日本語の分類タスクにおけるカリキュラム学習とマルチタスク学習の効果検証

植松拓也　河原大輔
早稲田大学理工学術院
takuya1009@akane.waseda.jp　dkw@waseda.jp

概要

複数の言語処理タスクを順番もしくは同時に学習する方法として，カリキュラム学習やマルチタスク学習がある．これらは英語のベンチマークにおいて包括的に調査されており有効性が報告されているが，日本語では調査されていない．本研究では，日本語言語理解ベンチマーク JGLUE に含まれる分類タスクにおいて，カリキュラム学習およびマルチタスク学習を適用し，有効性を検証する．結果として，親和性のあるタスク同士でカリキュラム学習，マルチタスク学習を行うことで精度が向上することを確認した．

1　はじめに

機械学習に基づく自然言語処理においてよく用いられる手法の一つに転移学習がある．転移学習の一手法として，事前学習をしてから対象のタスクごとにファインチューニングをするという 2 段階で行うことが多い．転移学習のモデルとしては BERT [1] など様々なモデルが開発されている．BERT は，言語理解ベンチマーク GLUE [2] において当時の最高スコアを達成しており，広く利用されている．

事前学習やファインチューニングを行う際に，それぞれにおいてどのようなタスクをどのような順番で解くかが重要である．この点に着目した手法にカリキュラム学習があり一定の成果が出ている [3, 4, 5, 6, 7]．人間が学習する際，まず難しいことから学ぶのではなく簡単なことから学んだ方が良い．それと同様に，学ぶ順番が大切なことは機械学習にも同じことが言える．機械学習におけるカリキュラム学習は，学習データを最も効果的な順番に並び替え段階的に学習する手法で，簡単な学習データから学習して徐々に難易度を上げていくことが多い．

また，近年はマルチタスク学習も注目されている [8, 9, 10, 11]．マルチタスク学習は一度に 2 つ以上のタスクで学習する手法であり，あるタスクから得られた知識を他のタスクにも適用することで精度が向上する場合がある．ただし，タスク同士に親和性がない場合は精度が低下する可能性があるため，親和性のあるタスク同士でマルチタスク学習を行うことが重要である [12]．

このような研究は英語などで活発に行われている一方，これまで日本語では包括的な調査が行われていない．本研究では，日本語におけるタスク同士の親和性や効果的な学習方法を調査することを目的とし，日本語データセットを用いてカリキュラム学習およびマルチタスク学習を行う。事前学習モデルには日本語 BERT を用い，タスクには日本語言語理解ベンチマーク JGLUE [13] に含まれる自然言語推論 (JNLI)，意味的類似度計算 (JSTS)，評価極性分類 (MARC-ja) の 3 種類の分類タスクを用いる．カリキュラム学習では，あるタスクでファインチューニングした後，さらに対象タスクでファインチューニングすることで対象タスクの精度が向上するかを評価する．マルチタスク学習では，上記 3 種類のタスクから 2 つを選び同時に学習し，有効性を評価する．

2　関連研究

2.1　カリキュラム学習

カリキュラム学習については，まず簡単な事例から学習し，徐々に難しい事例を学習する場合が多い．例えば，画像分類タスクにおいてカリキュラム学習の有効性を示した実験結果がある [3]．この実験では，自動的に生成された三角形や楕円，長方形の図形を分類するタスクを解く際に，まず正三角形や円，正方形などの簡単な図形から学び，徐々に色の種類を増やしたり難しい図形について学ぶこと

により，カリキュラム無しの学習よりも精度が高くなっている．

カリキュラム学習は自然言語処理でも活用されているが，難易度の定義が研究ごとに異なる．難易度の指標としては語彙サイズや入力された単語列の長さがある．Bengio らは，言語モデルにおいて次の単語を予測する際に，はじめに小さな語彙サイズのモデルを学習して徐々に語彙サイズを増やしながら学習した方がカリキュラム無しよりも精度が高くなることを報告している [3]．Cirik らは，入力文の評価極性を 5 つのクラスに分類する際に，文長の短い事例から学習して徐々に長い事例を学習するカリキュラム学習を提案した [4]．その結果，精度が向上し，特に学習データ数が少ないときに効果があることを示している．

難易度の定義の方法は他にもある．Guo らは，非自己回帰機械翻訳においてファインチューニングする際にカリキュラム学習を導入した [5]．非自己回帰機械翻訳モデルは自己回帰機械翻訳モデルと比較して，トークンを並列に生成できるため推論の速度は向上するが，精度は低下する．この問題に対処するために，まず自己回帰訓練を行い，徐々に非自己回帰訓練に変更していくようなカリキュラム学習を提案し，有効性を示している．Xu らは，GLUE ベンチマークでカリキュラム学習する実験を行った [6]．まず，対象のデータセットを分割して交差検証を行い，精度を難易度とみなす．次に，分割されたデータセットを難易度順に並び替え，簡単な事例からファインチューニングする．結果としては GLUE における 9 つのタスクのうち 8 つでベースラインよりも優れていた．

カリキュラム学習において，難易度順に並び替えるのではなく，学習するタスクの順序を入れ替えて他のタスクに与える効果を検証した実験がある．Yogatama らは，あるタスクを BERT などで学習して得られた知識を他のタスクを解く際に有効活用できないかを検証した [7]．どのようなカリキュラムで学習をするのが最適で，破滅的忘却を起こさないかを実験しており，学習の高速化を達成したタスクの組合せやそのときの精度について述べている．

2.2 マルチタスク学習

マルチタスク学習に基づく方法は多数提案されている．Collobert らは，品詞タグ付けやチャンキングなど関連性のあるタスクを同時に学習するマルチタスク学習を提案し，有効性を示した [8]． Liu らは，映画のレビューについてのテキスト分類タスクを複数用意してマルチタスク学習を行った [9]．その結果，対象タスクの精度が他のタスクの助けを借りて向上することを示した．

また，マルチタスク学習を BERT などの事前学習モデルに適用した研究も行われている．事前学習モデルのファインチューニングの際にマルチタスク学習を適用したモデルに MT-DNN がある [10]．このモデルは，GLUE ベンチマークにおける 9 つのタスクのうち 8 つのタスクと，自然言語推論のコーパスである SNLI [14] や SciTail [15] で BERT の性能を上回った．Zhou らは，品詞タグ付け，構文解析，意味役割付与などの言語解析タスクを事前学習時にマルチタスクで行う LIMIT-BERT を提案した [11]．LIMIT-BERT は言語解析タスクの精度を向上させ，GLUE ベンチマークや SNLI において BERT ベースラインよりも優れていることが報告されている．

2.3 本研究の位置づけ

上記の研究のように，これまでは英語での実験は行われているが日本語における包括的な調査は行われていない． 本研究では，日本語のデータセットにおけるカリキュラム学習とマルチタスク学習の有効性を調査する．

3 対象タスクと学習方法

本節では，本研究において対象とするタスク，および，カリキュラム学習とマルチタスク学習の方法について述べる．

3.1 対象タスク

対象のタスクとして，日本語言語理解ベンチマーク JGLUE に含まれる自然言語推論 (JNLI)，意味的類似度計算 (JSTS)，評価極性分類 (MARC-ja) の 3 種類の分類タスクを用いる．JNLI と JSTS は文ペア分類タスクであり，MARC-ja は文章分類タスクである．各タスクのデータセットの事例数，評価指標を表 1 に示す．combined score は，Pearson と Spearman 相関係数の平均をとったスコアである．事前学習モデルを train セットでファインチューイングし，valid セットで評価を行う．

表 1　データセットの事例数と評価指標

	JNLI	JSTS	MARC-ja
train	20,073	12,451	187,528
valid	2,434	1,457	5,654
metric	accuracy	combined score	accuracy

3.2　学習方法

3.2.1　カリキュラム学習

本研究では，3 種類のタスクの組合せを変えて，2 段階のファインチューニングを行う．具体的には，まず JNLI，JSTS，MARC-ja のぞれぞれについてファインチューニングを行い，その結果得られたモデルを用いて再度ファインチューニングを行う． 異なるタスクで再びファインチューニングを行うことの有効性を調査するため，同じタスクで再びファインチューニングした場合とも比較する．

事前学習モデルには東北大 BERT-base[1)]を用いる．ファインチューニングの際，CLS トークンに対して JSTS は回帰問題，JNLI と MARC-ja は分類問題を解く．いずれもエポック数は 4 で固定し，4 エポック終了時点の評価値を比較する．ハイパーパラメータについてはいくつかの設定から最適な組合せを選ぶ．ハイパーパラメータの詳細な設定は付録 A に示す．

3.2.2　マルチタスク学習

ファインチューニングする際，JSTS，JNLI，MARC-ja の 3 つのタスクの中から 2 つを選びマルチタスク学習を行う．マルチタスク学習が終わった後に，JSTS，JNLI，MARC-ja を別々に評価する．

事前学習モデルにはカリキュラム学習と同様に東北大 BERT-base を用い，マルチタスク学習のモデルに分類ヘッドを 2 つ用意する．その概要を図 1 に示す．ハイパーパラメータについては最適な組合せを選択する．

4　実験結果と議論

4.1　カリキュラム学習の実験結果

カリキュラム学習の結果を表 2 に示す．ここで，1 段階目のタスクはファインチューニングの 1 段階目で使用したタスク，列名のタスクは 2 段階目のファインチューニングで使用したタスクを示す．また，評価は全て 2 段階目のファインチューニング後の精度である．

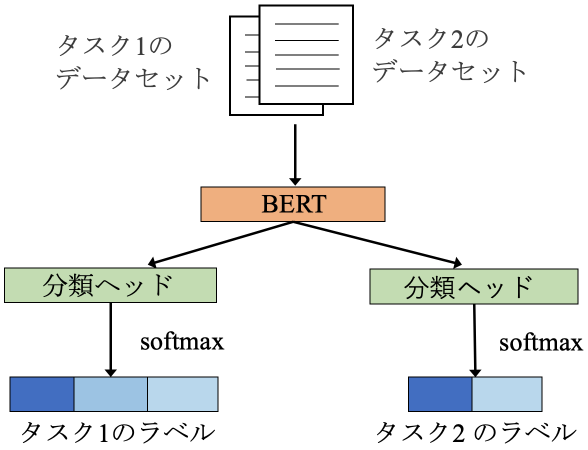

図 1　マルチタスク学習のモデル

表 2　カリキュラム学習の実験結果

1 段階目のタスク	JNLI	JSTS	MARC-ja
なし	0.901	0.891	0.956
JNLI	0.905	0.894	0.957
JSTS	0.915	0.894	0.957
MARC-ja	0.890	0.881	0.957

JNLI における評価　JNLI で評価したときの結果を比較する．JSTS でファインチューニングしてから，そのモデルを用いて JNLI で再びファインチューニングをしたときに精度が最も高いことが確認できる．この結果は 2 段階とも JNLI でファインチューニングしたときと比較して 1.0%高い．一方，MARC-ja でファインチューニングしてから JNLI でファインチューニングすると精度が下がっている．

JSTS における評価　JSTS で評価したときの結果を比較する．JNLI でファインチューニングしたモデルを用いて JSTS をファインチューニングをすると，JSTS で 1 回ファインチューニングしたときより約 0.3%精度が上がっている．ただし，JSTS でファインチューニングした後，さらに JSTS でファインチューニングしたときと精度は同じであった．MARC-ja でファインチューニングしてから JSTS でファインチューニングすると精度は下がった．

MARC-ja における評価　MARC-ja で評価したときの結果を比較する．MARC-ja を解く際は，事前に JNLI，JSTS，MARC-ja でファインチューニングしてから MARC-ja でファインチューニングしても，MARC-ja で 1 回ファインチューニングをしたときと精度はほとんど変化がない．

1）　https://huggingface.co/cl-tohoku/bert-base-japanese-v2

表 3　カリキュラム学習による各ラベルの予測数

タスク	中立	含意	矛盾
JNLI (正解データ)	1,347	353	734
JNLI のみ	1,366	349	719
JSTS → JNLI	1,363	329	742

表 4　マルチタスク学習の実験結果

タスク	JNLI	JSTS	MARC-ja
シングルタスク	0.901	0.891	0.956
JNLI と JSTS	0.913	0.899	−
JNLI と MARC-ja	0.802	−	0.958
JSTS と MARC-ja	−	0.890	0.952

カリキュラム学習についての考察　以上の結果より，JNLI と JSTS は親和性が高いと言える．JNLI と JSTS はともに文ペア分類タスクであり, 構成する文ペアのほとんどが重複していることが 1 つの要因と考えられる．特に，JSTS でファインチューニングしてから JNLI を解いたとき精度がかなり向上した．

JNLI をファインチューニングしたときと，1 段階目を JSTS，2 段階目を JNLI でファインチューニングしたときの各ラベルの予測数について表 3 にまとめる．ただし，1 行目は JNLI の valid データにおける正解ラベルの分布を示す．事前に JSTS でファインチューニングすると，含意の予測数が減少し，矛盾の予測数が増加した．また，ラベルごとの正解数については，矛盾の正解数が他のラベルに比べて増えていた．特にカリキュラム学習を行うことで予測ラベルが変化した文ペアの中で，中立の予測が矛盾に変化したときの正答率が高くなっていた．含意，矛盾は中立と比較して類似度が高く，先に JSTS で学習することで類似度の弁別性能が上がり中立とそれ以外のラベルの分類が得意になる可能性がある．

一方，MARC-ja の結果から MARC-ja と JSTS，MARC-ja と JNLI の親和性は高くないと言うことができる．MARC-ja は文章分類タスクであり，JNLI や JSTS とは異なるタスクであり，データセットの構成も異なる．カリキュラム学習を行う際は，事前に親和性のないタスクでファインチューニングをすると逆効果になる可能性を示唆している．

4.2　マルチタスク学習の実験結果

マルチタスク学習の結果を表 4 に示す．タスクの列は，マルチタスク学習を行ったタスクのペアを示す．1 行目はシングルタスク学習の精度を示す．

各タスクペアにおける比較　1 行目のシングルタスク学習の精度と比較すると，JNLI と JSTS のマルチタスク学習を行うとそれぞれ精度が 1.2%，0.8% 上がっている．一方，JNLI と MARC-ja のマルチタスク学習をしたときは JNLI の精度が約 10%下がっている．また，JSTS と MARC-ja のマルチタスク学習をしたときは，JSTS，MARC-ja それぞれ 0.9%，0.4%精度が下がっている．

マルチタスク学習についての考察　上記の結果から，親和性のあるタスク同士でマルチタスク学習すると各タスクにおいて精度が上がると考えられる．一方，あまり親和性のないデータセット同士でマルチタスク学習すると両方の精度が落ちる，もしくは片方のタスクの精度が下がる可能性があり，マルチタスク学習の効果はほとんどないと考えられる．また，マルチタスク学習では，タスク同士の競合が発生する可能性がある [16]．MARC-ja と他のタスクが競合し，学習データのサイズが大きい MARC-ja はあまり影響を受けず，学習データのサイズが小さい JNLI と JSTS は精度がかなり低下したといえる．

カリキュラム学習との比較　上記の結果を比較すると，JNLI で評価する際は事前に JSTS でファインチューニングをするカリキュラム学習の方が適している．一方，JSTS で評価する際は，JNLI と同時に学習するマルチタスクの方が適している．カリキュラム学習とマルチタスク学習についてはタスクによって適している学習方法を採用する必要がある．

5　おわりに

本研究では，JGLUE ベンチマークにおける JNLI，JSTS，MARC-ja の 3 つの分類タスクを組合せてカリキュラム学習やマルチタスク学習を実行した．その結果により，タスク同士の親和性やどの順序でファインチューニングをすると精度が向上するのかを確かめた．本研究の結果は，学習する際に用意できるデータセットが少ない場合でも，親和性のある他のデータセットと組合せてカリキュラム学習やマルチタスク学習を行うことで精度が改善する可能性があることを示している．

今後は，タスク間の親和性について，QA タスクなど他のタスクを含めて調査したい．また，BERT 以外にも RoBERTa [17] など高性能な事前学習モデルでも実験して効果を確かめたい．また，カリキュラム学習とマルチタスク学習を組合せて学習することで精度が向上するかを確認する予定である．

謝辞

本研究は JSPS 科研費 JP21H04901 の助成を受けて実施した.

参考文献

[1] 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, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[2] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In **Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP**, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[3] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In **Proceedings of the 26th Annual International Conference on Machine Learning**, ICML '09, p. 41–48, New York, NY, USA, 2009. Association for Computing Machinery.

[4] Volkan Cirik, Eduard H. Hovy, and Louis-Philippe Morency. Visualizing and understanding curriculum learning for long short-term memory networks. arXiv, 2016. abs/1611.06204.

[5] Junliang Guo, Xu Tan, Linli Xu, Tao Qin, Enhong Chen, and Tie-Yan Liu. Fine-tuning by curriculum learning for non-autoregressive neural machine translation. In **Proceedings of the aaai conference on artificial intelligence**, Vol. 34, pp. 7839–7846, 2020.

[6] Benfeng Xu, Licheng Zhang, Zhendong Mao, Quan Wang, Hongtao Xie, and Yongdong Zhang. Curriculum learning for natural language understanding. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 6095–6104, Online, July 2020. Association for Computational Linguistics.

[7] Dani Yogatama, Cyprien de Masson d'Autume, Jerome Connor, Tomas Kocisky, Mike Chrzanowski, Lingpeng Kong, Angeliki Lazaridou, Wang Ling, Lei Yu, Chris Dyer, and Phil Blunsom. Learning and evaluating general linguistic intelligence. arXiv, 2019. abs/1901.11373.

[8] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In **Proceedings of the 25th international conference on Machine learning**, pp. 160–167, 2008.

[9] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Recurrent neural network for text classification with multi-task learning. arXiv, 2016. abs/1605.05101.

[10] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Multi-task deep neural networks for natural language understanding. In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**, pp. 4487–4496, Florence, Italy, July 2019. Association for Computational Linguistics.

[11] Junru Zhou, Zhuosheng Zhang, Hai Zhao, and Shuailiang Zhang. LIMIT-BERT : Linguistics informed multi-task BERT. In **Findings of the Association for Computational Linguistics: EMNLP 2020**, pp. 4450–4461, Online, November 2020. Association for Computational Linguistics.

[12] Michael Crawshaw. Multi-task learning with deep neural networks: A survey. arXiv, 2020. abs/2009.09796.

[13] Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. JGLUE: Japanese general language understanding evaluation. In **Proceedings of the Thirteenth Language Resources and Evaluation Conference**, pp. 2957–2966, Marseille, France, June 2022. European Language Resources Association.

[14] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**, pp. 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[15] Tushar Khot, Ashish Sabharwal, and Peter Clark. Scitail: A textual entailment dataset from science question answering. In **Proceedings of the AAAI Conference on Artificial Intelligence**, Vol. 32, 2018.

[16] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, **Advances in Neural Information Processing Systems 31**, pp. 525–536. Curran Associates, Inc., 2018.

[17] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. arXiv, 2019. abs/1907.11692.

A　参考情報

A.1　実験時のハイパーパラメータ

実験を行う際のハイパーパラメータの一覧を表 5 にて示す．この中から最適なハイパーパラメータを選ぶ．

表 5　ファインチューニングを行う際のハイパーパラメータ

Name	Value(s)
learning rate	5e-5, 2e-5, 1e-5
epoch	4
warmup ratio	0.1
max seq length	512 (MARC-ja), 128 (JSTS，JNLI)
batch size	8, 16, 32