多言語統語解析処理のための Multi-task LoRA SFT 方式の評価

松田 寛
株式会社リクルート Megagon Labs
hiroshi_matsuda@megagon.ai

浅原 正幸
国立国語研究所
masayu-a@ninjal.ac.jp

概要

大規模言語モデル (LLM) の性能向上とその微調整技術の普及は，様々な下流タスクの性能を引き上げると同時に，自然言語処理の基礎技術である統語解析処理の性能向上にも寄与している．本稿では，LLM の微調整技術である LoRA SFT を用いた多言語統語解析モデルを提案する．提案手法は，文書を入力とする言語判定＋文区切りタスク，文を入力とする単語分割＋言語固有品詞推定タスク，文と単語リストを入力とする依存構造解析タスクで構成され，これらのタスクを貫通動作させることで，言語を問わずテキストを入力するだけで依存構造解析結果を得ることができる．Universal Dependencies の 40 言語のデータセットを用いた実験により，マルチタスク学習では文区切り精度がボトルネックとなること，単語分割とともに言語固有品詞推定を行うことで単語分割精度が向上する等の知見を得た．研究成果のモデルおよび解析ライブラリは，商用利用可能なライセンスのもとで公開予定である．

1 はじめに

大規模言語モデルの台頭　近年の大規模言語モデル (LLM) の性能向上により，従来，自然言語処理 (NLP) 技術が用いられてきた応用タスクに LLM を適用することで，大幅な性能向上が得られるようになった．同時に，タスクやドメインに特化するためのチューニングも，Few-shot プロンプトでの事例提示や，指示応答関係の Supervised Fine-tuning (SFT) を効率的に実行する Parameter Efficient Tuning (PEFT) 技術の普及により，容易に実現できるようになった．こうした（従来は NLP 技術の専門家でなければ難しかった）チューニングの容易さの恩恵により，エンドユーザが LLM をブラックボックスとして様々な下流タスクに適用することが可能になった．

定量分析と NLP 技術　一方，LLM は（モデルが大規模であるが故に）推論に要する計算コストが高いこと，また LLM 単独ではテキストに対する定量的分析の性能が低い（単語出現頻度統計のような単純なタスクにも苦戦する）ことなどから，大規模テキストデータに対するマイニングタスクへの LLM の適用はあまり進んでおらず，この種のタスクでは従来型の NLP 技術が引き続き利用されている．テキストマイニングでは基礎処理として依存構造解析が用いられる [1] が，その学習・評価には Universal Dependencies (UD) [2] が広く用いられている．UD は 2015 年頃から構築が開始され，本稿執筆時点の最新版 (r2.17) では 186 言語に対応している．

NLP 技術の開発継続可能性　近年のオープンソース NLP フレームワークのほとんどは LLM が普及する以前の設計であり，主に Encoder 型モデルを用いて品詞推定・依存構造解析などの機能コンポーネントを実装し，それらを束ねて処理パイプラインを構成している．こうした従来型 NLP パイプラインの開発・メンテナンスには高度な専門知識と開発能力が求められる．しかし，市場における技術需要の大部分は既に従来型 NLP から LLM に移行しており，従来型 NLP パイプラインの改良とサポートを行う体制の維持には限界が生じる可能性がある．

LLM による統語解析　本稿では，従来型 NLP パイプラインを LLM で置き換えることにより，実装やモデルの学習フローの単純化と精度向上を行うことで，技術市場における NLP 技術の開発継続可能性の確保を念頭に，実応用を見据えた検討を行う[1)]．具体的には，先行研究 [3] の手法をもとに，モデルの重みが公開されている多言語 LLM に対して，タスク別に Low-Rank Adapter (LoRA) [4] を用いた SFT を行った結果を組み合わせることで，言語が未知である入力テキストの依存構造解析結果を出力できるようにする．さらに，複数の統語タスクをマージした場合の性能変化や，モデル手法の統語タスクでの有効性についても検討する．

1) LLM の推論効率の課題はハードウェアの性能向上に伴い徐々に解消されていくと期待する．

2 関連研究

2.1 NLP フレームワーク

本稿執筆時点で解析機能の改良が継続されている NLP フレームワークについて概観する（公開年順）．

spaCy Non-Monotonic Arc-Eager Transition System [5] で決定的もしくはビームサーチによる依存構造解析を行う．Transformer と解析コンポーネントとの間でマルチタスク学習が可能である．24 言語＋多言語モデルが提供されている．`GiNZA` [6] では日本語の文節を扱う拡張が行われている．

Stanza 言語判定から依存構造解析までパイプライン全段を End-to-End で学習可能[2]．依存構造解析は Bi-LSTM を用いた拡張 Biaffine モデルを使用 [7]．70 言語以上のモデルが提供されている．近年は句構造解析の取り組みも行われている [8]．

Spark NLP 200 以上の言語と幅広い開発環境 (Python/Scala/Java/R) への対応を謳う NLP フレームワーク [9]．依存構造解析のアルゴリズム等の詳細は公開されていない．

HanLP 共通の Encoder と複数の解析コンポーネントとの間でマルチタスク学習を行う．依存構造解析には Biaffine 派生モデルが用いられている [10]．100 言語以上のモデルが提供されている．

2.2 依存構造解析

前述の NLP フレームワークの多くは Biaffine モデル [11] の派生モデルで依存構造解析を行っている．近年は，`Hexatagger` [12] のように構文木の局所構造を少数のタグで表現して Encoder モデルで学習する手法が高い解析精度を示すことが報告されている．

LLM のような自己回帰型 Decoder モデルを用いた依存構造解析手法では，構文木をブラケティング表現で出力する方式 [13] より，`CoNLL-U` のようなテーブル構造で単語インデックスを介して依存構造を表現する方式 [3] の精度が高いとされる．本稿では先行研究 [3] の手法を拡張し，依存構造解析以外の処理も LLM で実行する手法を提案する．

2.3 モデルマージ手法

異なるタスクで訓練された複数の LoRA モデルをマージしてマルチタスクに対応した新しい LoRA モデルを構築手法が研究されている [14, 15]．また，`mergekit` [16] を用いることで[3]，様々なモデルマージ手法を試すことができる．簡単な予備実験でこれらのモデルマージ手法を試したところ，安定した推論が行えなかった[4]ため，モデルマージ手法の効果検証は今後の課題とする．

他方，`vLLM` [17] 等の LLM 高速推論ライブラリの直近のバージョンでは，GPU メモリ上で 1 つのベースモデルに複数の LoRA モデルを組み合わせて推論を行う機能が追加されており，個別のタスクで訓練された複数の LoRA モデルを切り替えて効率的に推論することができる．

3 提案手法

言語が未知のテキストを入力に与え，一連の統語解析処理を LLM で実行した結果を出力する手法を提案する．Appendix に使用するプロンプトを示す．

3.1 言語判定と文区切り

言語判定 (LANG) と文区切り (SENT) を指示するプロンプト (LS) とともにテキストを入力し，応答となる言語判定結果および文リストと入力の対応関係を訓練する．予備実験では次の 2 つの手法を比較したが，文区切りと単語分割を同時に行う手法では，同一内容が繰り返し出力される挙動により入出力の対応が取れないケースが頻発するため，安定して動作する文区切りのみのプロンプトを採用する．

- テキストを構成する文を改行区切りで出力する
- テキストを構成する単語を改行区切りで出力しつつ文区切りでは空行を出力する

3.2 単語分割と言語固有品詞推定

日本語の形態素解析では，単語分割の曖昧性解消と品詞推定を同時に行う形で隠れマルコフモデルを学習することで，単語分割精度が向上することが知られている [18]．同様の効果が LLM でも得られることを期待して，単語分割 (WORD) と言語固有品詞推定 (XPOS) を指示するプロンプト (WX) とともに言語と文を入力し，応答となる単語リスト (インデックス付き TSV) と入力との対応関係を訓練する．また言語固有品詞推定を行わず単語分割のみを行う場合と比較する．

2） 日本語モデルでは形態素解析器も利用可能．

3） mergekit で LoRA モデルを扱うには事前にベースモデルとのマージが必要．

4） 手法や実装には問題はなく，マージアルゴリズムの選択を含めたハイパーパラメータ探索が足りていないことが原因と考えている．

3.3 UD に基づく依存構造解析

UD 品詞推定・主辞判定・依存関係ラベリングの実行を指示するプロンプト (UD) とともに言語・文・インデックス付き単語リストを入力し，応答となる TSV 形式の依存構造出力との対応関係を訓練する．[3] では UD 品詞推定・主辞判定・依存関係ラベリングの 3 ステップで段階的に解析するプロンプトを用いる場合に最良の精度が得られることが報告されているが，ステップ数が多くなることで出力文脈が長くなる欠点もある[5]．本稿では，出力文脈を短縮するために，依存構造解析タスクの全てを 1 ステップで実行し，さらに出力 TSV から単語表記の列を省く形で訓練を行う．予備実験では，提案手法の文脈長（指示・応答の合計）は [3] の半分以下となり，UD 品詞推定精度 (UPOS)・ラベルなし主辞判定精度 (UAS)・ラベルつき主辞判定精度 (LAS) の 3 指標で [3] と同等の精度が得られることを確認した．

4 実験

4.1 データセット

商用利用可能なライセンスを持つ UD データセットのうち，Train が 4 万語以上ある次の 40 言語のデータセットを使用して実験を行う (Appendix に各言語のデータセットとその統計を示す)．Armenian, Belarusian, Bororo, Chinese, Chinese (Simplified), Croatian, Czech, Danish, Dutch, English, Estonian, Finnish, French, German, Haitian_Creole, Hebrew, Icelandic, Indonesian, Irish, Japanese, Korean, Latvian, Lithuanian, Naija, Norwegian, Persian, Portuguese, Romanian, Russian, Scottish_Gaelic, Serbian, Sindhi, Slovak, Slovenian, Spanish, Swedish, Thai, Turkish, Ukrainian, Western_Armenian.

日本語については，UD の他の膠着言語の単語分割基準に近い国語研長単位 [19] に基づいた UD_Japanese-GSDLUW (GSDLUW) と，UD_Japanese-BCCWJLUW[6] から新聞記事を除外したもの (BCCWJLUW-PN) を組み合わせて用いる．

学習には，日本語の GSDLUW・BCCWJLUW-PN と他の 39 言語[7] を全てマージしたものを使用する．評価には，日本語以外の 39 言語をマージしたものと，日本語の GSDLUW・BCCWJLUW-PN をそれぞれ独立に使用する[8]．これらのマージ済みのデータセットから，図 1 のプロンプトで言語判定＋文区切りタスク (LS) を，図 2 のプロンプトで単語分割＋言語固有品詞推定タスク (WX) を，図 3 のプロンプトで依存構造解析タスク (UD) を，それぞれ構築してシングルタスクの LoRA SFT を行う．さらに LS・WX・UD を組み合わせたマルチタスク学習も行う．

4.2 実験条件と精度指標

ハードウェア Google Cloud G4 インスタンス (8 × RTX Pro 6000 GPUs, 384-core AMD Turin CPUs, 768GB RAM), 250GB Hyperdisk, 10TB Filestore.

ソフトウェア Ubuntu 24.04, CUDA 12.8, Python 3.12.11, PyTorch 2.8.0, Transformers 4.57.0, Tokenizers 0.22.1, TRL 0.19.1, PEFT 0.17.1.

ベースモデル [3] で良好な精度を示した `gemma-2-9b` を使用する．

ハイパーパラメータ [3] の設定を踏襲するが，訓練時間短縮のためにエポック数は 2 に減らす[9]．

精度指標 精度指標として，言語判定精度 (LANG)，文区切り精度 (SENT)，単語分割精度 (WORD)，言語固有品詞推定精度 (XPOS)，UD 品詞推定精度 (UPOS)，ラベルなし主辞判定精度 (UAS)，ラベルつき主辞判定精度 (LAS) を用いる．実験は同一設定について学習と評価を 4 回試行する．精度には F1 値の平均を使用し，必要に応じて標本標準偏差を添える．

4.3 実験結果

シングルタスク学習とマルチタスク学習の比較 LS・WX・UD の個別タスクと，それらを組み合わた複合タスクを用いて `gemma-2-9b` を LoRA SFT した上で，タスク別に推論＋精度評価を行った．結果を表 1 に示す．全ての精度指標でシングルタスクの結果が最良であったが，文区切り (SENT) 以外ではマルチタスク学習との差は軽微であり，統語解析のマルチタスク学習では文区切り精度がボトルネックとなることが示された[10]．推論時に全タスクを貫通動作させる際に，GPU メモリ消費を抑えつつ精度を維持するには，LS と WX+UD の 2 つの LoRA を切り替えて使用する形が望ましい．

5） 出力文脈長は推論効率に強く影響する．また LLM の文脈長制限により，長文入力時に出力末尾で欠損が生じる．
6） 文および単語表記は一般向けには非公開．
7） XPOS が未付与の場合は一律にアンダースコアを出力する
8） 評価時の入力には正解の文区切りと単語分割を用いる．
9） num_epochs: 3, max_seq_length: 8192, lr: 3e-4, lr_scheduler: "cosine_with_min_lr", min_lr: 0.1, lora_r: 8, lora_dropout: 0.05, target_modules: "all-linear" (エンベッティング層は除外)
10） LoRA ランクを倍増しても傾向は変化しなかった．

表 1 学習タスクの組み合わせ方による精度変化．日本語を除く 39 言語と日本語 (GSDLUW) の解析精度について，4 回試行の平均値 ± 標本標準偏差を記載．下線は最良値．標本標準偏差のうち斜体で示したものは繰り返し同じ内容が出力される挙動の影響で増大している．

Models	LS:LANG		LS:SENT	
	日本語以外	ja_gsdluw	日本語以外	ja_gsdluw
LS	99.9±0.1	100.0±0.0	96.6±0.2	97.2±0.4
LS+WX	99.7±0.1	100.0±0.0	93.1±0.7	95.2±***1.7***
LS+UD	99.8±0.1	100.0±0.0	94.2±0.3	95.6±***1.3***
LS+WX+UD	99.4±0.3	100.0±0.0	93.0±***2.2***	94.1±0.7
Models	**WX:WORD**		**WX:XPOS**	
	日本語以外	ja_gsdluw	日本語以外	ja_gsdluw
WX	99.5±0.0	98.7±0.1	95.7±0.3	97.7±0.1
LS+WX	99.5±0.0	98.6±0.1	95.3±0.0	97.5±0.1
WX+UD	99.4±0.0	98.6±0.1	95.4±0.0	97.6±0.1
LS+WX+UD	99.4±0.1	98.5±0.2	95.3±0.3	97.5±0.2
Models	**UD:UPOS**		**UD:LAS**	
	日本語以外	ja_gsdluw	日本語以外	ja_gsdluw
UD	97.3±0.0	99.1±0.0	88.6±0.0	95.9±0.2
LS+UD	97.3±0.0	99.1±0.0	88.5±0.0	95.7±0.1
WX+UD	97.1±0.0	99.0±0.0	87.6±0.0	95.6±0.1
LS+WX+UD	97.2±0.2	99.1±0.1	87.8±0.8	95.7±0.2

表 2 BCCWJLUW-PN の追加学習効果の評価．各セルのカンマの左側は BCCWJLUW-PN を追加しない場合，右側は BCCWJLUW-PN を追加する場合の精度 (4 回試行の平均値，**太字**は有意差 $p < 0.05$ を示す)．

Task	日本語		日本語以外の
	GSDLUW	BCCWJLUW-PN	39 言語全体
LANG	100.0, 100.0	99.9, 100.0	99.7, 99.9
SENT	93.7, **97.2**	75.3, **88.2**	96.6, 96.6
WORD	97.6, **98.7**	92.7, **98.3**	99.5, 99.5
XPOS	96.2, **97.7**	90.3, **97.1**	95.4, 95.7
UPOS	98.6, **99.1**	97.3, **98.9**	97.3, 97.3
UAS	96.0, **96.5**	92.2, **95.5**	91.6, 91.5
LAS	94.8, **95.9**	89.0, **94.3**	88.7, 88.6

Appendix 表 4 にシングルタスク学習の言語別の評価結果を示す．LAS が低い言語のうち，ハイチ語は訓練データ量が少ないことが影響している可能性があるが，その倍の訓練データ量を持つボローロ語も精度が低く，訓練データ量だけが精度低下要因であるとは判断できない．また，文区切り精度が極端に低いスコットランド・ゲール語とタイ語については，エラー分析を通じた対策が必要である．

日本語データの追加学習効果 BCCWJLUW-PN を追加学習する効果の評価結果を表 2 に示す．日本語では言語判定以外の全指標で有意に精度が向上しており，特に WORD および XPOS において精度向上幅が大きい．対して，日本語以外の言語の精度に与える影響は軽微である．

表 3 単語分割を単独で行う場合 (WORD) と，単語分割とともに言語固有品詞推定を行う場合 (WORD+XPOS) の単語分割精度の比較（4 回試行の平均値 ± 標本標準偏差）．

	WORD	WORD+XPOS
日本語 GSDLUW	98.5±0.2	98.7±0.1
日本語 BCCWJLUW-PN	97.9±0.3	98.3±0.3
日本語以外の 39 言語全体 (単語分割精度 99.5 以下)	99.4±0.1	99.5±0.0
Chinese	97.0±0.6	97.7±0.3
Chinese (Simplified)	97.1±0.5	97.7±0.1
Hebrew	92.1±4.7	96.5±1.1
Indonesian	99.2±0.2	99.5±0.2
Russian	99.2±0.4	99.4±0.1
Scottish_Gaelic	97.8±0.5	98.4±0.1
Thai	90.6±0.8	91.5±0.4
Turkish	96.6±0.1	96.4±0.2

品詞推定が単語分割に与える影響 提案手法で，単語分割を単独で行う場合と，単語分割とともに言語固有品詞推定を行う場合の単語分割精度を比較した．結果を表 3 に示す．言語固有品詞推定を同時に行うことで，日本語では GSDLUW で 2 ポイント，BCCWJLUW-PN で 4 ポイント，単語分割精度が向上している．また，日本語以外で単語分割精度が 99.5 ポイント以下の言語では，トルコ語を除く 7 言語で精度向上傾向にある．

5 まとめ

LLM を用いた多言語統語解析モデルの学習において，LoRA SFT はシングルタスクで実施する方がマルチタスク学習より精度が高く，特に文区切りタスクでその傾向が顕著だった．推論時に GPU メモリ消費と精度をバランスするには，言語判定＋文区切りタスクで訓練した LoRA と，残りのタスクで訓練した LoRA を切り替えて使用する形が望ましい．

提案手法では，日本語の大規模データセットを追加学習することで，日本語の単語分割ならびに言語固有品詞推定の精度が大きく向上した．また，単語分割とともに言語固有品詞推定を行うことで単語分割精度が向上した．

今後の課題

- 言語別・タスク別のエラー分析
- レンマ化や形態論情報付与の LLM による実装
- 研究成果の解析モデルと推論ライブラリの公開
- 日本語に特化した高精度なモデルの構築と公開
- モデルマージ手法のハイパーパラメータ探索実験（マージアルゴリズムの比較を含む）

謝辞

本研究は，株式会社リクルート・国立国語研究所共同研究「日本語版 Universal Dependencies に基づく日本語依存構造解析モデルの研究開発」によるものです．

参考文献

[1] 石野亜耶, 小早川健, 坂地泰紀, 嶋田和孝, 吉田光男. Python ではじめるテキストアナリティクス入門. 講談社サイエンティフィク, 2022.

[2] Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. Universal Dependencies v1: A multilingual treebank collection. In **Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)**, May 2016.

[3] Hiroshi Matsuda, Chunpeng Ma, and Masayuki Asahara. Step-by-step instructions and a simple tabular output format improve the dependency parsing accuracy of LLMs. In **Proceedings of the 18th International Conference on Parsing Technologies (IWPT, SyntaxFest 2025)**, pp. 11–19, Ljubljana, Slovenia, August 2025.

[4] 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 **International Conference on Learning Representations**, 2022.

[5] Matthew Honnibal and Mark Johnson. An improved non-monotonic transition system for dependency parsing. In **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**, Lisbon, Portugal, September 2015.

[6] 松田寛. GiNZA - Universal Dependencies による実用的日本語解析. 自然言語処理, Vol. 27, No. 3, pp. 695–701, 2020.

[7] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations**, pp. 101–108, July 2020.

[8] John Bauer and Christopher D. Manning. High-accuracy transition-based constituency parsing. In **Proceedings of the 18th International Conference on Parsing Technologies (IWPT, SyntaxFest 2025)**, pp. 26–39, Ljubljana, Slovenia, August 2025.

[9] Veysel Kocaman and David Talby. Spark NLP: Natural language understanding at scale. **Software Impacts**, p. 100058, 2021.

[10] Han He and Jinho D. Choi. The stem cell hypothesis: Dilemma behind multi-task learning with transformer encoders. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, pp. 5555–5577, Punta Cana, Dominican Republic, November 2021.

[11] Timothy Dozat and Christopher D. Manning. Deep biaffine attention for neural dependency parsing. In **International Conference on Learning Representations**, 2017.

[12] Afra Amini, Tianyu Liu, and Ryan Cotterell. Hexatagging: Projective dependency parsing as tagging. In **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 1453–1464, Toronto, Canada, July 2023.

[13] Oriol Vinyals, Łukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. Grammar as a foreign language. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 28, p. 2773–2781. Curran Associates, Inc., 2015.

[14] Xun Wu, Shaohan Huang, and Furu Wei. Mixture of loRA experts. In **The Twelfth International Conference on Learning Representations**, 2024.

[15] Ryota Miyano and Yuki Arase. Adaptive LoRA merge with parameter pruning for low-resource generation. In **Findings of the Association for Computational Linguistics: ACL 2025**, pp. 19353–19366, Vienna, Austria, July 2025.

[16] Charles Goddard, Shamane Siriwardhana, Malikeh Ehghaghi, Luke Meyers, Vladimir Karpukhin, Brian Benedict, Mark McQuade, and Jacob Solawetz. Arcee's MergeKit: A toolkit for merging large language models. In **Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track**, pp. 477–485, Miami, Florida, US, November 2024.

[17] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In **Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles**, 2023.

[18] Masayuki Asahara and Yuji Matsumoto. Extended models and tools for high-performance part-of-speech tagger. In **Proceedings of the 18th Conference on Computational Linguistics - Volume 1**, COLING '00, p. 21–27, 2000.

[19] 小椋秀樹, 小磯花絵, 冨士池優美, 宮内佐夜香, 小西光, 原裕. 『現代日本語書き言葉均衡コーパス』形態論情報規程集 第 4 版（上）. Technical report, 国立国語研究所内部報告書, 2011.

A Appendix

```
We will now perform language identification and
sentence delimitation. First, output the language
identification result of the input text, then after
a blank line, output the sentence delimitation result.

input text:
<<<TEXT>>>
```

図 1 言語判定と文区切りに用いるプロンプト (LS).

```
We will now perform tokenization and part-of-speech tagging
on <<<LANGUAGE>>> sentence.
Split the input sentence into words with the word indexes
from 1 in TSV format, and add a field of part-of-speech tag.

input sentence:
<<<SENTENCE>>>
```

図 2 単語分割と言語固有品詞判定に用いるプロンプト (WX).

```
We will now perform dependency parsing on <<<LANGUAGE>>> sentence.
After splitting the input sentence into words with indexes as shown below, create a TSV with four fields:
word index from 1 to <<<TOKEN_NUM>>> + part of speech + the dependent word index + the Universal Dependencies relation.
However, for the word that is the main predicate of the sentence, the dependent word index should be 0.

input sentence:
<<<SENTENCE>>>

indexes and words:
<<<TOKEN_TSV:INDEX_FORM>>>
```

図 3 UD に基づく依存構造解析に用いるプロンプト (UD).

表 4 実験に使用した UD データセットの統計とシングルタスク学習時の精度（4 回試行の平均値 ± 標本標準偏差）．黄色のハイライトは相対的に訓練データが少ないこと，または，相対的に精度が低いことを示す．緑のハイライトは相対的に標本標準偏差が大きいことを示す．

Dataset	# of Sentences			# of Words			Unit-LS		Unit-WX		Unit-UD		
	Train	Dev	Test	Train	Dev	Test	LANG	SENT	WORD	XPOS	UPOS	UAS	LAS
UD_Armenian-ArmTDP	4,352	575	600	82,871	10,755	10,667	99.1 ± 0.7	98.0 ± 0.5	99.8 ± 0.1	99.8 ± 0.1	97.4 ± 0.1	90.9 ± 0.3	87.0 ± 0.4
UD_Belarusian-HSE	22,852	1,301	1,077	273,179	15,931	15,997	99.8 ± 0.4	95.2 ± 1.7	99.7 ± 0.0	97.8 ± 0.2	98.8 ± 0.1	92.1 ± 0.1	90.4 ± 0.1
UD_Bororo-BDT	17,107	2,138	2,139	128,418	15,678	16,260	99.9 ± 0.2	93.6 ± 1.8	100.0 ± 0.0	86.8 ± 0.4	87.1 ± 0.1	73.6 ± 0.2	65.9 ± 0.1
UD_Chinese-GSD	3,997	500	500	98,614	12,665	12,010	100.0 ± 0.0	99.9 ± 0.1	97.7 ± 0.1	95.0 ± 0.1	97.1 ± 0.1	88.7 ± 0.3	86.2 ± 0.3
UD_Chinese-GSDSimp	3,997	500	500	98,614	12,665	12,010	100.0 ± 0.0	100.0 ± 0.1	97.7 ± 0.3	95.0 ± 0.4	97.1 ± 0.0	88.5 ± 0.3	86.0 ± 0.4
UD_Croatian-SET	6,914	960	1,136	152,857	22,292	24,260	99.2 ± 0.4	99.0 ± 1.5	100.0 ± 0.0	96.1 ± 0.5	98.8 ± 0.1	94.2 ± 0.2	91.4 ± 0.1
UD_Czech-CAC	23,478	603	628	472,609	10,912	10,862	98.8 ± 0.5	100.0 ± 0.0	100.0 ± 0.0	97.4 ± 0.5	99.6 ± 0.0	95.4 ± 0.2	94.1 ± 0.2
UD_Danish-DDT	4,383	564	565	80,378	10,332	10,023	99.7 ± 0.5	98.3 ± 0.2	100.0 ± 0.0	100.0 ± 0.0	98.3 ± 0.1	89.5 ± 0.1	87.5 ± 0.1
UD_Dutch-Alpino	12,289	718	596	186,027	11,541	11,046	100.0 ± 0.0	95.7 ± 0.3	99.9 ± 0.0	96.9 ± 0.1	98.1 ± 0.1	95.0 ± 0.3	93.3 ± 0.3
UD_English-EWT	12,544	2,001	2,077	204,572	25,147	25,094	99.7 ± 0.2	94.1 ± 0.3	99.6 ± 0.0	97.7 ± 0.1	98.4 ± 0.1	95.6 ± 0.1	94.1 ± 0.1
UD_Estonian-EWT	5,444	833	913	67,429	10,001	13,152	99.6 ± 0.4	87.0 ± 1.9	99.2 ± 0.1	96.2 ± 0.4	95.9 ± 0.1	87.9 ± 0.1	84.4 ± 0.1
UD_Finnish-TDT	12,217	1,364	1,555	162,815	18,308	21,070	99.8 ± 0.3	96.8 ± 0.8	99.8 ± 0.0	98.3 ± 0.3	98.1 ± 0.1	93.8 ± 0.1	92.0 ± 0.1
UD_French-GSD	14,450	1,476	416	354,648	35,721	10,017	100.0 ± 0.0	97.7 ± 0.3	99.8 ± 0.0	99.8 ± 0.0	98.7 ± 0.0	96.2 ± 0.2	94.7 ± 0.2
UD_German-GSD	13,813	799	977	263,777	12,480	16,499	99.8 ± 0.4	98.2 ± 1.0	99.9 ± 0.0	97.8 ± 0.1	97.3 ± 0.0	90.5 ± 0.3	87.5 ± 0.3
UD_Haitian_Creole-Adolphe	2,605	265	444	55,527	6,186	10,021	100.0 ± 0.0	99.2 ± 0.2	100.0 ± 0.0	100.0 ± 0.0	95.5 ± 0.1	65.7 ± 0.4	61.9 ± 0.4
UD_Hebrew-IAHLTwiki	4,298	348	393	120,832	9,395	10,734	100.0 ± 0.0	96.8 ± 2.0	96.5 ± 1.1	93.8 ± 1.4	97.4 ± 0.1	95.0 ± 0.1	92.6 ± 0.1
UD_Icelandic-GC	3,960	500	540	78,568	10,694	10,349	100.0 ± 0.0	98.4 ± 1.9	100.0 ± 0.0	79.6 ± 1.3	94.4 ± 0.1	83.3 ± 0.2	78.4 ± 0.3
UD_Indonesian-GSD	4,482	559	557	97,601	12,661	11,756	100.0 ± 0.0	94.5 ± 1.5	99.5 ± 0.2	94.0 ± 0.6	94.5 ± 0.1	89.5 ± 0.3	83.2 ± 0.2
UD_Irish-IDT	4,005	451	454	95,880	10,000	10,109	100.0 ± 0.0	96.6 ± 3.2	99.6 ± 0.3	94.6 ± 0.6	96.4 ± 0.1	87.7 ± 0.3	81.7 ± 0.1
UD_Korean-Kaist	23,010	2,066	2,287	296,446	25,278	28,366	100.0 ± 0.0	99.1 ± 0.8	100.0 ± 0.0	90.1 ± 0.3	96.8 ± 0.1	90.1 ± 0.2	88.2 ± 0.2
UD_Latvian-LVTB	15,058	2,110	2,412	258,266	34,729	37,323	100.0 ± 0.0	99.6 ± 0.1	99.9 ± 0.0	92.7 ± 0.7	98.3 ± 0.1	93.0 ± 0.1	90.5 ± 0.2
UD_Lithuanian-ALKSNIS	2,341	617	684	47,641	11,560	10,846	100.0 ± 0.0	95.6 ± 0.7	99.9 ± 0.1	93.8 ± 1.0	97.2 ± 0.1	87.7 ± 0.3	84.5 ± 0.3
UD_Naija-NSC	7,279	990	972	111,916	14,571	14,350	100.0 ± 0.0	100.0 ± 0.1	100.0 ± 0.0	100.0 ± 0.0	98.3 ± 0.1	93.3 ± 0.3	90.9 ± 0.3
UD_Norwegian-Nynorsk	14,174	1,890	1,511	245,330	31,250	24,773	99.9 ± 0.2	99.4 ± 0.1	100.0 ± 0.0	99.0 ± 0.1	98.5 ± 0.0	94.6 ± 0.1	93.3 ± 0.1
UD_Persian-PerDT	26,196	1,456	1,455	452,496	25,147	24,133	100.0 ± 0.0	99.9 ± 0.1	99.8 ± 0.0	97.6 ± 0.1	97.8 ± 0.0	95.0 ± 0.1	92.8 ± 0.1
UD_Portuguese-Porttinari	5,893	842	1,683	117,972	16,528	33,580	99.9 ± 0.2	100.0 ± 0.1	99.9 ± 0.0	99.9 ± 0.0	99.3 ± 0.0	97.0 ± 0.1	96.1 ± 0.1
UD_Romanian-RRT	8,043	752	729	185,124	17,073	16,324	100.0 ± 0.0	98.4 ± 0.7	99.8 ± 0.0	97.2 ± 0.2	98.2 ± 0.1	94.4 ± 0.1	91.5 ± 0.1
UD_Russian-GSD	3,850	579	601	74,900	11,709	11,385	100.0 ± 0.0	99.3 ± 0.2	99.4 ± 0.1	97.6 ± 0.2	98.6 ± 0.1	94.4 ± 0.3	91.6 ± 0.4
UD_Scottish_Gaelic-ARCOSG	3,544	656	548	68,775	10,644	10,597	100.0 ± 0.0	70.8 ± 2.3	98.4 ± 0.1	88.3 ± 1.3	96.1 ± 0.1	87.1 ± 0.2	82.5 ± 0.3
UD_Serbian-SET	3,328	536	520	74,259	11,993	11,421	98.4 ± 0.6	100.0 ± 0.0	100.0 ± 0.0	96.4 ± 0.2	99.2 ± 0.0	95.6 ± 0.2	93.6 ± 0.2
UD_Sindhi-Isra	4,096	773	887	74,589	9,579	11,059	100.0 ± 0.0	84.4 ± 0.5	100.0 ± 0.0	91.1 ± 0.5	92.7 ± 0.2	84.0 ± 0.2	74.5 ± 0.2
UD_Slovak-SNK	8,483	1,060	1,061	80,874	12,754	12,744	99.3 ± 0.0	98.1 ± 0.7	100.0 ± 0.0	94.3 ± 0.5	98.1 ± 0.1	96.3 ± 0.4	95.0 ± 0.3
UD_Slovenian-SSJ	10,903	1,250	1,282	215,155	26,500	25,442	99.8 ± 0.2	99.5 ± 0.2	99.9 ± 0.0	97.0 ± 0.5	98.8 ± 0.0	95.2 ± 0.1	93.8 ± 0.1
UD_Spanish-GSD	14,187	1,400	427	382,444	37,154	12,002	100.0 ± 0.0	100.0 ± 0.0	99.9 ± 0.0	99.9 ± 0.0	97.5 ± 0.1	93.8 ± 0.2	91.8 ± 0.2
UD_Swedish-Talbanken	4,315	504	1,219	66,646	9,797	20,377	100.0 ± 0.0	99.6 ± 0.1	100.0 ± 0.0	97.5 ± 0.3	99.0 ± 0.1	94.6 ± 0.2	92.5 ± 0.1
UD_Thai-TUD	2,902	362	363	62,011	7,521	7,683	100.0 ± 0.0	72.4 ± 6.6	91.5 ± 0.4	91.5 ± 0.4	90.9 ± 0.0	86.8 ± 0.4	77.0 ± 0.4
UD_Turkish-BOUN	7,803	979	979	100,713	12,289	12,210	100.0 ± 0.0	89.2 ± 2.7	96.4 ± 0.2	85.3 ± 0.2	93.1 ± 0.1	82.6 ± 0.3	76.2 ± 0.2
UD_Ukrainian-ParlaMint	5,611	739	792	87,369	10,842	10,955	99.9 ± 0.2	99.7 ± 0.1	99.9 ± 0.1	98.0 ± 0.2	98.9 ± 0.1	94.5 ± 0.1	92.3 ± 0.3
UD_Western_Armenian-ArmTDP	5,274	693	677	95,823	13,430	13,499	99.0 ± 0.7	99.2 ± 0.5	99.9 ± 0.0	99.9 ± 0.0	96.5 ± 0.2	90.8 ± 0.3	86.3 ± 0.5
UD_Japanese-GSDLUW	7,050	507	543	130,283	9,531	10,428	100.0 ± 0.0	97.2 ± 0.4	98.7 ± 0.1	97.7 ± 0.1	99.1 ± 0.0	96.5 ± 0.2	95.9 ± 0.2
UD_Japanese-BCCWJLUW-PN	27,313	6,991	6,767	613,047	149,353	140,992	100.0 ± 0.1	88.2 ± 0.5	98.3 ± 0.3	97.1 ± 0.4	98.9 ± 0.0	95.5 ± 0.1	94.3 ± 0.1