

論文の階層構造を活用した被引用数予測

平子 潤 笹野 遼平 武田 浩一

名古屋大学大学院情報学研究科

hirako.jun.e5@s.mail.nagoya-u.ac.jp

{sasano,takedasu}@i.nagoya-u.ac.jp

概要

日々、大量に発表される論文の中から、重要な論文を自動推薦するため、論文の将来的な被引用数を予測する技術の重要性は増している。被引用数の予測性能の向上に、論文本文の活用は重要な要素だと考えられるが、論文本文は一般に記述量の多さから扱いが難しく、先行研究においても活用法は十分に探索されていない。そこで、本研究では、被引用数予測において、本文を有効に活用することを目指し、論文の階層構造を活用した新たな被引用数予測手法を提案する。さらに、既存手法を含めた、複数の手法との広範囲な比較実験を行い、被引用数予測における提案手法の有効性を示す。

1 はじめに

近年、人工知能分野など様々な分野において論文の発表数が急激に増加している。これに伴い、重要な論文を推薦するため、論文の質を予測する技術の重要性が高まっている。そのため、本研究では、論文の質を近似する指標として被引用数を採用し [1, 2]、論文の将来的な被引用数を予測するタスクに取り組む。

被引用数を予測する上で、論文本文の活用は重要な要素であると考えられる。しかしながら、既存の被引用数予測の研究 [3, 4] では、論文のテキスト情報としてアブストラクトを用いることが多く、論文の本文はほとんど活用されていない。その原因は、本文が非常に長く、扱いが難しいことだと考えられる。特に、近年盛んに利用されている BERT [5] などの Transformer [6] ベースのモデルは、計算量が入力長の 2 乗に比例するため、非常に長いトークン列を入力すると、計算量は極めて大きなものとなる。この問題を解決するため、Transformer の構造自体を改善するアプローチ [7, 8] や、入力文を分割して扱うアプローチ [9, 10] などが提案されているが、被引用数予測における有効性は不明である。

そこで、本研究では、被引用数予測において、本文を有効に活用することを目指し、論文の階層構造を活用した新たな被引用数予測手法を提案する。この手法では、BERT を用いて論文の各節をエンコードし、そのセクション表現を集約することで被引用数の予測を行う。また、arXiv と bioRxiv から収集した論文を用いて、既存手法を含めた複数手法との広範囲な比較実験を行い、提案手法の有効性を示す。

2 関連研究

2.1 被引用数予測

被引用数予測の既存手法は、テキスト情報を利用しない手法と利用する手法に大別することができる。このうちテキスト情報を利用しない手法としては、論文の引用関係をグラフで表現した引用グラフを用いる手法 [11, 12, 13] や、RNN を用いて短い期間の被引用数の推移から長い期間の被引用数を予測する手法 [14] などが挙げられる。

テキスト情報を利用する手法として初期の研究で提案された手法には、アブストラクトの Bag-of-Words を利用する手法 [3] や頻度ベースの重み付けを利用する手法 [15]、LDA より推定されたトピックを利用する手法 [16, 17] などがある。近年では、深層学習を用いて論文のテキストの意味的特徴量を捉える手法も研究されており、タイトルとアブストラクトを Doc2Vec でエンコードして利用する手法 [4] や、事前学習済みモデルを利用して論文の本文全てを活用する手法 [18] も提案されている。

2.2 長い系列を扱う手法

Transformer ベースのモデルは、近年の自然言語処理において盛んに利用される一方で、計算コストが入力長の 2 乗に比例するという問題点があり、長い系列を扱うことは難しい。この問題を解決するために、大きく 2 つのアプローチで研究が行われている。1 つ目は、Transformer の構造自体を改善し、よ

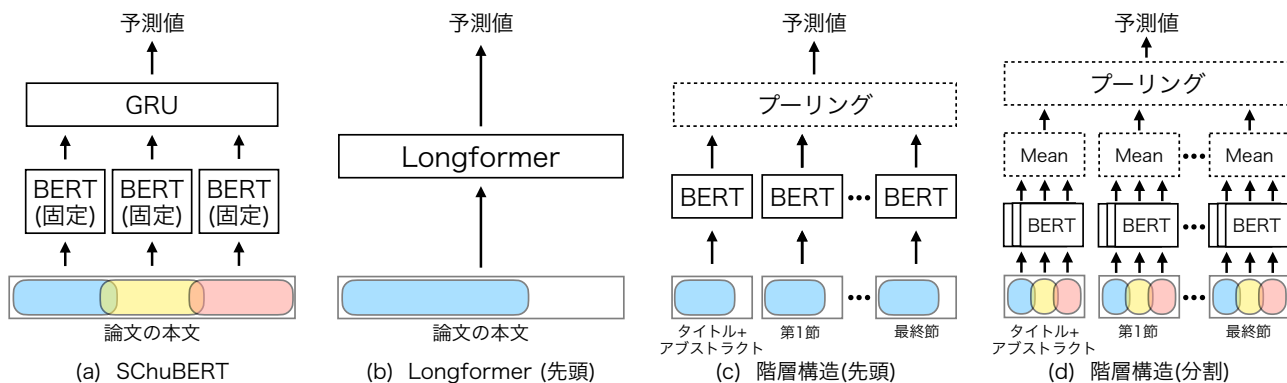


図1 本研究で比較する本文を活用した被引用数予測手法の概要図

り効率的に長い系列を扱うアプローチ [7, 8, 19] である。これらの手法では、主に Self-Attention を改良することで、計算量の改善を目指している。2つ目は、入力する長い系列を分割して処理することで、計算量の増加を抑えるアプローチである。これらの手法には、分割したテキストを BERT でエンコードし、Attention や RNN など比較的単純な手法で集約する手法 [9, 18] や、BERT でエンコードした表現を更に Transformer に入力する階層的な手法 [10, 20] がある。

3 本文を活用した被引用数予測手法

3.1 既存手法: SChuBERT

論文本文から被引用数を予測する手法として、SChuBERT [18] が挙げられる。この手法では、図 1 の (a) で示すように本文をオーバーラップさせながらチャンクに分割し、それぞれを BERT でエンコードして、GRU [21] を用いて集約することで、被引用数を予測する。SChuBERT は BERT の fine-tuning を行わないため、比較的小さな計算コストで、全ての本文を活用することができる。ただし、本文は固定の文字数で分割しており、節の途中で分割されることもあるため、その場合は論文の構造を捉えることができない。

3.2 本文の先頭部分を利用する手法

長い系列を扱うモデルである Longformer [8] を用いて、本文の先頭部分から被引用数の予測を行う手法である Longformer (先頭) の概要を図 1 の (b) に示す。この手法では、Longformer の最大入力長である 4096 に収まるように、論文本文の後半部分を切り捨て、残った前半部分の本文から、被引用数の予測を行う。入力テキストが長い場合に、一部を切り捨てる手法は一般的に採用される [8, 22] 一方で、本文と

いう非常に長いテキストを扱う場合は、多くのテキストが切り捨てられてしまい、論文全体をバランス良く考慮することができない場合が多い。例えば、本実験で使用するデータセット (4.2 節を参照) において、Longformer (先頭) では、平均で本文の約 46% のテキストが切り捨てて入力される。

3.3 階層構造を利用する手法

本研究では、新たに本文の階層構造を活用した 2 つの被引用数予測手法を提案する。これらの手法はどちらも、各節をそれぞれパラメータを共有した BERT を用いてエンコードすることでセクション表現を生成し、各セクション表現をプーリングにより集約することで、被引用数を予測する。ただし、2 つの手法は、各節の先頭部分のみを活用するか、各節を分割し、より多くのテキストを活用するかで異なる。各節の先頭部分を活用する手法である階層構造 (先頭) の概要を、図 1 の (c) に示す。この手法では、各節の見出しを 1 文目、内容を 2 文目として BERT に入力することで、セクション表現を生成し、そこから被引用数を予測する。ただし、2 文目として入力する内容は、BERT の最大入力長である 512 を超えないように、後半を切り捨てる。この手法は、計算コストの増加を抑えることができる一方で、入力できる内容は限定的である。

次に、各節を分割して活用する手法である階層構造 (分割) の概要を、図 1 の (d) に示す。この手法では、まず各節の内容を 50 トークンずつオーバーラップさせながらチャンクに分割し [10, 18]、それぞれを BERT を用いてエンコードする。そして、それらのチャンク表現を Mean プーリングすることで、セクション表現を生成し、そこから被引用数を予測する。また、各チャンクをエンコードする際は、全ての場合で 1 文目に見出し、2 文目にチャン

クを入力する。これは、2つ目以降のチャンクを入力する際に、チャンクのみを入力すると、BERTがパラメータを共有していることから、どの節のチャンクかをBERTが捉えることができないためである。この手法は、階層構造(先頭)と比べ、より多くの内容を扱うことができる一方で、計算コストは増加する。また、計算コストの都合上、各セクションの内容は最大で8つのチャンクに分割し、残りのテキストは切り捨てる。

これらの2つの手法では、各セクション表現をプーリングにより集約することで被引用数を予測する。プーリング手法としては、単純なMeanに加え、Transformerを用いたより表現力が高い手法を採用する。Transformerによるプーリングでは、各セクション表現を系列として1層のTransformerに入力し、その出力の平均を取ることでベクトルを集約する。

4 実験

4.1 タスク設定

本研究では、論文の本文から、その論文の発表から1年後の被引用数を予測するタスクに取り組む。ただし、被引用数 c に対し、 $\log(c+1)$ を予測対象とする[18]。さらに本研究では、先行研究[23]で提案された、現実的な被引用数予測の設定を採用する。これは、学習データとして利用する論文の被引用数の情報を、評価用論文の発表時点に取得できるものに限定する設定であり、この設定を採用することで、より現実に即したモデルの評価が可能となる。この設定では、例えば評価用論文の6ヶ月前に発表された学習用論文は、発表から6ヶ月後の被引用数しか取得できない。すなわち、発表から1年未満の学習用論文は発表から1年後の被引用数が取得できないため、先行研究で提案された手法[23]を用いて被引用数を補完し、学習データとして利用する。

4.2 データセット

本研究では、計算言語学分野の論文からCLデータセット、生物学分野の論文からBioデータセットを構築し、実験を行う。

CLデータセットは、2014年6月から2020年6月の間にarXiv¹⁾のcs.CLカテゴリに投稿された論文で構築しており、被引用数を計算するための引用情報

はSemantic Scholar²⁾から取得する。論文の本文については、論文のHTML版³⁾をパースすることで収集し、HTML版の収集とパースができない論文については、データセットから除外する。また、先行研究[23]に倣い、本研究ではデータセットから13個のサブセットを作成し、学習と評価を行う。具体的には、2019年6月から2020年6月のうちのある月を起点として、そこから過去5年間に投稿された論文で起点別に13個のサブセットを作成し、それぞれのサブセットの最新月に投稿された論文を評価用論文、残りの論文を学習用論文とする。これらのサブセットのうち、2019年6月に発表された論文を評価用論文とするサブセットは、開発セットとして用い、残りのサブセットは評価値の算出に用いる。また、サブセットの平均学習用論文数は11383、平均評価用論文数は449であった。

Bioデータセットは、2015年4月から2021年4月の間にbioRxiv⁴⁾の3つのカテゴリ⁵⁾に投稿された論文で構築し、bioRxivで公開されているHTML版をパースすることで本文を収集する。また、CLデータセットと同様に、Bioデータセットから2020年4月から2021年4月に投稿された論文を評価用論文とする13個のサブセットを作成し、評価用論文が最も古いサブセットは、開発セットとして用いる。これらのサブセットの平均学習用論文数は5737、平均評価用論文数は256であった。

4.3 評価方法

評価指標には、スピアマンの順位相関係数(ρ)、平均二乗誤差(MSE)、実際の上位 $n\%$ 論文が予測値の上位 $k\%$ 論文に含まれる割合($n\%@k\%$)を用いる。また、評価値は、各サブセットでそれぞれモデルの学習、評価用論文に対する予測を行い、全12ヶ月分の予測値を合わせて算出する。全ての設定において、異なるランダムシードで3回実験を行い、評価値の平均と標準偏差を算出する。

また、実験において一部の手法の予測値の平均が、正解値の平均と離れてしまい、MSEでの性能評価に悪影響があることが判明したため、開発セットにおける予測値の平均と正解値の平均の差を補正值とし、評価時の予測値から補正值を引いた値に対しMSEを算出するMSE*も評価指標に加える。

2) <https://www.semanticscholar.org/>

3) <https://ar5iv.labs.arxiv.org/>

4) <https://www.biorxiv.org/>

5) Biochemistry, Plant Biology, Pharmacology and Toxicology

1) <https://arxiv.org/>

表 1 比較実験の結果. 階層構造の括弧内の 2 つ目の要素はプーリング手法を示す. (MSE, MSE*以外の評価値 $\times 100$)

データ	手法	ρ	MSE	MSE*	5%@5%	5%@25%	10%@10%	10%@50%
CL	BERT (title+abst)	37.0 \pm 0.2	1.295 \pm 0.016	1.082 \pm 0.014	28.8 \pm 0.8	69.2 \pm 1.0	34.1 \pm 1.9	83.9 \pm 0.4
	BERT (先頭)	38.5 \pm 0.8	1.246 \pm 0.047	1.028 \pm 0.010	29.0 \pm 2.1	71.5 \pm 1.5	34.6 \pm 1.7	86.0 \pm 1.0
	SChuBERT	37.5 \pm 0.9	0.983 \pm 0.015	0.987 \pm 0.014	24.7 \pm 3.3	62.3 \pm 2.1	29.0 \pm 1.4	82.2 \pm 1.3
	Longformer (先頭)	37.9 \pm 0.1	1.245 \pm 0.027	1.046 \pm 0.014	27.9 \pm 1.4	74.3 \pm 1.6	34.6 \pm 0.3	85.8 \pm 0.3
	階層構造 (先頭 \times Mean)	42.6 \pm 0.1	1.184 \pm 0.025	0.955 \pm 0.006	33.1 \pm 1.7	74.0 \pm 2.0	38.3 \pm 0.4	86.7 \pm 0.9
	階層構造 (先頭 \times Transformer)	43.6 \pm 0.6	1.228 \pm 0.051	0.976 \pm 0.015	33.6 \pm 0.4	76.7 \pm 1.3	38.4 \pm 0.7	88.1 \pm 1.1
	階層構造 (分割 \times Mean)	42.9 \pm 0.5	1.157 \pm 0.005	0.958 \pm 0.009	32.7 \pm 1.1	75.4 \pm 0.6	37.3 \pm 0.5	87.3 \pm 0.9
	階層構造 (分割 \times Transformer)	43.2 \pm 1.2	1.170 \pm 0.013	0.985 \pm 0.019	34.2 \pm 0.9	74.0 \pm 0.8	37.8 \pm 1.7	87.8 \pm 1.0
Bio	BERT (title+abst)	34.7 \pm 0.5	0.517 \pm 0.012	0.518 \pm 0.012	48.8 \pm 0.4	82.1 \pm 0.8	47.4 \pm 1.3	86.0 \pm 1.0
	BERT (先頭)	38.4 \pm 0.7	0.484 \pm 0.010	0.489 \pm 0.009	49.5 \pm 1.0	85.8 \pm 2.3	51.5 \pm 0.0	87.8 \pm 2.1
	SChuBERT	27.2 \pm 4.8	0.663 \pm 0.015	0.613 \pm 0.023	32.2 \pm 2.5	65.6 \pm 2.3	35.5 \pm 2.3	74.4 \pm 3.5
	Longformer (先頭)	39.1 \pm 0.2	0.479 \pm 0.005	0.484 \pm 0.007	50.3 \pm 1.7	84.7 \pm 2.6	49.6 \pm 0.2	86.4 \pm 0.7
	階層構造 (先頭 \times Mean)	39.4 \pm 0.7	0.450 \pm 0.008	0.448 \pm 0.008	51.4 \pm 2.7	86.5 \pm 0.4	51.4 \pm 0.7	90.2 \pm 1.2
	階層構造 (先頭 \times Transformer)	40.9 \pm 0.5	0.452 \pm 0.006	0.453 \pm 0.007	52.7 \pm 1.0	87.8 \pm 1.0	51.4 \pm 0.5	90.8 \pm 0.2
	階層構造 (分割 \times Mean)	38.5 \pm 0.0	0.467 \pm 0.006	0.461 \pm 0.009	51.0 \pm 1.3	86.7 \pm 1.0	49.6 \pm 0.7	88.2 \pm 1.4
	階層構造 (分割 \times Transformer)	40.7 \pm 0.9	0.456 \pm 0.001	0.457 \pm 0.001	52.1 \pm 1.5	89.1 \pm 1.5	50.3 \pm 0.8	91.0 \pm 1.0

4.4 比較手法

本実験では, 3 節で述べた手法に加え, BERT を用いた 2 つの手法を比較する. 1 つ目の **BERT (title+abst)** は, BERT にタイトルを 1 文目, アブストラクトを 2 文目として入力し, 被引用数を予測する手法である. 2 つ目の **BERT (先頭)** は, Longformer (先頭) と同様に, 本文の先頭部分を BERT に入力することで被引用数を予測する手法である.

4.5 実験設定

既存手法である SChuBERT は, ハイパーパラメータの探索範囲を除き, 先行研究 [18] と同様の設定で実験を行う. その他の全ての手法では, BERT と Longformer の出力として, [CLS] トークンに対応するベクトル表現を利用し, モデルが最終的に出力したベクトルを全結合層に入力し, 線形変換を行うことで, 予測値を出力する. また, 予測値と正解値の MSE を損失として学習し, 学習中には, 最終的に出力したベクトルに Dropout [24] を適用する⁶⁾.

4.6 結果

実験結果を表 1 に示す. 既存手法である SChuBERT を除き, 論文の本文を活用した手法は, 両方のデータセットにおいて BERT(title+abst) を上回っており, 被引用数予測において, 本文を活用することの有用性が確認できた. また, 本文を活用する手法の中では, 本研究で提案した階層構造を利用する手法が最も高い性能を達成した. 提案手法

は, 順位相関係数において, その他の全ての比較手法と比べ, CL データセットでは 5.1 ポイント以上, Bio データセットでは 1.8 ポイント以上上回っている. MSE や MSE*, n%@k%においても, ほとんどの場合で比較手法を上回っており, 提案手法が CL と Bio という異なる分野の論文集合に汎用的に有用性を持つことを確認した. 次に, 提案手法において, 階層構造 (先頭) は, 階層構造 (分割) と同等以上の性能を示しており, 各節の先頭部分のみを利用することで, 計算コストの増加を抑えながら, 高い予測性能が達成できることが判明した. また, セクション表現のプーリング手法においても, Mean より Transformer の方が全体的に高い性能を示すことを確認した. 既存手法である SChuBERT は, CL データセットに対する MSE, MSE*は向上したものの, 他の場合では BERT (title+abst) と比べて, 同等以下の結果となった. これは, SChuBERT が本文全文を入力するために, BERT の fine-tuning を行わないことが原因であると考えられる [10].

5 おわりに

本研究では, 論文の階層構造を活用した被引用数予測手法を提案し, 既存手法である SChuBERT や Longformer を利用した手法など複数の手法との比較実験を行い, 提案手法の有効性を示した. 特に, 提案手法において, 各節の先頭部分のテキストのみを利用することで, 計算コストの増加を抑えつつ, 高い性能を達成できることを確認した. 今後の展望としては, 引用グラフなど論文のテキスト以外の情報も同時に扱うモデルの構築をしたいと考えている.

6) 詳細な学習設定は付録 A を参照.

謝辞

本研究の一部は、JST ムーンショット型研究開発事業 JPMJMS2033 の支援を受けたものです。

参考文献

- [1] Daryl E. Chubin and Eugene Garfield. Is citation analysis a legitimate evaluation tool? **Scientometrics**, Vol. 2, pp. 91–94, 1979.
- [2] Dagfinn W. Aksnes. Citation rates and perceptions of scientific contribution. **J. Assoc. Inf. Sci. Technol.**, Vol. 57, pp. 169–185, 2006.
- [3] Alfonso Ibáñez, Pedro Larrañaga, and Concha Bielza. Predicting citation count of bioinformatics papers within four years of publication. **Bioinformatics**, Vol. 25 24, pp. 3303–9, 2009.
- [4] Anqi Ma, Yu Liu, Xiujuan Xu, and Tao Dong. A deep-learning based citation count prediction model with paper metadata semantic features. **Scientometrics**, Vol. 126, pp. 6803 – 6823, 2021.
- [5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **NAACL**, pp. 4171–4186, 2019.
- [6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, **NeurIPS**, Vol. 30, 2017.
- [7] Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In **ICLR**, 2020.
- [8] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. **arXiv**, Vol. abs/2004.05150, , 2020.
- [9] Arash Afkanpour, Shabir Adeel, Hansenclever de F. Bassani, Arkady Epshteyn, Hongbo Fan, Isaac Jones, Mahan Malihi, Adrian Nauth, Raj Sinha, Sanjana Woonna, Shiva Zamani, Elli Kanal, Mikhail Fomitchev, and Donny Cheung. Bert for long documents: A case study of automated icd coding. **arXiv**, Vol. abs/2211.02519, , 2022.
- [10] R. Pappagari, Piotr Żelasko, Jesús Villalba, Yishay Carmiel, and Najim Dehak. Hierarchical transformers for long document classification. **ASRU**, pp. 838–844, 2019.
- [11] Carlos Castillo, Debora Donato, and A. Gionis. Estimating number of citations using author reputation. In **SPIRE**, 2007.
- [12] Feruz Davletov, Ali Aydin, and Ali Cakmak. High impact academic paper prediction using temporal and topological features. 11 2014.
- [13] Nataliia Pobiedina and Ryutaro Ichise. Citation count prediction as a link prediction problem. **Applied Intelligence**, Vol. 44, pp. 252 – 268, 2015.
- [14] Alireza Abrishami and Sadegh Aliakbary. Predicting citation counts based on deep neural network learning techniques. **J. Informetrics**, Vol. 13, pp. 485–499, 2018.
- [15] Lawrence D. Fu and Constantin F. Aliferis. Models for predicting and explaining citation count of biomedical articles. **AMIA**, pp. 222–6, 2008.
- [16] Rui Yan, Jie Tang, Xiaobing Liu, Dongdong Shan, and Xiaoming Li. Citation count prediction: learning to estimate future citations for literature. In **CIKM**, 2011.
- [17] Tanmoy Chakraborty, Suhansan Kumar, Pawan Goyal, Niloy Ganguly, and Animesh Mukherjee. Towards a stratified learning approach to predict future citation counts. **IEEE/ACM Joint Conference on Digital Libraries**, pp. 351–360, 2014.
- [18] Thomas van Dongen, Gideon Maillette de Buy Wenniger, and Lambertus Schomaker. SchuBERT: Scholarly document chunks with bert-encoding boost citation count prediction. In **SDP**, 2020.
- [19] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big Bird: Transformers for longer sequences. **arXiv**, Vol. abs/2007.14062, , 2020.
- [20] Jin Xue, Xiaoyi Tang, and Liyan Zheng. A hierarchical bert-based transfer learning approach for multi-dimensional essay scoring. **IEEE Access**, Vol. 9, pp. 125403–125415, 2021.
- [21] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In **EMNLP**, pp. 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [22] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, **NeurIPS**, Vol. 33, pp. 6256–6268. Curran Associates, Inc., 2020.
- [23] 平子潤, 笹野遼平, 武田浩一. 直近1年の動向を考慮した最新論文のインパクト予測. 言語処理学会第25回年次大会 発表論文集, pp. 1262–1267, 2022.
- [24] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. **JMLR**, Vol. 15, No. 56, pp. 1929–1958, 2014.
- [25] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In **EMNLP: System Demonstrations**, pp. 38–45, 2020.

A 詳細な学習設定

本研究では、BERT の初期重みとして、Transformers [25] で公開されている bert-base-uncased を、Longformer の初期重みとして、allenai/longformer-base-4096 を利用した。また、Transformer プーリングで用いた 1 層の Transformer は、BERT の最終層の重みを用いて初期化した。

既存手法である SChuBERT 以外の手法の学習では、バッチサイズは 32、最適化関数は AdamW を用い、全体の学習ステップの 10% で warm-up し、残りのステップで線形に減衰する学習率スケジューリングを行った。また、エポック数は {3, 4}、学習率は {2e-5, 3e-5, 5e-5} でグリッドサーチを行い、開発セットにおいて最も順位相関係数が高くなった値を利用した。SChuBERT では、エポック数のみハイパーパラメータ探索を行い、残りの設定については先行研究と同様に学習を行った。また、エポック数の探索範囲は、先行研究の値に基づき {20, 30, 40} とした。