時間的常識理解へ向けた効果的なマスク言語モデルの検証

木村 麻友子 [1]　Lis Kanashiro Pereira[1]　浅原 正幸 [2]
Fei Cheng[3]　越智 綾子 [2]　小林 一郎 [1]
[1] お茶の水女子大学　[2] 国立国語研究所　[3] 京都大学
{g1720512,koba}@is.ocha.ac.jp,kanashiro.pereira@ocha.ac.jp
{masayu-a,a.ochi}@ninjal.ac.jp,feicheng@i.kyoto-u.ac.jp

概要

本研究では，時間的常識推論に対する言語モデルの開発に焦点を当て，効果的なマスク言語モデルの検証を行った．ファインチューニングの前に，時間的常識を問う対象のデータセットを用いた Masked Language Modeling（MLM）と Next Sentence Prediction を行い，また，MLM においてマスクする対象となるトークンを様々な設定に変更して実験を行なった．実験の結果，対象タスクのトピックに合わせて単語をマスクすることにより，時間的常識を問うタスクにおいて，4,5%程度の精度の向上を確認できた．

1　はじめに

近年，BERT [1] などの事前学習済み言語モデルが幅広い自然言語処理（NLP）タスクで大きな成果を上げているが，これらのモデルは時間推論においてはまだ性能が低い場合と言われている [2]．特に困難な課題として，時間的常識を扱う推論が挙げられる．例えば，「旅行に行く」と「散歩に行く」という 2 つの事象が与えられたとき，多くの人間は「休暇は散歩よりも長く，発生頻度も少ない」ということを知っているが，コンピュータは「休暇は散歩よりも長く，発生頻度も少ない」という時間的常識をもって推論することが困難である．

そこで，本研究では，時間的常識推論に対するモデルの開発に焦点を当て，先行研究 [3] の継続として時間的常識を理解するための汎用言語モデルの開発を行う．対象のデータセットを用いた Masked Language Modeling（MLM）と Next Sentence Prediction（NSP）をタスク適応事前学習（task-adaptive pre-training, TAPT）として実験を行う．また，MLM においてマスクする対象となるトークンを様々な設定に変更することにより，対象タスクを解くのに必要な常識的知識を付加したモデルを提案する．

2　提案手法

2.1　時間常識推論課題 MC-TACO

本研究では，対象となる課題として MC-TACO [4] を使用する．MC-TACO では，時間特性に関する 5 つの特徴量 (duration, temporal ordering, typical time, frequency, stationarity) を定義しており，自然言語で表現された事象の時間的常識を理解する課題から構成されるデータセットである．5 つの特徴量のいずれかの特性について記述された文章とその文章に関する質問，それに対する答えの候補，各候補に対して正解には yes，不正解には no とラベル付けされたものから構成されており，yes か no かを予測する二値分類のタスクである．

2.2　タスク適応事前学習 (TAPT)

BERT は，対象タスクに対してファインチューニングを行うだけで良い性能を発揮するが，事前学習されたモデルと対象課題との間にドメインの不整合がある場合，タスクの精度向上が見込めない場合がある．この問題を解決するために，対象データセットを用いて事前学習を行うことは，事前学習されたモデルをターゲットタスクに適応させるために有用である [5]．これに基づき，MC-TACO データセットを用いて，BERT に対して Masked Language Modeling（MLM）タスクと Next Sentence Prediction（NSP）タスクを実施する． MLM は，トークンの一部をランダムに特殊なトークン（例えば，[MASK]）に置き換え，モデルにその予測をさせるタスクである．NSP は，与えられた 2 文のペアに対して，一方が他方に続く文かどうかを判断するタスクである．本研究では，MLM において，一般的に使用されるマスクする対象のトークンをランダムに 15%選択する方式を任意の基準を採用したものに変更し，課題の精度へ

の影響を精査する．以下に，採用した基準について説明する．

2.3 対象タスクのトピック

対象タスクのトピックに関連する語彙をマスクすることにより，対象タスクを解くのに必要な知識を重点的に獲得することを目指す．今回の対象データセットである MC-TACO は時間的常識を問うタスクであり，モデルは時間に関する常識的知識を獲得する必要がある．この観点から，MLM において，時間関係の単語をマスクすることにより，特に時間に関する知識を獲得することを目指す．

時間関係の単語は，MC-TACO のデータを確認し手動で定義する．主な内容としては，数字，形容詞/副詞/前置詞（often, before, after, every など），時間の単位（hours, years など）である（付録 A 参照）．

2.4 対象タスクの特性

対象とするデータセットの形式に着目したマスク方法を考える．MC-TACO の 1 サンプルの入力は，文章＋質問＋答えで構成されている．このうち，答えの部分がもっとも二値分類タスクの予測結果に関わる部分であり，モデルの学習に重要であるという仮定に基づき，答えの部分を優先してマスクする．

2.5 注意機構（Attention）

注意機構（Attention）[6] の値が大きい単語に着目したマスク方法を考える．他の単語からの Attention の値が大きい単語は，他の単語からの注目度が高いということになり，入力された単語の中でも文脈の理解に大きく貢献していると考えられる．

2.6 損失に対する顕著性（Saliency）

損失に対する顕著性（Saliency）の値が高い単語に着目したマスク方法を考える．Saliency は各単語の埋め込みベクトルが最終的な判断にどれだけ寄与しているかを表す指標である [7]．Saliency の値が高い単語は入力文の中で重要な単語と言えるため，その値に着目して実験を行う．

Saliency を求める際には損失と単語の埋め込みベクトルの勾配を求めることになるが，今回使用するモデルである BERT [1] の入力を構成する，Token Embeddings, Position Embeddings, Segment Embeddings の 3 種類の埋め込みベクトルのうち，全てを使用する場合と，Token Embeddings のみを使用する場合の 2 通りで Saliency を求める．以下，前者を求め方 1，後者を求め方 2 とする．

3 実験

TAPT によって精度が改善することを確認した後に，マスクする対象の選び方を変更した実験を行う．また，作成したモデルを MC-TACO 以外のデータに用いた性能評価を行い，その結果を分析する．

3.1 実験設定

MC-TACO でのファインチューニングおよび TAPT を行う際のパラメータの設定を表 1 に示す．MCTACO に対しては，Hugging Face の Transformers[1)] で提供されている分類問題用のモデルである，BertForSequenceClassification モデルを使用する．MLM 及び NSP には，BertForPreTraining モデルを使用する．

表 1 実験設定

	max seq_len	train batch_size	num train_epoch	learning rate
standard fine-tuning	128	16	5	1e-5
TAPT	128	32	3	3e-5

本研究において，モデルには bert-base-uncased を使用し，評価指標としては Exact Match (EM) と F1 スコアを採用した．EM は各質問に対する全ての答えを正しくラベル付けすることができる確率であり，F1 スコアは適合率と再現率の調和平均である．

3.2 実験結果

まず，通常通りファインチューニングを行なった場合と，対象のデータセットを用いた Masked Language Modeling と Next Sentence Prediction を TAPT として実験を行った場合の結果を比較する．なお，ここでのマスク方法はランダムである．結果を表 2 に示す．

ここからの実験結果の記載方法については，MCTACO の評価データを使用した結果と，() 内には 5 分割交差検証を行なった結果を記載している．

表 2 ベースラインと TAPT の比較

内容	EM [%]	F1 [%]
standard fine-tuning	40.9 (42.1)	69.9 (68.2)
TAPT	**44.5 (45.2)**	**71.9 (72.4)**

こちらに示した通り，TAPT を行うことでベースラインよりも精度が上がることがわかる．

1） https://github.com/huggingface/transformers

表 3　時間関係の単語を優先してマスク

Masking Probability（時間関係の単語）[%]	Masking Probability（その他の単語）[%]	EM [%]	F1 [%]
100	0	42.7 (46.6)	71.0 (71.7)
90	10	44.1 (43.3)	71.6 (70.1)
80	20	**45.1 (44.3)**	**72.7 (70.7)**
70	30	42.9 (42.6)	71.9 (69.5)

表 4　答えの部分の単語を優先してマスク

Masking Probability（答えの部分の単語）[%]	Masking Probability（その他の単語）[%]	Masking Probability（全体）[%]	EM [%]	F1 [%]
100	0	11.9	**44.7 (45.2)**	**71.2 (73.1)**
100	5	16.3	43.6 (42.5)	71.0 (70.9)
90	10	19.5	43.2 (44.8)	71.0 (69.7)

以下に，提案手法における MLM におけるマスクの方法，マスクする単語の選び方を変更し，各実験の結果を示す．なお，今後の表内において，表 2 に記載のランダムでマスクした場合の TAPT よりも精度が改善した場合は，太字で結果を記載している．

対象タスクのトピック　対象タスクのトピックに関連する時間関係の単語を高い割合でマスクする設定で実験を行なった．結果を表 3 に示す．実験の結果，ランダムでマスクした場合よりも精度の向上が確認できた．

対象タスクの特性　MC-TACO の各サンプルの構造に着目して，答えの部分を高い割合でマスクする設定で実験を行なった．結果を表 4 に示す．実験の結果，ランダムでマスクした場合とほぼ同等の精度が確認された．

ここで，答えの中に含まれるかつ時間関係の単語を高い割合でマスクするという，上二つを組み合わせた設定でも実験を行なった．結果を表 5 に示す．実験の結果，ランダムでマスクした場合よりも精度の向上が確認できた．また，二つを組み合わせることにより，単一で基準として採用するよりも精度が向上した．

注意機構（Attention）　Attention の値に着目し，11 番目と 12 番目のレイヤーにおいて，他の単語からの Attention の値が大きい単語，つまり他の単語からの注目度が高い単語を高い割合でマスクする設定で実験を行なった．レイヤーの選び方については，全 12 層のうち，上のレイヤーの Attention の方が最終的な予測に大きく関係しているのではないかという考えに基づいた．結果を表 6 に示す．

実験の結果，マスクの割合を 15%に設定した際の精度はランダムでマスクした場合と同等である事が確認できた．

損失に対する顕著性（Saliency）　2.6 節で述べた 2 通りの方法で Saliency の値を求め，その値が高い単語上位 15 %をマスクする設定で実験を行なった．結果を表 7 に示す．

実験の結果，3 つの埋め込みベクトルのうち Token Embeddings のみを使用して Saliency を求めた場合の精度は，ランダムでマスクした場合より向上することが確認できた．

ここで，Saliency が高い単語に加えて，Saliency が高い単語からの Attention が大きい単語も優先してマスクするという，二つを組み合わせた設定でも実験を行う．これは，単語単体の指標である Saliency に加えて，Attention を用いることによって単語同士の関係性も考慮するためである．ここでの Saliency は，Token Embedding のみを使用する方法を採用する．結果を表 8 に示す．実験の結果，単語単体の Saliency のみを使用した場合よりも精度が少し下がってしまうことが確認された．

汎化性能評価　今までは MC-TACO においてモデルの性能評価を行なっていたが，他のデータセットにおいても性能評価を行い，時間情報処理のための汎用言語モデルとしての汎化性能を評価する．今回は，他のデータセットとして，TimeML [8, 9] と MATRES [10] を採用する．時間的常識を問う MC-TACO とは異なり，TimeML はイベントの期間，MATRES はイベントの順番を問うタスクで構成されている．性能評価には，ランダムでマスクをする TAPT によって作成されたモデルと，時間関係の単語をマスクする TAPT によって作成されたモデルを使用する．結果を表 9 に示す．

実験の結果，BERT モデルをそのまま使用した場合と，著者らが作成したモデルで同程度の精度となった．

表 5　答えの中の時間関係の単語を優先してマスク

Masking Probability（答えの中の時間関係の単語）[%]	Masking Probability（その他の単語）[%]	Masking Probability（全体）[%]	EM [%]	F1 [%]
100	0	6.0	42.5 (45.1)	69.6 (71.8)
100	10	15.4	**45.7 (44.8)**	**71.9 (72.0)**
100	20	24.8	43.2 (44.3)	71.0 (71.5)
80	20	22.4	44.2 (44.1)	71.3 (71.3)

表 6　Attention の値が大きい単語を優先してマスク

Masking Probability [%]	EM [%]	F1 [%]
15	44.1 (43.9)	71.1 (71.4)
30	41.4 (43.8)	69.9 (70.1)
45	42.4 (43.3)	71.0 (69.1)
60	41.5 (41.1)	70.2 (68.7)

表 7　Saliency の値が高い単語を優先してマスク

Saliency の求め方	EM [%]	F1 [%]
求め方 1	44.2 (44.0)	71.6 (71.2)
求め方 2	**44.8 (42.7)**	**72.2 (70.4)**

表 8　Saliency が高い単語＋ Attention が高い単語を優先してマスク

Masking Probability [%]	EM [%]	F1 [%]
15	44.1 (42.7)	71.3 (69.5)
30	44.1 (42.9)	71.1 (70.3)

表 9　汎化性能評価

使用モデル	TimeML acc [%]	MATRES acc[%]
standard fine-tuning	**82.6**	71.7
TAPT(random)	81.9	**72.6**
TAPT（時間関係）	82.2	71.6

3.3　考察

マスク言語モデルにおけるマスク対象をランダムな選択から様々な設定に基づく選択で実験を行なってきたが，最も結果が良かったのは答えの中の時間関係の単語を高い割合でマスクした場合であり，次に良かったのは，データセット全体において時間関係の単語を高い割合でマスクした場合であった．これらはランダムでマスクした場合よりも高い精度を出している．このことは，マスクする単語を慎重に選択することが性能に影響すること，対象タスクのトピックに合わせて単語を選択しマスクをすることで，タスクを解くのに必要な知識を重点的に獲得できることを示している．また，Saliency の値に着目した実験設定では，Token Embeddings のみを使用して Saliency の値を計算することによって選択した単語をマスクした場合に，ランダムでマスクする設定よりも精度が向上した．Saliency の値は算術的に求められるものであり，今後も詳しく検討する価値のある手法であると考えている．

一方で，Attention の値に着目した実験設定では，ランダムでマスクした場合と同等程度の性能となった．これは，予測に関する値が高い単語をマスクすることは，ランダムにマスクする場合以上のタスクを解くのに必要な知識の獲得には繋がらない可能性を示唆している．このことは，Transformer の Attention は最終的な識別には大きな貢献をしていないという研究報告 [11] にも合致している．

MC-TACO 以外のデータセットを使用した汎化性能の評価では，精度が改善しなかった．これは作成したモデルが MC-TACO のタスクを解くのに適したものになってしまっているからだと考えられ，現在の手法では汎用性に欠けることがわかった．これに対しては，今後，複数のデータセットのタスクを同時に解くマルチタスク学習などを通じて汎用性を高めることを検討したい．

4　おわりに

本研究では，対象のデータセットを用いた Masked Language Modeling（MLM）と Next Sentence Prediction をタスク適応事前学習（TAPT）として実験を行った．また，MLM においてマスクする対象となる単語の選び方を従来のランダムの選択から様々な基準に変更して実験を行なった．提案モデルを難易度の高い multiple choice temporal common-sense (MC-TACO) [4] で評価したところ，標準的なアプローチよりも性能が大幅に向上していることを確認できた．中でも，対象タスクのトピックに合わせて単語をマスクすることにより精度が大きく改善し，タスクを解くのに必要な知識を獲得することができたと考える．今後は，他のデータセットやモデルにおいても同様に精度が向上するよう，汎化性能を高めていきたいと考えている．

謝辞

本研究は，科研費（18H05521）の支援を受けた．ここに謝意を表す．

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] Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*, 2020.

[3] 木村麻友子 Lis Kanashiro Pereira, 小林一郎. 時間的常識を理解する言語モデルの構築へ向けて. 言語処理学会年次第 27 回大会, pp. 1193–1198, March 2021.

[4] Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. "going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3363–3369, Hong Kong, China, November 2019. Association for Computational Linguistics.

[5] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8342–8360, Online, July 2020. Association for Computational Linguistics.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. 2017.

[7] Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. Visualizing and understanding neural models in NLP. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 681–691, San Diego, California, June 2016. Association for Computational Linguistics.

[8] Roser Saurí, Jessica Littman, Bob Knippen, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. Timeml annotation guidelines. *Version*, Vol. 1, No. 1, p. 31, 2006.

[9] Feng Pan, Rutu Mulkar-Mehta, and Jerry R Hobbs. Extending timeml with typical durations of events. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pp. 38–45, 2006.

[10] Qiang Ning, Hao Wu, and Dan Roth. A multi-axis annotation scheme for event temporal relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1318–1328, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[11] Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7057–7075, Online, November 2020. Association for Computational Linguistics.

A 付録

2.3 節に記載している時間関係の単語について，そのリストを記載する．このリストに数字（0,1,2,...）を追加したものを使用した．

['after', 'afternoon', 'afterwards', 'age', 'ago', 'already', 'always', 'am', 'annually', 'april', 'august', 'before', 'between', 'billion', 'centuries', 'century', 'current', 'currently', 'daily', 'day', 'days', 'decade', 'decades', 'during', 'earlier', 'early', 'eight', 'eternity', 'evening', 'eventually', 'ever', 'every', 'everyday', 'fifteen', 'fifty', 'first', 'five', 'for', 'forever', 'four', 'fourth', 'frequency', 'friday', 'from', 'future', 'hour', 'hourly', 'hours', 'hundred', 'hundreds', 'in', 'january', 'june', 'just', 'later', 'long', 'march', 'midnight', 'millions', 'minute', 'minutes', 'monday', 'mondays', 'month', 'monthly', 'months', 'morning', 'mornings', 'never', 'night', 'noon', 'now', 'often', 'once', 'one', 'overnight', 'past', 'per', 'pm', 'previously', 'prior', 'quickly', 'rarely', 'saturday', 'season', 'second', 'seconds', 'september', 'seven', 'several', 'since', 'six', 'sometime', 'spring', 'still', 'summer', 'sunday', 'ten', 'then', 'third', 'thirteen', 'thirty', 'thousand', 'three', 'time', 'times', 'today', 'tomorrow', 'tuesday', 'twenty', 'two', 'until', 'usual', 'usually', 'wednesday', 'week', 'weekday', 'weekdays', 'weekend', 'weekends', 'weekly', 'weeks', 'when', 'whenever', 'while', 'will', 'within', 'year', 'yearly', 'years', 'yesterday', 'yet', 'zero']

また，本文に記載の実験以外にもいくつかの設定で実験を行なったので，こちらに記載する．

注意機構（Attention） MC-TACO で通常通りファインチューニングしたベースラインモデルと，今までに作成して精度が改善したモデルとの間で，レイヤー 11,12 における他の単語からの Attention の値の上がり幅が大きい単語を高い割合でマスクする．これは，精度が改善しているということは Attention の当たり方も改善されており，上がり幅が大きい単語は精度の改善に大きく影響している単語なのではないかという考えに基づいている．今まで作成したモデルとして，先行研究 [3] で作成したモデル 2 つを使用する．Multi-Step Fine-Tuning に SWAG を使用したものと，ランダムでマスクした TAPT である．結果を表 10 に示す．今回の実験設定では，ランダムでマスクした場合よりも精度が下がってしまうことが確認された．

表 10 2 つの手法間で Attention の値の上がり幅が大きい単語を優先してマスク

Masking Probability [%]	EM [%]	F1 [%]
ベースライン（fine-tuned on MC-TACO）	40.9 (42.1)	69.9 (68.2)
ベースラインと Multi-Step Fine-Tuning（SWAG）		
15	41.2 (42.7)	69.5 (70.6)
30	43.5 (42.7)	71.2 (70.6)
45	43.0 (41.2)	71.0 (68.9)
60	42.2 (39.9)	71.1 (68.4)
ベースラインと TAPT（ランダムにマスク）		
15	42.2 (42.3)	70.8 (70.2)
30	42.4 (42.5)	70.5 (69.2)
45	42.3 (42.3)	70.8 (69.6)
60	41.4 (43.6)	69.4 (70.1)

損失に対する顕著性（Saliency） Saliency の値が高い単語ではなく，高くも低くもない真ん中の単語を高い割合でマスクする設定でも実験を行なった．結果を表 11 に示す．実験の結果，精度は上がらず，Saliency に関しては値が高い単語に着目する方が良い結果が出ることがわかった．

表 11 Saliency の値が高くも低くもない単語を優先してマスク

Saliency の求め方	EM [%]	F1 [%]
求め方 1	43.8 (42.6)	71.3 (68.7)
求め方 2	44.2 (44.0)	71.6 (71.2)