TaCOMET: 時間を考慮したイベント常識生成モデル

村田栄樹　河原大輔
早稲田大学理工学術院
{eiki.1650-2951@toki., dkw@}waseda.jp

概要

常識的知識は言語モデルの通常の学習では獲得されにくい. そのため, 人手で常識を収集した常識知識グラフやそれらをニューラル化した常識生成モデルが構築されている. しかし, イベントを扱う既存の常識生成モデルは, 粒度や時間を考慮していない. 我々は時間を考慮した常識生成モデル TaCOMET を提案する. まず時間付きの常識知識グラフ TimeATOMIC を構築し, これを用いて既存の常識生成モデルを継続訓練することで TaCOMET を構築する. 評価実験において, TimeATOMIC や継続学習により時間を考慮した生成がなされることを確かめた. さらに, ロボットの意思決定タスクにおいて TaCOMET の応用可能性を確かめた.

1　はじめに

言語モデルは事前学習により, 言語知識 [1, 2] や事実的知識 [3, 4] を獲得する. しかし, これらは人間によって発せられた言語の表面のみから学習するため, 暗黙的になりがちな常識的な知識を上手く扱うことができない [5, 6].

そこで, 常識的な知識を主に人手で収集した常識知識グラフが構築されてきた. 代表的なものとして, エンティティの関係を扱った ConceptNet [7] やイベントやメンタルステートを扱った ATOMIC [8] などがある. 常識知識グラフはシンボリックに知識を収集するため, そのカバレッジは有限である. ATOMIC などの知識を言語モデルのパラメタに保存する COMET [9] などの常識生成モデルもある. 本論文ではこれらのイベントについての常識知識グラフを扱う.

知識グラフから, モデル内に知識を蓄える COMET などへの移行によってカバレッジの課題は軽減したものの未だ完全ではない. 大きな問題の一つに, イベントは粒度をもっているにも関わらず既存の常識生成モデルはそれを考慮しないことがある. イベントの粒度には, エンティティや動作の全体部分関係, 因果関係, イベントの持続時間やイベント間の時間など様々な側面がある [10]. 例えば, “go to school” というイベントに対して (1) “学校という場所に行く” や (2) “教育を受けている” のような粒度の異なる解釈が可能である [11]. イベント “go to school” の後に続くイベントとして, (1) の解釈ならば “go to the library”, (2) ならば “get a job” のように粒度の解釈によって推論に曖昧性が生じる. これは常識生成モデルの訓練や評価において問題となる.

これを解決するために, 我々は時間を考慮した常識生成モデル **TaCOMET** (**T**ime-**a**ware **COMET**) を提案する. イベントの粒度の最も重要な要素の一つとしてイベント間の時間に着目する. TaCOMET は時間の入力によって推論生成をコントロールできる常識生成モデルである.

モデルの構築は 2 つのステップからなる. 第一に, ChatGPT [1)] を使用して既存の知識グラフにイベント間の時間を付与したデータセット (**TimeATOMIC**) を作成する. 第二に, TimeATOMIC で既存の常識生成モデルを追加訓練する. このシンプルだが効果的な手法により, 入力時間に対応した知識を生成するモデルを構築する.

本研究では日本語で実験する. 構築したモデルが入力時間に対応した妥当な推論を生成することを自動評価と人手評価によって示した.

また, 構築したモデルの有効性を下流タスクにおいても確かめる. タスクとしては対話ロボットの意思決定に関するデータセットを使用する. 我々のモデルの性能は時間を付与しないモデルを上回り, ロボットにとって適切であろう時間を入力することで性能が向上することを確認した.

2　関連研究

常識知識グラフ　テキストに現れない暗黙的な知識を扱うために常識知識グラフが構築されてきた.

1)　https://chat.openai.com/

人手で構築された代表的な常識知識グラフとして, 主にエンティティ間の関係を扱ったグラフ形式の WordNet [12] や ConceptNet [7] がある.

本研究ではイベント間の常識を扱う. イベントの関係に着目した常識知識グラフとして, ATOMIC [8] がある. ATOMIC はイベントやメンタルステートの間の if-then 関係を, (“X makes Y’s coffee”, xEffect, “X gets thanked”) のように 3 つ組の形式で収録する. 本論文では, 3 つ組のうち推論の元となる第 1 項目を head, 推論のタイプを示す第 2 項目を relation, 推論先となる第 3 項目を tail と呼ぶ. ATOMIC と ConceptNet を統合・拡張した ATOMIC2020 [6] もある. 人手で構築された常識知識グラフのカバレッジ問題に対処するために, 自動構築手法が提案されてきた. ルールに基づくもの [13, 14] や, GPT-3 で ATOMIC2020 を拡張した ATOMIC10x [15] がある. Dense-ATOMIC [16] は既存の常識知識グラフに対して, relation をモデルで予測することでカバレッジの問題を軽減した. しかし, 常識知識グラフには, 未だ構築コストの高さや有限のカバレッジという課題がある.

常識生成モデル　常識知識グラフで GPT-2 などの言語モデルを訓練することで, そのパラメタの中に常識知識グラフの知識を蓄える常識生成モデルが構築されてきた. これらは常識知識グラフのカバレッジの限界の課題を軽減する. COMET [9] は ATOMIC や ConceptNet で Transformer を訓練した常識生成モデルで, 常識知識グラフに含まれない未見の入力に対しても常識を生成し, カバレッジの限界を取り払った. ATOMIC2020 と ATOMIC10x のそれぞれで訓練された COMET2020 [6] や $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ [15] もある. $\text{COMET}^{\text{DIS}}_{\text{TIL}}$ はベースモデルが GPT-2 であるにも関わらず GPT-3 を上回る精度で常識を生成することができる. 英語以外の言語を対象にした研究 [17, 18] もある.

また, COMET の知識を一般の言語モデルに転移することで, 一般の言語能力を保持したまま常識タスクの性能を向上させることもできる [19].

3　Time-aware COMET

既存のイベントに関する常識知識グラフや常識生成モデルがイベントの粒度を扱っていないことを受けて, 我々は常識生成モデルに時間の知識を付与したモデル TaCOMET を提案する. 粒度を扱うことで, 推論や評価における曖昧性を軽減し, 下流タスクへの応用の幅も広がると考える.

3.1　TimeATOMIC 構築

まず, 時間的常識を付与するためのデータセットを日本語で構築する. イベントの遷移を扱うため, relation としては以下の 2 つを採用する.

- **xNeed**: X が head の前にするイベント
- **xEffect**: X が head の後にするイベント

データセットの構築には gpt-3.5-turbo[2] (以降, ChatGPT) を使用する. ChatGPT に head, relation に加えて time を入力し tail を出力する.

ChatGPT に入力する head に使用するイベントとして, ATOMIC-ja [17] の xNeed と xEffect に対する head と tail からランダムに 2,000 件を選択する. 入力する time は, 適当な候補集合 (付録 B.2) からランダムに選ぶ. それぞれの head に対して, 2 つの relation があり, さらにそれぞれ 3 つの time を付与する. つまり合計で 12,000 件の (head, relation, time) を ChatGPT に入力する.

非現実的な head と time のペアなどに対しては ChatGPT が生成を拒むケースがあるため, 簡単なフィルタリングを施す. その結果, 11,249 件のデータを得た. ChatGPT の利用に約 6 ドルかかった.

3.2　ファインチューニング

3.1 節で構築したデータセットを用いて既存の COMET を継続訓練する. 既存の COMET を使用することで, 通常の言語モデルは持たない, 人間から得た常識知識を利用する.

フォーマット　それぞれのサンプルに対して head, relation, time を入力したときの tail を出力するように訓練する. 既存の COMET のフォーマットを可能な限り維持したまま, time を挿入する. 損失計算は tail に対してのみ行う.

モデル　COMET ベースモデルとして, COMET-ja [17] を使用する. およそ 200K の ATOMIC で訓練された COMET である. GPT-2-xl のアーキテクチャを採用しており, パラメタ数はおよそ 1.5B である.

また, 異なるモデルサイズの実験として, GPT-2-small をベースとした COMET-ja でも実験する. 同様に 200K の ATOMIC で訓練された, パラメタ数が 110M のモデルである.

手法の比較として, COMET を経由せずに, GPT-2 を直接, 時間付きデータセットで訓練したモデルも用意する. COMET を訓練する TaCOMET と区別し

2)　2023 May 12 version.

表 1: TaCOMET の評価結果. **太字**は最良値. BS_{INNER} のみ小さい方が良い.

Size	Model Type	$BS_{GEN-REF}$	BS_{INNER} ↓	$\rho_{spearman}$	$\rho_{pearson}$	$\rho^{log}_{pearson}$	valid%
XL	TaCOMET	**0.754 ± 0.119**	0.763 ± 0.102	**0.473**	**0.065**	**0.458**	**0.784**
	$TaCOMET_{SCRATCH}$	0.752 ± 0.118	**0.760 ± 0.103**	0.467	0.049	0.454	0.771
	COMET	0.569 ± 0.068	0.818 ± 0.154	0.154	0.026	0.153	0.722
small	TaCOMET	**0.605 ± 0.071**	0.715 ± 0.129	0.450	**0.131**	0.438	**0.682**
	$TaCOMET_{SCRATCH}$	0.604 ± 0.072	**0.711 ± 0.132**	**0.467**	-0.008	**0.444**	0.640
	COMET	0.571 ± 0.063	0.873 ± 0.173	0.062	0.022	0.059	0.344

て, このモデルを $TaCOMET_{SCRATCH}$ と呼ぶ.

3.3 評価指標

構築したモデルでテストセットに対して推論を行い, 推論結果に対して自動評価と人手評価を行う. “[head][relation] [time] [前 | 後],” を入力し続きとなる tail を生成させることで推論する.

自動評価 テキスト間の類似度計算方法である BERTScore [20] を用いて, 2 つの自動評価を行う.

一つ目の自動指標は TaCOMET の生成と参照文の間の類似度 ($BS_{GEN-REF}$) である. これは, 通常の生成タスクにおいてモデルの性能を測る指標であり, 数値が高いほどよい. 二つ目は時間入力を変更したときの, 生成間の不一致度 (BS_{INNER}) である. 同じ head, ralation に対して時間のみが異なる入力をしたときの TaCOMET による生成同士の類似度を測る. 入力する時間に応じて異なる生成をすることが求められるので, この値は小さい方がよい.

人手評価 クラウドソーシング[3)]によって, 人手評価を行う. TaCOMET の生成に関して, 二つのラベルを作成する. 入力された head と生成された tail に対して想定される time を一つ目のラベルとして付与する. 時間幅は提示せずに同様のペアに対してその推論が適切か否かを二つ目のラベルとする. 二つのラベルをもとに, 以下の 4 つの指標を求める.

時間幅について, COMET へ入力されたデータセット内のものとクラウドソーシングによるラベルの二つの系列の間で相関を測る. スピアマンの順位相関係数 ($\rho_{spearman}$) とピアソンの積率相関係数 ($\rho_{pearson}$) に加えて, 両対数変換を施した系列に対する積率相関係数 ($\rho^{log}_{pearson}$) の 3 つを算出する.

時間幅を考慮せずにそもそも妥当な推論を行っているかを確認するために, 適切なラベルを付与された推論の割合 (valid%) も算出する.

3.4 結果・考察

評価結果を表 1 に示す. 各ベースモデルに対して, 3.2 節で訓練した 2 つにベースラインを加えた, 以下の 3 つの設定を比較する.

- TaCOMET: COMET + TimeATOMIC
- $TaCOMET_{SCRATCH}$: GPT-2 + TimeATOMIC
- COMET: 訓練なしのベースライン

BERTScore 生成タスクとしての性能を示す $BS_{GEN-REF}$ は, 訓練によって向上した. TaCOMET と $TaCOMET_{SCRATCH}$ のどちらも, 我々のデータセットに適応させることができた.

時間ごとの生成の違いを示す BS_{INNER} についても, 訓練によってより良い結果を得た. 同じ head と relation に対しても入力する時間を変更することで, それに応じた適切な生成を使い分けることができていると言える. $BS_{GEN-REF}$, BS_{INNER} のどちらも, TaCOMET と $TaCOMET_{SCRATCH}$ の間では差が見られなかった.

相関係数 順位相関係数は, COMET ではほとんど 0 に近いが, 訓練したモデルでは 0.4 から 0.5 程度の値で, 正の相関が確認された. ただ時間によって生成を変えるだけではなく, その大小によって適切な tail を生成していると言える.

積率相関係数について, もとの系列ではほとんど相関は見られなかった. 両対数変換後は, 訓練をしたどのモデルにも正の相関が見られた.

Valid% 時間軸を無視して, 推論が適切か確認する. valid%について, XL サイズのモデルは 75% 以上, small でも 60%を超えている. 時間を付与しても本来の COMET としての性能も維持しているといえる. しかしどのベースモデルでも TaCOMET と $TaCOMET_{SCRATCH}$ の間には有意差は見られなかった.

4 下流タスクでの検証

提案するモデルが下流タスクにも転用できる例の一つとして, ロボットの意思決定用のデータ

3) https://crowdsourcing.yahoo.co.jp/

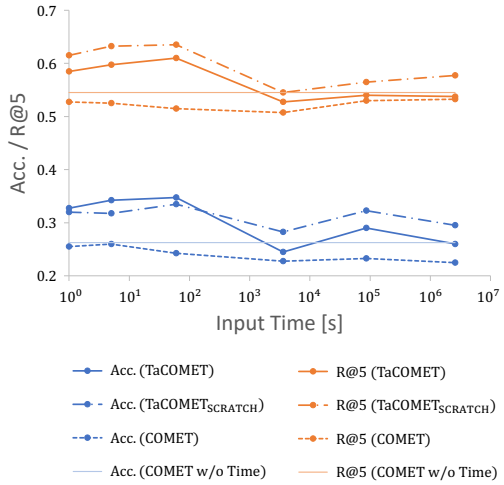

図 1: ロボットデータでのテストの結果.

セットでテストする. 家庭内ロボットの意思決定に TaCOMET の生成確率を使用し, 時間入力の効果の検証や手法の比較を行う.

4.1 データセット

使用するデータは, 曖昧な発話に対する家庭内ロボットの意思決定能力を測定するために作成された [21]. 発話として, 直接的な指示ではなく独り言のような曖昧な発話を収集しているため, 常識的な推論をすることが求められる. データは 400 件あり, ロボット視点の画像とユーザの発話, 画像から読み取れる description を入力とし, 40 個の action から正解を選ぶ分類タスクとなっている. COMET を分類に使用した研究 [22] もある.

4.2 方法

TaCOMET の生成確率を利用して action を分類する. TaCOMET のテンプレートは “[head] xEffect [time] [前 | 後], [tail]” である. 発話と action をそれぞれ TaCOMET の head と tail とし, action 部分の生成確率を計算する. これを action リストの 40 個全てに行い, 生成確率が最大の action をモデルの予測とする.

1 秒から 1 分のような, 家庭内ロボットとユーザのやりとりとして自然な time を与えた時に, xEffect として自然になり性能が上がると考える. 一方で, 1 日や 1 ヶ月のような不自然な入力に対しては, 答えが定まりにくいと考える.

4.3 実験設定

入力する time を変化させて, 適切な時間を入力した場合に性能が上がり, 現実的ではない時間を入力すると性能が下がることを確認する. 代入する time は {1 秒, 5 秒, 1 分, 1 時間, 1 日, 1 ヶ月 } の 6 つとする.

日本語の XL モデルで実験する. 3.4 節と同様に TaCOMET, TaCOMET$_{SCRATCH}$ と COMET で上記のフォーマットで実験する. また, 時間を入力しない COMET とも比較する.

40 個の action を生成確率でソートした時に, 上位 1 位と上位 5 位以内に正解がある割合をそれぞれ Accuracy (Acc.) と Recall at 5 (R@5) として算出する.

4.4 結果

図 1 に結果を示す. まず, 訓練したモデルは 1 秒から 1 分の入力で Acc. と R@5 がともに時間入力なしと比較して上回った. 適切な時間を与えることで時間に応じた予測を行い, 時間なしのモデルを超えることができた. 逆に 1 時間以上の入力をしたときは, 時間入力なしと同じかそれ以下の結果となった. 時間感覚を捉えているからこそ, 不自然な時間入力によって action の予測に影響した.

TaCOMET と TaCOMET$_{SCRATCH}$ を比較する. Acc. について, 1 秒から 1 分の間は TaCOMET の方が優れており, 1 時間以上では TaCOMET が劣っている. 自然な入力と不自然な入力によって結果が大きく変化しており, TaCOMET の方が時間をよりよく捕捉して時間入力に対応しているといえる. R@5 に関しては TaCOMET$_{SCRATCH}$ が TaCOMET を常に上回った. 1 時間以上の範囲では性能が下がることが想定されるが, TaCOMET$_{SCRATCH}$ は常に時間なしのベースラインを上回り続けている. 一方で TaCOMET はその範囲ではベースラインを下回っており, より時間を意識できているといえる.

COMET に時間を入力したものは, フォーマットの変化に対応できず時間なしのモデルを下回った.

5 おわりに

本論文では, 既存の常識知識グラフが粒度を扱わないことを受けて, 時間を考慮した常識生成モデル TaCOMET を提案した.

今後の展望として, コンテキストの考慮や relation の拡張, 粒度の他の側面の採用, マルチモーダル化がある. また, より幅広い下流タスクでのテストや一般の言語モデルへの知識移行などを含めたより一般的なタスクへの応用も考えられる.

謝辞

本研究は SB Intuitions 株式会社と早稲田大学の共同研究により実施した.

参考文献

[1] John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In **NAACL HLT 2019**, pp. 4129–4138, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[2] Christopher D. Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. Emergent linguistic structure in artificial neural networks trained by self-supervision. **Proceedings of the National Academy of Sciences**, Vol. 117, No. 48, pp. 30046–30054, 2020.

[3] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In **EMNLP-IJCNLP2019**, pp. 2463–2473, Hong Kong, China, November 2019. Association for Computational Linguistics.

[4] Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. A review on language models as knowledge bases, 2022.

[5] Xuhui Zhou, Yue Zhang, Leyang Cui, and Dandan Huang. Evaluating commonsense in pre-trained language models. **Proceedings of the AAAI Conference on Artificial Intelligence**, Vol. 34, No. 05, pp. 9733–9740, Apr. 2020.

[6] Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. (comet-) atomic 2020: On symbolic and neural commonsense knowledge graphs. **Proceedings of the AAAI Conference on Artificial Intelligence**, Vol. 35, No. 7, pp. 6384–6392, May 2021.

[7] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. **Proceedings of the AAAI Conference on Artificial Intelligence**, Vol. 31, No. 1, Feb. 2017.

[8] Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. Atomic: An atlas of machine commonsense for if-then reasoning. **Proceedings of the AAAI Conference on Artificial Intelligence**, Vol. 33, No. 01, pp. 3027–3035, Jul. 2019.

[9] Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. COMET: Commonsense transformers for automatic knowledge graph construction. In **ACL2019**, pp. 4762–4779, Florence, Italy, July 2019. Association for Computational Linguistics.

[10] Rutu Mulkar-Mehta, Jerry Hobbs, and Eduard Hovy. Granularity in natural language discourse. In **Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011)**, 2011.

[11] Inderjeet Mani. A theory of granularity and its application to problems of polysemy and underspecification of meaning. In **International Conference on Principles of Knowledge Representation and Reasoning**, 1998.

[12] George A. Miller. WordNet: A lexical database for English. In **Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994**, 1994.

[13] Hongming Zhang, Xin Liu, Haojie Pan, Yangqiu Song, and Cane Wing-Ki Leung. Aser: A large-scale eventuality knowledge graph, 2019.

[14] Hongming Zhang, Daniel Khashabi, Yangqiu Song, and Dan Roth. Transomcs: From linguistic graphs to commonsense knowledge. In Christian Bessiere, editor, **Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20**, pp. 4004–4010. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

[15] Peter West, Chandra Bhagavatula, Jack Hessel, Jena Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. Symbolic knowledge distillation: from general language models to commonsense models. In **Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 4602–4625, Seattle, United States, July 2022. Association for Computational Linguistics.

[16] Xiangqing Shen, Siwei Wu, and Rui Xia. Dense-ATOMIC: Towards densely-connected ATOMIC with high knowledge coverage and massive multi-hop paths. In **ACL2023**, pp. 13292–13305, Toronto, Canada, July 2023. Association for Computational Linguistics.

[17] Tatsuya Ide, Eiki Murata, Daisuke Kawahara, Takato Yamazaki, Shengzhe Li, Kenta Shinzato, and Toshinori Sato. Phalm: Building a knowledge graph from scratch by prompting humans and a language model, 2023.

[18] Chenhao Wang, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. CN-AutoMIC: Distilling Chinese commonsense knowledge from pretrained language models. In **Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing**, pp. 9253–9265, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[19] Wangchunshu Zhou, Ronan Le Bras, and Yejin Choi. Commonsense knowledge transfer for pre-trained language models. In **Findings of the Association for Computational Linguistics: ACL 2023**, pp. 5946–5960, Toronto, Canada, July 2023. Association for Computational Linguistics.

[20] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In **International Conference on Learning Representations**, 2020.

[21] Shohei Tanaka, Konosuke Yamasaki, Akishige Yuguchi, Seiya Kawano, Satoshi Nakamura, and Koichiro Yoshino. Do as i demand, not as i say: A dataset for developing a reflective life-support robot. **IEEE Access**, pp. 1–1, 2024.

[22] 山﨑康之介, 田中翔平, 河野誠也, 湯口彰重, 吉野幸一郎. 常識推論に基づく気の利いた家庭内ロボットの行動選択. 言語処理学会第 29 回年次大会発表論文集. 言語処理学会, 2023.

A 具体例

3.1 節で構築した TimeATOMIC および 3.2 節で構築したモデルの生成をそれぞれ表 2 と表 3 に示す. COMET ではほとんど同じ生成しかされないが, 提案手法では多様な生成が見られる.

表 2: TimeATOMIC の例 (head: “X がスーパーへ買い物に行く”).

relation	time	tail
xNeed	10 秒	X が靴を履く
xNeed	2 日	X が財布にお金を入れる
xNeed	1 ヶ月	X が予算を立てるために貯金計画を立てる
xEffect	30 秒	X が財布を取り出す
xEffect	30 分	X が自宅に帰る
xEffect	3 時間	X が家で料理をする

表 3: TaCOMET と COMET の生成例 (head: “X が店まで走る”).

Rel	Time	TaCOMET	COMET
xNeed	30 分	X が運動着に着替える	X が家を出る
xNeed	4 時間	X が運動不足になる	X が家を出る
xNeed	2 日	X がジョギングシューズを買う	X が家から出る
xEffect	4 秒	X が息を切らす	X が財布を落とす
xEffect	10 分	X が息を切らす	X が財布を忘れる
xEffect	4 日	X が筋肉痛になる	X が財布を落とす

B 実験の詳細

B.1 評価指標

3.3 節で使用した BERTScore では RoBERTa-large[4] の最終層を使用した. TF-IDF による重みづけはしていない.

B.2 その他

3.1 節において ChatGPT に入力した時間は, {1,2,3,4,5}×{ 秒, 0 秒, 分, 0 分, 時間, 日, ヶ月 } からランダムに選び, 文字列として連結したものである.

4) https://huggingface.co/nlp-waseda/roberta-large-japanese/