日本語意味役割タスクにおいて複数 TokenID が与える影響

曽和晃太郎 [1]　竹内孔一 [2]
[1] 岡山大学　[2] 岡山大学大学院
pty01m90@s.okayama-u.ac.jp, takeuc-k@okayama-u.ac.jp

概要

本研究では，Byte-Level BPE アルゴリズムを採用したトークナイザのエンコードによって，1 つのトークンから複数のトークン ID が生成される場合に日本語意味役割タスクに与える影響について調査した．その結果，二つの日本語意味役割タスクそれぞれにおいて複数のトークン ID を持つデータが各タスクの精度に与える影響を明らかにし，各タスクにおける有効なデータの形を示した．

1　はじめに

昨今，ChatGPT の登場を皮切りに，様々な大規模言語モデル（Large Language Model: LLM）が提供されている．日本語に特化したモデルも多く提供され，それに伴い様々なトークナイザも登場している．しかし，それらがすべての単語をカバーできているわけではなく，うまくトークナイズ出来ず，不適切なデータを作成してしまうことも少なくない．

1.1　Byte-Level BPE

これまでの言語処理タスクで多く使用された言語モデルである東北大学 BERT モデルでは，トークナイズアルゴリズムに Mecab による形態素解析および，Subword 分割が採用されている [1]．一方で，CyberAgent から提供されている OpenCALM などの言語モデルでは，トークナイズアルゴリズムに Byte-Level BPE（Byte Pair Encoding）が採用されている．Byte-Level BPE は，Byte 単位で処理を行い，頻度の高い Byte 列を語彙として登録する [2]．

Byte-Level BPE を採用するトークナイザでは未知語などの予期せぬトークンが出現した際には，1 トークンに含まれる Byte 列から複数のトークン ID に変換してしまうといった現象が発生する．

鯛を華麗に捌く（鯛：このしろ）

図 1　未知語を含むテキスト

例えば，図 1 のような未知語を含んだテキストをトークナイズするとする

text	鯛を華麗に捌く								
Tokenized text	<0xE9>	<0xAF>	<0xAF>	を	華	麗	に	捌	く
Token ids	240	182	182	277	1286	5288	275	31129	492

図 2　未知語を含むテキストのトークナイズ結果

この文章を Byte-Level BPE のトークナイザを用いてトークナイズを行うと図 2 のようになり，1 つのトークンから 3 つのトークン ID が出力される．

1.2　意味役割付与 (Semantic Role Labeling: SRL) について

意味役割付与とは，文中の述語を基準として，「いつ」「どこで」「誰が」「何を」などの意味的関係を持つ項を予測し，それに対応したラベルを付与するタスクである．以下に例を示す．

彼	は	友人	を	家	に	招い	た
Arg0		Arg1		ArgM_LOC		V	O

図 3　「彼は友人を家に招いた」に含まれる意味役割

「彼は友人を家に招いた」という文章が与えられたとき，「招く」という述語を基準として，「彼は」に動作主を表す Arg0 というラベルを，招かれた対象の「友人を」に対象を表す Arg1 というラベルを，招いた場所である「家に」に場所を表す ArgM_LOC というラベルを付与する．

本研究で使用する意味役割ラベルは Propbank 形式を採用する．

1.3　概念フレーム付与について

概念フレーム付与とは，文中の述語の語義を予測し，それに対応した FrameID を付与するというタスクである．図 3 の例では「招く」という述語の語義として**勧誘**が正解の語義であり，それに対応した FrameID=470 が正解の ID となる．ここで，「招く」という述語は勧誘の他に，「専門家を外国から自社に招く」という文章に付されている**着点への移動**（FrameID=17）や「誤解を招く」という文章に付されている**因果**（FrameID=613）などという概念フ

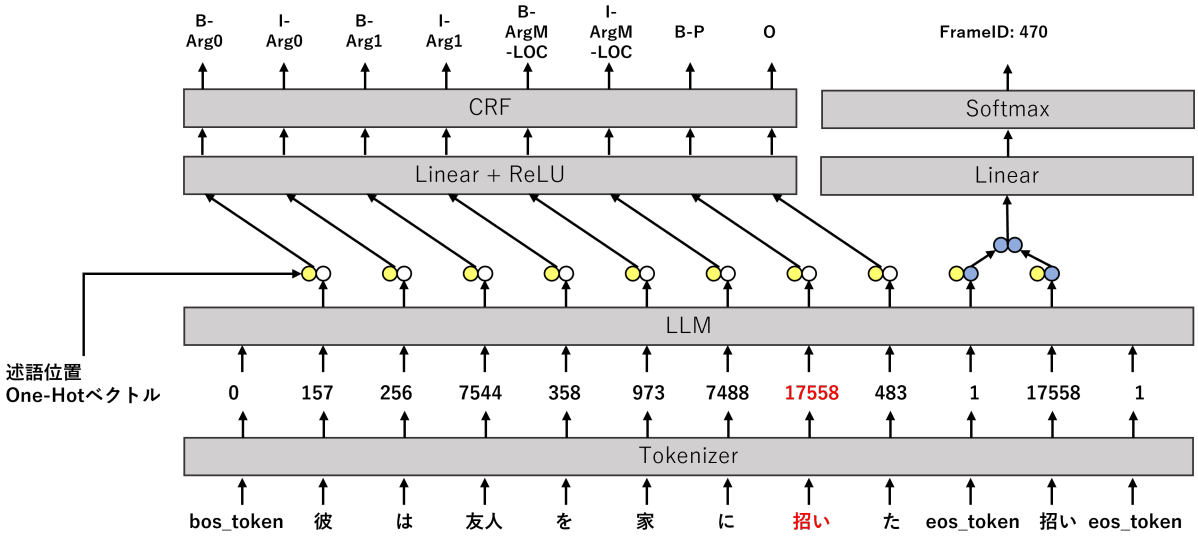

図 4　CRF を利用した意味役割および概念フレーム付与モデル

レームをもつ．

2　実験

2.1　データセット

本研究で使用するデータセットに含まれる日本語意味役割ラベルおよび概念フレームは，岡山大学竹内研究室で構築している述語項構造シソーラス [3] を基に作成している．国立国語研究所によって公開されている日本語コーパスである NPCMJ を，述語項構造シソーラスを用いて人手で分析し，概念フレームと項の意味役割を付したデータセットである NPCMJ-PT[4] を使用する．

NPCMJ-PT は，31 種類の意味役割ラベルと 1009 種類の概念フレームを持つ．文章数は 33958 件で，述語 1 つに対して 1 データとなっており計 54141 件のデータセットを作成している．また，述語項構造シソーラスでは，1096 件の FrameID を付与しているが，NPCMJ-PT にはその 92%の 1009 件の FrameID が含まれる．

今回の実験では学習データ，開発データ，テストデータをそれぞれ 8:1:1 の割合で使用する．

2.2　意味役割および概念フレーム付与モデル

2.2.1　モデルの概観

今回使用するモデルを図 4 に示す．このモデルでは，文章とその文に含まれる述語を入力として，トークン ID を取得，その後 LLM を通して特徴量を取り出す．取り出した特徴量を SRL, FID それぞれのデコーダへと渡す．SRL デコーダでは，CRF を用いてラベル列を出力する．FID デコーダでは Softmax 関数を通して 1096 種類の FID から確率の高い ID を出力する．

2.2.2　使用する大規模言語モデル

本研究では CyberAgent が提供している大規模言語モデルである OpenCALM[5] および CALM2[6] を使用する．OpenCALM はパラメータサイズによって複数モデルが提供されているが，その中から OpenCALM-Small, OpenCALM-Medium, OpenCALM-Large を使用する．

2.3　複数トークン ID の取り扱い

1.1 節で示した例のように一つのトークンから複数のトークン ID が出た場合以下の二つの方法でタスクを実行する．

1. 先頭の ID をそのトークンのトークン ID として使用する
2. 複数のトークン ID が出るデータは使用せず，すべてのトークンから単一の ID が得られるデータのみ使用する

1 の方法で実験を行う際に使用するデータを Full_HeadID，2 の方法で実験を行う際に使用するデータを Only_UniID とする．

今回使用するデータはあらかじめ Unidic を基にトークナイズしたものを使用する．こうすることで，LLM 側の予期しないトークンを相当量確保する

表 1 各モデルとデータによる比較

Model	Data	FID accuracy	SRL F1	SRL Precision	SRL Recall
OpenCALM-Small	Full_HeadID	**63.62**	41.06	46.61	39.16
OpenCALM-Small	Only_UniID	59.87	**49.19**	**55.37**	**46.75**
OpenCALM-Medium	Full_HeadID	**66.17**	43.14	49.17	40.89
OpenCALM-Medium	Only_UniID	62.33	**46.20**	**52.91**	**43.20**
OpenCALM-Large	Full_HeadID	**64.82**	42.29	48.59	39.87
OpenCALM-Large	Only_UniID	52.84	**44.20**	**51.04**	**40.94**
CALM2-7B	Full_HeadID	**52.91**	37.70	43.43	35.33
CALM2-7B	Only_UniID	46.16	**41.15**	**47.88**	**37.83**

ことが可能となる．ためしに OpenCALM のトークナイザを用いて全 54141 件のデータをトークナイズすると，複数トークン ID が現れたデータ数は 7716 件となり，Only_UniID は 46425 件となる．これにより複数トークン ID の与える影響が小さくなると考えられる．

表 2 各モデルにおける単一トークン ID の数と割合

Model	Full_HeadID	Only_UniID	data_ratio
OpenCALM-Small	54141	30130	55.7%
OpenCALM-Medium	54141	30130	55.7%
OpenCALM-Large	54141	30130	55.7%
CALM2-7B	54141	24976	46.1%

また，今回使用するモデルとのデータ数の関係を表 2 に示す．OpenCALM はパラメータ数によらずトークナイザは同一のものであるため，単一 ID のみで構成されたデータ Only_UniID はすべて 30130 件で統一されている．また，Full_HeadID に対する割合は 55.7%である．CALM2 は Only_UniID が 24976 件，Full_HeadID に対する割合は 46.1%である．

3 実験結果

実験結果を表 1 に示す．SRL に関しては，すべてのモデルにおいて単一の ID のみのデータを使用したほうが高いスコアを得られた．一方で，FID ではすべてのモデルにおいて単一の ID のみのデータを使用したほうがスコアが低くなるという結果になった．

この結果より，概念フレーム付与タスクに関しては，少量のトークン境界が適切なデータを用いた場合よりも，トークン境界が不適切なデータを含んでいたとしても，大量のデータを用いたほうが良い結果が得られることがわかる．一方で，意味役割付与タスクの場合はデータの量よりもトークン境界が適切なデータを少量でも集めたほうが有効であることがわかる．この結果は CALM2 の結果から元のデータ量と比べて半数以上少ないデータの場合にも同様のことが言える．

表 3 OpenCALM 各モデルにおける Full_HeadID と Only_UniID のスコアの差分

	Only_UniID - Full_HeadID			
Model	FID accuracy	SRL F1	SRL Precision	SRL Recall
OpenCALM-Small	-3.75	8.13	8.76	7.59
OpenCALM-Medium	-3.84	3.06	3.74	2.31
OpenCALM-Large	-11.98	1.91	2.45	1.07

また，OpenCALM ではパラメータ数で各スコアの増減に特徴がみられた．表 3 に各モデルにおいて Only_UniID のスコアから Full_HeadID のスコアを引いた差分を示す．

この結果より，パラメータ数が大きくなるほど，FID のスコアの低下が大きくなり，SRL のスコアの増加量が小さくなっていることが分かる．すなわち，パラメータ数が少ないモデルほど単一トークン ID のデータの恩恵が大きく，パラメータ数が大きいモデルほどデータの量の低下に大きな影響を受けることが分かる．

4 まとめ

実験結果から，意味役割付与と概念フレーム付与の二つのタスクではデータに求められる要件が異なることが示された．意味役割付与タスクでは，適切なトークン境界を揃えた質的なデータセットが求められる一方で，概念フレーム付与タスクでは量的なデータセットが求められる．すなわち，この二つのタスクを同時に行うマルチタスク学習においては，この質的データセットと量的データセットの両方の要件を満たす必要があると考える．

参考文献

[1] 築地俊平, 新納浩幸. Tokenizer の違いによる日本語 BERT モデルの性能評価. 言語処理学会 第 27 回年次大会 発表論文集, pp. 781–784, 2021.

[2] 井上誠一, Nguyen Tung, 中町礼文, 李聖哲, 佐藤敏紀. 日本語 GPT を用いたトークナイザの影響の調査. 言語処理学会 第 28 回年次大会 発表論文集, pp. 6–10, 2022.

[3] 岡山大学竹内研究室. 述語項構造シソーラス, 2023. https://pth.cl.cs.okayama-u.ac.jp/.

[4] Koichi Takeuchi, Alastair Butler, Iku Nagasaki, Takuya Okamura, and Prashant Pardeshi. Constructing Web-Accessible Semantic Role Labels and Frames for Japanese as Additions to the NPCMJ Parsed Corpus. In **Proceedings of The 12th Language Resources and Evaluation Conference (LREC2020)**, 2020.

[5] Alex Andonian, Quentin Anthony, Stella Biderman, Sid Black, Preetham Gali, Leo Gao, Eric Hallahan, Josh Levy-Kramer, Connor Leahy, Lucas Nestler, Kip Parker, Michael Pieler, Shivanshu Purohit, Tri Songz, Wang Phil, and Samuel Weinbach. GPT-NeoX: Large Scale Autoregressive Language Modeling in PyTorch, 8 2021.

[6] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. **arXiv preprint arXiv:2302.13971**, 2023.