AccAlign [10] と AwesomeAlign [9] がある。SimAlign は、多言語事前学習済みエンコーダーモデルの文脈化埋め込みを使用して単語アライメントを取得することを提案する。AccAlign は、mBERT [11] や XLM-R [12] などの多言語モデルを使用せず、代わりに多言語文埋め込みモデルである LaBSE [13] を活用して、単語アライメントモデルとしての可能性を探る。AwesomeAlign は翻訳データを訓練データとして使用し、単語アライメントモデルを訓練するため

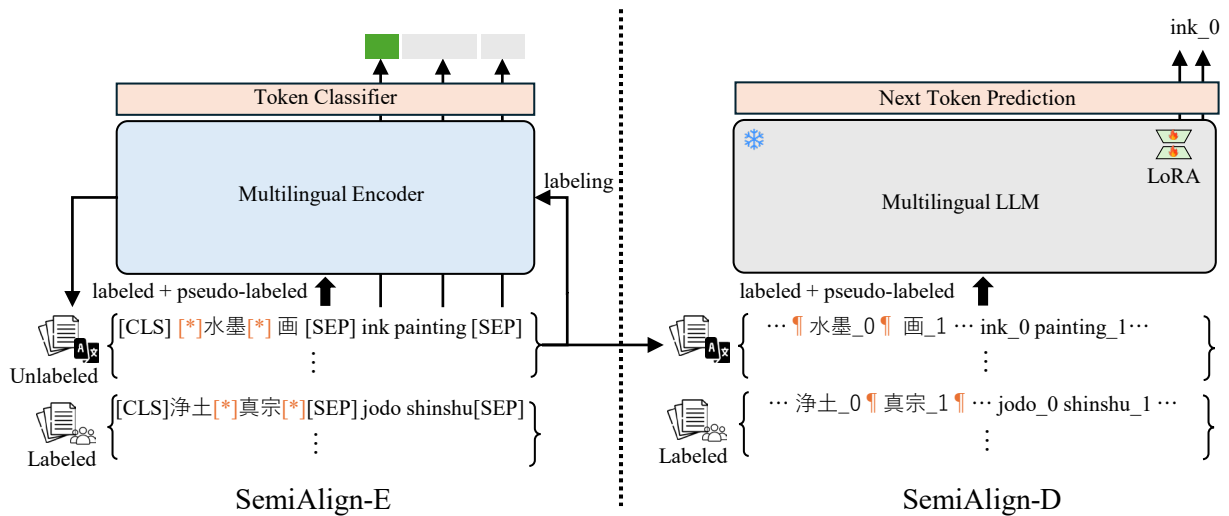


図1 提案する半教師付きフレームワーク。左の図には、自己訓練による多言語 Transformer エンコーダモデルの学習プロセスを示す。右の図には、LoRA と Transformer エンコーダモデルが予測した擬似ラベル付きデータとラベル付きデータの組み合わせを用いた多言語 LLM の学習過程を示す。

に5つの異なる学習オブジェクティブを設計する。本研究と最も近い手法は AwesomeAlign である。どちらも翻訳データを使用して単語アライメントの性能を改善する。もう1つのアプローチは、単語アライメントの教師あり学習である。代表的な手法として、SpanAlign [5]、WSPAlign [6]、BinaryAlign [7] などがある。SpanAlign は、単語アライメントタスクをスパン予測による QA タスクとして再定義する。WSPAlign は、ウィキペディアのエンティティリンクと Transformer エンコーダモデルの文脈化単語埋め込みに基づいて、約 200 万ペアの大規模な弱い教師情報付きデータセットを構築する。BinaryAlign は、単語アライメントをバイナリ系列ラベリングタスクとして再定式化し、ラベル付きデータで現時点での最高精度を達成している。どちらのカテゴリーも、多言語 Transformer エンコーダーの言語モデルに基づいている。本研究では、自己回帰の Transformer デコーダーの LLM が多くの自然言語処理タスクで素晴らしい性能を達成していること [14, 15] を考慮し、多言語 LLM を単語アライナーとして使用する可能性を探る。

3 提案手法

背景 原文 $\mathbf{x} = [x_1, \dots, x_n]$ と翻訳文 $\mathbf{y} = [y_1, \dots, y_m]$ が与えられた場合、単語アライメントは、原文と翻訳文の間で互いに翻訳になっている単語の対を見つけることを目的とする。単語アライメントのラベルは、原文と翻訳文の間で互いに

翻訳になっている単語の対の集合である：

$$\mathcal{A} = \{(x_i, y_j) : x_i \in \mathbf{x}, y_j \in \mathbf{y}\}. \quad (1)$$

x_i と y_i は、単語アライメントが位置情報を考慮するため、通常、原文と翻訳文の単語の位置として表現される。いくつかの単語アライメントデータセットでは、ペアの集合 \mathcal{A} には可能なアライメント (\mathcal{P}) と確実なアライメント (\mathcal{S}) の2種類がある。

3.1 ラベルなしデータの収集

ウィキペディアのエンティティ単語に対してウィキペディアのエンティティハイパーリンクを使って、一般的な単語に対して、文脈化単語埋め込みの類似度を弱い単語アライメント信号として使用する WSPAlign とは異なり、本研究では弱い信号を含まない翻訳文ペアを利用する。

生の翻訳文ペアが与えられたら、さらに訓練を進めるためにいくつかの前処理を行う必要がある。最初のステップはトークン化である。第2のステップでは、原文が小文字の場合、翻訳文も小文字に変換する。中国語と日本語の文字については、このステップをスキップする。

3.2 多言語エンコーダ：自己訓練

Transformer エンコーダーモデルに基づいた単語アライメントモデルを改善するために、ウィキペディアのエンティティリンクから弱い教師信号を使う代わりに、自己訓練を利用する。ラベルなしデータ

の前処理後、BinaryAlign を用いて、ラベルなしデータの擬似ラベル \mathcal{D}_u を予測する。擬似ラベルデータを \mathcal{D}_p とする。擬似ラベル付きデータ \mathcal{D}_p とラベル付きデータ \mathcal{D}_l を組み合わせて、バイナリ系列ラベリングの学習損失で Transformer エンコーダのモデルを再学習する。図 1 の左のサブ図に学習の過程を示す。

3.3 多言語 LLM：単語アライナーとしての可能性を探る

このセクションでは、まずラベル付きデータで学習した LLM の性能を評価し、バニラ LLM ベースの単語アライナーの有効性について考察する。次に、翻訳文ペアを使用することで、LLM ベースの単語アライナーの性能がさらに向上するかどうかを確認する。LLM ベースの単語アライナーでラベルを予測するのは時間がかかるため、Transformer エンコーダモデルを利用して擬似ラベルを予測する。

生成の粒度をコントロールする デコーダベースの LLM では、単語アライメントを位置情報を考慮した生成タスクとして定義する必要がある。また、Transformer デコーダベースの LLM に翻訳ペアを入力する際には、生成の粒度を考慮する必要がある。デコーダベースの LLM における単語アライメントに最適な生成粒度は、まだ検討されていない。本研究では、2 つのレベルの粒度を考えている：

- フルモード（文レベル）：原文と翻訳文を入力し、すべての対応された単語を一度に出力させる。
- マーカーモード（単語レベル）：まず原文の 1 つの単語を特別なトークンでマークする。このマークされた単語に対応する翻訳文の単語とその位置を出力させる。

実験で、両方の効果を検証した。比較すると、マーカーモードの方が効果的なので、本研究では主にマーカーモードを使用する。フルモードとマーカーモードの具体的なプロンプト例は、付録 A に記載されている。

アラインメントの対称化 マーカーモードの生成では、原文の単語を 1 つマークし、モデルに翻訳文中の対応する単語とその位置を予測させる。この予測は逆方向にも実行できる。マーカーモードを使うとき、SpanAlign [5] と WSPAlign [6] に従って、アラインメントの対称化を行う。具体的に言うと、マーカーモードでは、両方向からの予測を取得し、それ

らを組み合わせて最終的な予測（ユニオン）とする。

4 実験

4.1 データセット

主に 4 つの言語ペアにおいて提案手法の性能を評価する：日本語-英語（Ja-En）、ドイツ語-英語（De-En）、ルーマニア語-英語（Ro-En）、中国語-英語（Zh-En）。

ラベル付きデータ 日英単語アライメントについて、The Kyoto Free Translation Task (KFTT) 単語アライメントデータセット [16] を用いる。ルーマニア語-英語単語アライメントデータセットは [17] によって提供されている。ドイツ語-英語単語アライメントデータセットは [18] によって提供されている。中国語-英語単語アライメントデータセットは TsinghuaAligner のウェブサイト¹⁾から入手する。BinaryAlign に従って、v1 版を使用する。

ラベルなしデータ 日英については、KFTT 翻訳データのサブセット [16] を使用する。ルーマニア語-英語については、[18] の訓練用翻訳コーパスを使用する。ドイツ語と英語のペアでは、News Commentary v18.1 [19] のサブセットをラベルなしデータとして使用する。中英対には、News Commentary v16 データセットを使用する。

表 1 にラベル付きデータの統計情報を示す。表 2 にラベルなしデータの統計情報を示す。

Lang	# Train	# Val	# Test
Ja-En	653	225	357
De-En	300	-	208
Ro-En	150	-	98
Zh-En	450	-	450

表 1 ラベル付きデータにおける訓練、検証、テスト例の数。

Lang	Max Num
Ja-En	52,240
De-En	30,000
Ro-En	30,000
Zh-En	36,000

表 2 ラベルなしデータのサイズ。

1) <https://nlp.csai.tsinghua.edu.cn/~ly/systems/TsinghuaAligner/TsinghuaAligner.html>

4.2 Fine-tuning

ベースモデル Transformer エンコーダのベースモデルとして、mDeBERTa-v3-base²⁾ [20] を使用する。Transformer デコーダの LLM は、主に Llama-3.1-8B-Instruct [21] をベースモデルとしている。唯一の例外はルーマニア語-英語ペアである。このペアでは aya-expanse-8b³⁾ を使う。LLM の訓練に LoRA [22] を利用する。ランクパラメータは 64 に設定されている。スケーリングのアルファパラメータは 256 に設定する。LoRA レイヤーのドロップアウト確率を 0.05 に設定する。

Lang	Method	Precision (%)	Recall (%)	F1 (%) ↑	AER (%) ↓
Ja-En	SpanAlign [5]	77.3	78.0	77.6	22.4
	WSPAlign [6]	81.6	85.9	83.7	16.3
	BinaryAlign* [7]	87.74	84.46	86.07	13.93
	Llama3.1 Inst + SFT	78.65	89.76	83.84	16.16
	SemiAlign-E (ours)	88.34	86.91	87.62	12.38
	SemiAlign-D (ours)	88.09	88.63	88.36	11.64
De-En	SpanAlign [5]	89.9	81.7	85.6	14.4
	WSPAlign [6]	90.7	87.1	88.9	11.1
	BinaryAlign* [7]	93.79	90.73	92.23	7.74
	Llama3.1 Inst + SFT	85.46	90.98	88.14	11.93
	SemiAlign-E (ours)	94.55	90.93	92.70	7.27
	SemiAlign-D (ours)	93.97	90.69	92.31	7.66
Ro-En	SpanAlign [5]	90.4	85.3	86.7	12.2
	WSPAlign [6]	92.0	90.9	91.4	8.6
	BinaryAlign* [7]	92.67	92.51	92.59	7.41
	Aya Expanse + SFT	80.37	87.72	83.89	16.11
	SemiAlign-E (ours)	93.55	92.68	93.11	6.89
	SemiAlign-D (ours)	94.65	89.52	92.02	7.98
Zh-En	SpanAlign [5]	-	-	-	8.9
	WSPAlign [6]	-	-	-	7.6
	BinaryAlign* [7]	95.63	94.13	94.87	5.12
	Llama3.1 Inst + SFT	92.60	96.00	94.27	5.73
	SemiAlign-E (ours)	95.57	95.14	95.35	4.65
	SemiAlign-D (ours)	96.18	95.13	95.65	4.35
Avg.	SpanAlign [5]	-	-	-	14.48
	WSPAlign [6]	-	-	-	10.9
	BinaryAlign* [7]	92.46	90.46	91.44	8.55
	LLM + SFT	84.27	91.12	87.54	12.48
	SemiAlign (ours)	93.10	91.85	92.46	7.54

表 3 提案手法とベースラインの評価結果。「LLM + SFT」は、すべての言語ペアに対するラベル付き単語アライメントデータセットを使用した教師あり微調整の平均結果である。BinaryAlign*は、BinaryAlign の再現結果である。Ja-En、De-En、および Ro-En における SpanAlign と WSPAlign の結果は、WSPAlign からのものである。SpanAlign および WSPAlign の Zh-En における結果は、BinaryAlign から得られたものである。

4.3 評価方法

全ての実験において、Precision(P)、Recall(R)、F1、Alignment Error Rate (AER) [23] を用いて、単語アラ

イメントモデルを評価する。予測アライメントの集合 (\mathcal{A})、確実アライメントの集合 (\mathcal{S})、可能アライメントの集合 (\mathcal{P}) が与えられたとき、Precision, Recall, F1, AER は以下のように計算される：

$$Precision(\mathcal{A}, \mathcal{P}) = \frac{|\mathcal{A} \cap \mathcal{P}|}{|\mathcal{A}|} \quad (2)$$

$$Recall(\mathcal{A}, \mathcal{S}) = \frac{|\mathcal{A} \cap \mathcal{S}|}{|\mathcal{S}|} \quad (3)$$

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (4)$$

$$AER(\mathcal{A}, \mathcal{S}, \mathcal{P}) = 1 - \frac{|\mathcal{A} \cap \mathcal{S}| + |\mathcal{A} \cap \mathcal{P}|}{|\mathcal{A}| + |\mathcal{S}|} \quad (5)$$

4.4 結果

表 3 に実験結果を示す。実験結果から、提案手法が最先端の性能を達成していることがわかる。日英単語アライメントデータセットで最も大きな改善が得られることがわかる。Transformer エンコーダベースのモデルの場合、BinaryAlign と比較して、F1 スコアが 86.07 から 87.62 に向上した。少量のラベルなしデータを活用することで BinaryAlign のパフォーマンスが向上したことがわかる。Transformer デコーダーベースの LLM の場合、SFT ベースラインと比較して、提案手法により F1 スコアが 83.84 から 88.36 へ向上した。多言語 LLM ベースの単語アライナーに対して、ラベル付きデータのみを使用するのは不十分であることがわかる。

5 おわりに

本研究では、Transformer デコーダに基づく単語アライナーを提案する。また、翻訳文ペアをラベルなしデータとして活用し、Transformer エンコーダベースおよび Transformer デコーダベースの単語アライナーの両方に適用可能な単語アライメントのための半教師あり学習手法を提案する。現在の最先端手法である BinaryAlign が、数万のラベルなしデータを用いた半教師あり学習により、さらに改善できることを発見した。また、Transformer エンコーダベースの単語アライメントモデルから得られた擬似ラベル付きデータを、多言語 LLM ベースの単語アライナーの改善に活用した。実験結果は、数万の翻訳文ペアを用いた提案手法が、単語アライメントデータセットにおいて、現在の最先端手法を上回ることを示している。

2) <https://huggingface.co/microsoft/mdeberta-v3-base>

3) <https://huggingface.co/CohereForAI/aya-expanse-8b>

参考文献

- [1] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. **Computational Linguistics**, Vol. 19, No. 2, pp. 263–311, 1993.
- [2] Zewen Chi, Li Dong, Bo Zheng, Shaohan Huang, Xian-Ling Mao, Heyan Huang, and Furu Wei. Improving pretrained cross-lingual language models via self-labeled word alignment. In **ACL**, pp. 3418–3430, 2021.
- [3] Zhongtao Miao, Qiyu Wu, Kaiyan Zhao, Zilong Wu, and Yoshimasa Tsuruoka. Enhancing cross-lingual sentence embedding for low-resource languages with word alignment. In **Findings of NAACL**, pp. 3225–3236, 2024.
- [4] Qiyu Wu, Masaaki Nagata, Zhongtao Miao, and Yoshimasa Tsuruoka. Word alignment as preference for machine translation. In **EMNLP**, pp. 3223–3239, 2024.
- [5] Masaaki Nagata, Katsuki Chousa, and Masaaki Nishino. A supervised word alignment method based on cross-language span prediction using multilingual BERT. In **EMNLP**, pp. 555–565, 2020.
- [6] Qiyu Wu, Masaaki Nagata, and Yoshimasa Tsuruoka. WSPAlign: Word alignment pre-training via large-scale weakly supervised span prediction. In **ACL**, pp. 11084–11099, 2023.
- [7] Gaetan Latouche, Marc-André Carbonneau, and Benjamin Swanson. BinaryAlign: Word alignment as binary sequence labeling. In **ACL**, pp. 10277–10288, 2024.
- [8] Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In **Findings of EMNLP**, pp. 1627–1643, 2020.
- [9] Zi-Yi Dou and Graham Neubig. Word alignment by fine-tuning embeddings on parallel corpora. In **EACL**, pp. 2112–2128, 2021.
- [10] Weikang Wang, Guanhua Chen, Hanqing Wang, Yue Han, and Yun Chen. Multilingual sentence transformer as a multilingual word aligner. In **Findings of EMNLP**, pp. 2952–2963, 2022.
- [11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **NAACL**, pp. 4171–4186, 2019.
- [12] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In **ACL**, pp. 8440–8451, 2020.
- [13] Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. Language-agnostic BERT sentence embedding. In **ACL**, pp. 878–891, 2022.
- [14] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Nee-lakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, **Advances in Neural Information Processing Systems**, Vol. 33, pp. 1877–1901. Curran Associates, Inc., 2020.
- [15] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, **Advances in Neural Information Processing Systems**, Vol. 35, pp. 27730–27744. Curran Associates, Inc., 2022.
- [16] Graham Neubig. The Kyoto free translation task. <http://www.phontron.com/kftt>, 2011.
- [17] Rada Mihalcea and Ted Pedersen. An evaluation exercise for word alignment. In **Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond**, pp. 1–10, 2003.
- [18] David Vilar, Maja Popovic, and Hermann Ney. AER: do we need to “improve” our alignments? In **Proceedings of the Third International Workshop on Spoken Language Translation: Papers**, Kyoto, Japan, 2006.
- [19] Tom Kocmi, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Philipp Koehn, Benjamin Marie, Christof Monz, Makoto Morishita, Kenton Murray, Makoto Nagata, Toshiaki Nakazawa, Martin Popel, Maja Popović, and Mariya Shmatova. Findings of the 2023 conference on machine translation (WMT23): LLMs are here but not quite there yet. In **Proceedings of the Eighth Conference on Machine Translation**, pp. 1–42, Singapore, 2023. Association for Computational Linguistics.
- [20] Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTa: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In **ICLR**, 2023.
- [21] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. **arXiv preprint arXiv:2407.21783**, 2024.
- [22] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In **ICLR**, 2022.
- [23] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. **Computational Linguistics**, Vol. 29, No. 1, pp. 19–51, 2003.

A プロンプト

図2と図3には、LLM ベースの単語アライナーのフルモードとマーカーモードのプロンプト例をそれぞれ示す。

```
Prompt Example for LLM-based Word Alignment (Type: Full Mode)

Role: system
Content: You are a helpful AI assistant for word alignment.

Role: user
Content: Please give me the word alignment information between the following two sentences. The parallel sentences:
足利_0 義満_1 ( 2 あしかが_3 よしみつ_4 ) _5 は_6 室町_7 幕府_8 の_9 第_10 3_11 代_12 征夷_13 大_14
将軍_15 ( _16 在位_17 1368_18 年_19 _20 1394_21 年_22 ) _23 で_24 あ_25 る_26 。 _27

yoshimitsu_0 ashikaga_1 was_2 the_3 3rd_4 sei_5 taishogun_6 of_7 the_8 muromachi_9 shogunate_10 and_11
reigned_12 from_13 1368_14 to 1394_15 _16

Role: assistant
Content: Here is the word alignment information:
'''json
[["足利_0": ["ashikaga_1"], ["義満_1": ["yoshimitsu_0"], ["あしかが_3": ["ashikaga_1"], ["よしみつ_4":
["yoshimitsu_0"], ["室町_7": ["muromachi_9"], ["幕府_8": ["shogunate_10"], ["の_9": ["of_7"], ["第_10": ["the_3"],
["3_11": ["3rd_4"], ["代_12": ["3rd_4"], ["征夷_13": ["sei_5"], ["大_14": ["taishogun_6"], ["将軍_15":
["taishogun_6"], ["在位_17": ["reigned_12"], ["1368_18": ["1368_14"], ["年_19": ["1368_14"], ["1394_21":
["to1394_15"], ["年_22": ["to1394_15"], ["で_24": ["was_2"], ["あ_25": ["was_2"], ["る_26": ["was_2"], ["。_27":
["_16"]],
...
'''
```

図2 フルモードのプロンプト例。原文と翻訳文を入力し、モデルにすべての対応された単語を一度に出力させる。

```
Prompt Example for LLM-based Word Alignment (Type: Marker Mode)

Role: system
Content: You are a helpful AI assistant for word alignment.

Role: user
Content: Please give me the word alignment information between the following two sentences given the marked word
with ♠. The parallel sentences:
♠足利_0 ♠ 義満_1 ( 2 あしかが_3 よしみつ_4 ) _5 は_6 室町_7 幕府_8 の_9 第_10 3_11 代_12 征夷_13 大
_14 将軍_15 ( _16 在位_17 1368_18 年_19 _20 1394_21 年_22 ) _23 で_24 あ_25 る_26 。 _27

yoshimitsu_0 ashikaga_1 was_2 the_3 3rd_4 sei_5 taishogun_6 of_7 the_8 muromachi_9 shogunate_10 and_11
reigned_12 from_13 1368_14 to 1394_15 _16

Role: assistant
Content: Here is the word alignment information:
'''json
[["足利_0": ["ashikaga_1"]],
...
'''
```

図3 マーカーモードのプロンプト例。まず原文の1つの単語を特別なトークン（♠）でマークする。このマークされた単語に対して、翻訳文に対応する単語とその位置をモデルに予測させる。