E-042

国会会議録を用いた事前学習済み ELECTRA の構築と政治ドメインのタスクによる検証

Construction of Pre-trained ELECTRA Using Minutes of Diet and Validation by Task of Political Domain

永渕 景祐[1)]　木村 泰知[2)]　荒木 健治[3)]
Keiyu Nagafuchi　Yasutomo Kimura　Kenji Araki

概要

事前学習済み Transformer モデルは，特定のドメインのコーパスを事前学習に用いることで，そのドメインのタスクにおいて，高い精度を示すことが知られている．日本語の事前学習済み Transformer モデルにおいても，ビジネスや金融に特化したモデルが，いくつか公開されている．しかし，政治ドメインに特化させて事前学習を行なったモデルは我々の知る限り公開されておらず，研究の余地がある．本研究では，国会会議録から構築したデータセットを用いて，事前学習済み ELECTRA を構築する．また，政治ドメインのタスクによって汎用モデルとの比較を行う．

1　はじめに

近年，政府や地方公共団体は，オープンデータへの取り組みを推進している．オープンデータへの取り組みとは，政府や地方公共団体が保有する公共データを無償利用と二次利用が可能な形式で一般に公開する取り組みのことである．オープンデータへの取り組みの目的は，国民が公共データを活用することを促すことで諸課題の解決，経済活性化，行政の高度化・効率化等を図ることである．オープンデータの多くは，CSV，XML，JSON などの構造化されたフォーマットで公開され，機械判読による情報の抽出・分析が容易に可能となっている．しかし，オープンデータの中には会議録を始めとする膨大な量のテキストを持つデータもあり，そのようなデータに対して機械判読を行うには，自然言語処理技術を用いたアプローチが必要となる．

従来から，自然言語処理の分野において，このような政治ドメインのテキストデータを対象とした研究が行われている．NTCIR-14 QA Lab-PoliInfo[1]，NTCIR-15 QA Lab-PoliInfo-2[2]，NTCIR-16 QA Lab-PoliInfo-3[3] では，政治課題を解決する上で適切な情報を提示することを目的として，複数の Shared task に取り組んでいる．これらの Shared task に参加したチームが提案する手法には，BERT[4] をはじめとする事前学習を行った Transformer モデルを利用したものが多く見られる．このような手法のほとんどが，研究機関が公開している事前学習済みの汎用モデルをタスクに合わせてファインチューニングするというものである．事前学習済み Transformer モデルは，特定のドメインのコーパスを事前学習に用いることで，そのドメインのタスクにおいて，高い精度を示すことが知られている．日本語の事前学習済み Transformer モデルにおいても，ビジネスや金融などのドメインに特化したモデルが，いくつか公開されている．しかし，政治ドメインに特化させて事前学習を行なったモデルは我々の知る限り公開されておらず，研究の余地がある．

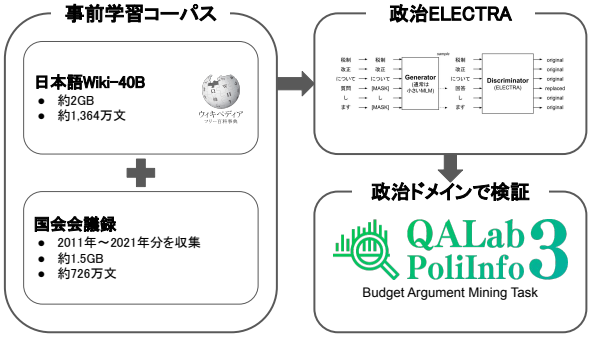

図 1　本研究の概要

そこで，我々は，政治ドメインのテキストから構築した事前学習コーパスを用いて，政治ドメインに特化した事前学習済み Transformer モデルの構築を行うことを目標とする．しかし，政治ドメインのテキストの収集や Transformer モデルの事前学習には，多くの時間と計算資源を要する．そのため，本研究では，収集が容易な政治ドメインのテキストデータである国会会議録と，比較的少ない計算資源でも学習がしやすい Transformer モデルである ELECTRA[5] を用いて，政治ドメインに特化した ELECTRA を構築し，政治ドメインに特化させることによる性能の変化を検証する．具体的には，Wikipedia と国会会議録を組み合わせたコーパスを構築し，そのコーパスを用いて ELECTRA の事前学習を行うことで，政治ドメインに特化した ELECTRA を構築する．また，Wikipedeia と CC-100[6][7] を組み合わせたコーパスを構築し，そのコーパスを用いた ELECTRA の事前学習を行うことで，検証時の比較対象となる汎用 ELECTRA も構築する．こうして構築した事前学習済み ELECTRA を用いて，NTCIR-16 QA Lab-PoliInfo-3 のサブタスクである Budget Argument Mining タスクによる比較検証を行う．Budget Argument Mining の詳細については，3 章の 1 節で述べる．図 1 に本研究の概要を示す．

本研究の貢献は，下記の 2 つである．

1. 政治ドメインに特化した ELECTRA の構築
2. 政治ドメインタスクによる汎用 ELECTRA と政治 ELECTRA の比較検証

2 章では ELECTRA の構築について述べ，3 章では政治ドメインのタスクによる検証について述べる．4 章ではまとめと今後の展望について述べる．

2　ELECTRA の構築

2.1　ELECTRA について

ELECTRA は，BERT の事前学習タスクである Masked Language Modeling (MLM) を改良することで，優れた学習

1）　北海道大学大学院情報科学院
2）　小樽商科大学
3）　北海道大学大学院情報科学研究院

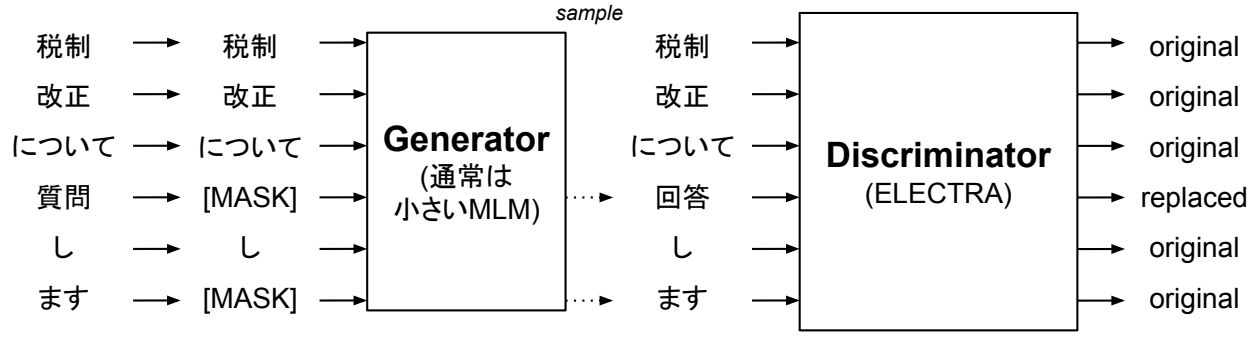

図 2　replaced token detection の概要

表 1　コーパスのデータサイズ

コーパス	データサイズ
Wikipedia	約 2.0 GB
Wikipedia + 国会会議録	約 3.5 GB
Wikipedia + CC-100	約 3.5 GB

効率と性能を得た Transformer モデルである．ELECTRA の事前学習タスクは，replaced token detection と呼ばれる．図 2 に，replaced token detection の概要を示す．replaced token detection では，Generator と Discriminator の 2 つのニューラルネットワークを学習させる．Genarator は，MLM を行うことで学習する．MLM は，入力文の一部 (通常 15%) のトークンを [MASK] トークンに置き換えることによりマスクし，そのマスクされた元のトークンを予測するタスクである．Discriminator は，入力文の各トークンが元の文と同じトークンであるか，Generator の予測で置き換えられたトークンであるかを区別することで学習する．モデルは GAN のような構造になっているが，敵対的に学習を行うのではなく，最尤法で学習を行う．事前学習後は，Generator を破棄し，Discriminator のみを ELECTRA として下流タスクに用いる．MLM では，マスクされたトークンに対してのみ学習をするため，学習に多くの計算時間を要する．対して，replaced token detection では，トークン全体に対して学習できるため，MLM と比較して少ない計算時間で学習することができる．また，同じモデルサイズ，データ，計算量の場合，MLM ベースで学習する Transformer モデルよりも，ELECTRA の方が優れた性能を示すことが報告されている [5].

2.2　事前学習に用いるコーパス

本研究では，ELECTRA の事前学習に用いるコーパスとして，3 種類のデータを用いた．

1 つ目は，Wikipedia のデータである．本研究では，Wikipedia に多く存在する記事ではないページ（曖昧さ回避，リダイレクトページ，削除済み，非エンティティ）のフィルタリングを施し，高品質に加工したデータセットである Wiki-40B[8] の日本語データセットを用いる．

2 つ目は，国会会議録のデータである．国会会議録の収集には，国会会議録検索システムが提供する検索用 API[1] を用いた．この API を用いて，2011 年 1 月 1 日から 2021 年 12 月 31 日の間に開かれた国会の会議録 10 年分を収集した．これらの会議録から，発言 (speech) 部分を抽出することでデータセットを構築した．

3 つ目は，CC-100[6][7] のデータである．CC-100 は，100 以上の単一言語のデータで構成される大規模なデータセットである．本研究では，日本語データセットから，国会会議録とおおよそ同じサイズのテキストを抽出し，データセットとして用いる．

以上の 3 つのデータセットを用いて，Wikipedia のみのコーパス，Wikipedia と国会会議録を組み合わせたコーパス，Wikipedia と CC-100 を組み合わせたコーパスの 3 種類を構築した．表 1 にそれぞれのコーパスのデータサイズを示す．

2.3　トークナイザーの学習

構築した 3 種類の事前学習コーパスを用いて，3 種類のトークナイザーの学習を行なった．トークナイザーの学習のパラメータは，鈴木らの金融ドメインにおける先行研究 [9] に倣って，東北大学の事前学習済み日本語 BERT モデル[2] を参考に設定した．形態素解析器には MeCab，辞書には IPAdic，サブワード分割のアルゴリズムには WordPiece を設定し，語彙数は 32,768 語とした．また，設定した語彙数のうち，[MASK] などの特殊トークンに 5 語，1 文字の単語に 6,129 語，ダウンストリームタスクにおけるファインチューニングの際に新たに学習される単語に 10 語を設定した．トークナイザーの学習には，東京大学の和泉研究室が公開する事前学習プログラム[3] を用いた．

2.4　ELECTRA の事前学習

ELECTRA の事前学習のハイパーパラメータは，Clark ら [5] が構築した ELECTRA-Small を参考に設定した．ELECTRA-Small を選択した理由は，ELECTRA-Base や ELECTRA-Large などの大きなモデルサイズと比較して，事前学習に必要な時間と計算資源が抑えられるためである．表 2 に設定したハイパーパラメータを示す．表 3 に事前学習の実行環境を示す．事前学習の完了までにかかった時間は，コーパスが Wikipedia のみの場合が約 2 日，Wikipedia と国会会議録を組み合わせた場合が約 2 日半，Wikipedia と CC-100 を組み合わせた場合も同様に約 2 日半であった．ELECTRA の事前学習には，トークナイザーと同じく，東京大学の和泉研究室が公開する事前学習プログラムを用いた．

1） https://kokkai.ndl.go.jp/api.html

2） https://github.com/cl-tohoku/bert-japanese

3） https://github.com/retarfi/language-pretraining

表 2　事前学習のハイパーパラメータ

ハイパーパラメータ	ELECTRA-Small
Number of layers	12
Hidden size	256
FFN inner hidden size	1,024
Attention heads	4
Embedding size	128
Generator size	1/4
Mask percent	15
Warmup steps	10,000
Learning rate	5e-4
Batch size	128
Train steps	1,000,000

表 3　事前学習の実行環境

CPU	Intel Core i9-10900
RAM	128GB
GPU	NVIDIA GeForce RTX 3090

3　検証

3.1　Budget Argument Mining について

検証に用いる政治ドメインのタスクには，Budget Argument Mining を用いた．Budget Argument Mining は，NTCIR-16 QA Lab-PoliInfo-3[4] の Shared Task の 1 つであり，地方議会と国会を対象として，予算項目と会議録に含まれる議論と結びつけ，その議論の役割を表すクラスを付与するというタスクである [3]．本タスクは，関連する予算項目の連結 (RID) と議論クラスの付与 (AC) の 2 つのサブタスクから構成されている．しかし，我々が本タスクに取り組んだ結果 [10] から，RID には Transformer モデルが向いていない傾向が示されている．従って，本研究では，本タスクの AC のみを用いて検証を行う．AC は，議論を 7 つのクラスに分類するタスクである．以下に，クラスの種類を示す．

1. Premise : 過去・決定事項
2. Premise : 未来（現在以降）・見積
3. Premise : その他（例示・訂正事項など）
4. Claim : 意見・提案・質問
5. Claim : その他
6. 金額表現ではない
7. その他

3.2　実験

実験は，各モデルをファインチューニングして行なった．まず，Budget Argument Mining のトレーニングデータセットから，合計 762 件の議論クラスの付与されたデータを抽出した．次に，このデータを用いて各モデルのファインチューニングを行い，最も良い結果を示したモデルを用いて，Budget Argument Mining のテストデータセットに対し議論クラスの予測を行なった．

表 4 に各モデルの正解率を示す．総合スコアでは，Wikipedia と国会会議録を組み合わせて事前学習を行なったモデルが，最も高いスコアを示した．会議ごとのスコアに注目すると，地方議会では，Wikipedia と国会会議録を組み合わせて事前学習を行なったモデルが，最も高いスコアを示し，国会では Wikipedia と CC-100 を組み合わせたコーパスで事前学習を行なったモデルが最も高いスコアを示した．また，Wikipedia のみのコーパスで学習を行なったモデルのスコアと比較すると，国会会議録を組み合わせて事前学習をしたにも関わらず，国会でスコアの低下が見られた．対して，CC-100 を組み合わせたコーパスで事前学習を行なったモデルは，地方議会と国会の両方でスコアの上昇が見られた．

4) https://poliinfo3.github.io/

表 4　各モデルの正解率

ELECTRA	地方議会	国会	総合
Wikipedia	0.3934	0.3846	0.3923
Wikipedia + 国会会議録	**0.4747**	0.3385	**0.4577**
Wikipedia + CC-100	0.4418	**0.4308**	0.4404

3.3　考察

実験結果のより詳細な分析を行うために，各モデルのクラスごとの正解数を算出した．表 5 に各モデルのクラスごとの正解数を示す．この表から考えられることは 2 つある．

1 つ目は，Wikipedia 以外のコーパスを追加することで汎化性能を得ているということである．Wikipedia のみで事前学習を行なったモデルは，正解したクラスの大部分を “Premise : 未来（現在以降）・見積” クラスが占めている．それと比較して，Wikipedia のみではなく国会会議録や CC-100 を組み合わせて事前学習を行ったモデルは，“Premise : 過去・決定事項” クラスや “Premise : その他（例示・訂正事項など）” クラスの正解数が大幅に増加している．

2 つ目は，コーパスによって分類の得意及び不得意なクラスが存在しているということである．Wikipedia と国会会議録を組み合わせて事前学習を行なったモデルは，“Premise : その他（例示・訂正事項など）” クラスの正解数が他のモデルと比べて多い．同様に，Wikipedia と CC-100 を組み合わせて事前学習を行なったモデルは，他のモデルが苦手な “Claim : 意見・提案・質問” クラスの分類が得意であることがわかる．このことによって，実験の結果にも説明がつく．表 6 に Gold standard データの国会と地方議会の割合を示す．国会のデータセットには全 65 件の議論クラスがあり，そのうちの 21 件が “Premise : その他（例示・訂正事項など）” クラス，4 件が “Claim : 意見・提案・質問” クラスであった．従って，Wikipedia と国会会議録を組み合わせて構築したモデルにおける国会の正解率が低かったのは，国会のデータセットには得意である “Premise : その他（例示・訂正事項など）” クラスが少ない上に，不得意である “Claim : 意見・提案・質問” クラスが多かったことが原因だと考えられる．

以上の 2 つから，国会会議録を用いて政治ドメインに特化した事前学習済みモデルの構築を行うことは，部分的に有効であったと考えられる．

4　まとめ

本研究では，収集が容易な政治ドメインのテキストデータである国会会議録と，比較的少ない計算資源でも学習がしやすい Transformer モデルである ELECTRA を

表 5　各モデルのクラスごとの正解数

Argument class	Gold standard	Wikipedia	Wikipedia + 国会会議録	Wikipedia + CC-100
Premise : 過去・決定事項	101	0	42	**47**
Premise : 未来（現在以降）・見積	196	**190**	147	161
Premise : その他（例示・訂正事項など）	145	14	**49**	16
Claim : 意見・提案・質問	42	0	0	**5**
Claim : その他	4	0	0	0
金額表現ではない	30	0	0	0
その他	2	0	0	0
合計	520	204	**238**	229

表 6　Gold standard データの国会と地方議会の割合

Argument class	地方議会	国会
Premise : 過去・決定事項	90	11
Premise : 未来（現在以降）・見積 / Estimates	171	25
Premise : その他（例示・訂正事項など）	141	4
Claim : 意見・提案・質問	21	21
Claim : その他	4	0
金額表現ではない	26	4
その他	2	0
合計	455	65

用いて，政治ドメインに特化した ELECTRA を構築し，政治ドメインに特化させることによる性能の変化を検証した．検証の結果，政治ドメインのコーパスによって正解率が上がったクラスが存在することから，政治ドメインに特化した Transformer モデルは部分的に有効であることが示された．

今後は，詳細なコーパス分析を行なった上で，地方議会会議録などを加えたさらに大きな政治ドメインのコーパスを作成し，大きなモデルサイズの事前学習済み Transformer モデルの構築を目指す．

謝辞

本研究は JSPS 科研費 21H03769 の助成を受けたものである．

参考文献

[1] Yasutomo Kimura, Hideyuki Shibuki, Hokuto Ototake, Yuzu Uchida, Keiichi Takamaru, Kotaro Sakamoto, Madoka Ishioroshi, Teruko Mitamura, Noriko Kando, Tatsunori Mori, Harumichi Yuasa, Satoshi Sekine, and Kentaro Inui. Overview of the ntcir-14 qa lab-poliinfo task. In *Proceedings of the 14th NTCIR Conference*, Tokyo, Japan, June 2019.

[2] Yasutomo Kimura, Hideyuki Shibuki, Hokuto Ototake, Yuzu Uchida, Keiichi Takamaru, Madoka Ishioroshi, Teruko Mitamura, Masaharu Yoshioka, Tomoyoshi Akiba, Yasuhiro Ogawa, Minoru Sasaki, Kenichi Yokote, Tatsunori Mori, Kenji Araki, Satoshi Sekine, and Noriko Kando. Overview of the ntcir-15 qa lab-poliinfo task. *Proceedings of The 15th NTCIR Conference*, 12 2020.

[3] Yasutomo Kimura, Hideyuki Shibuki, Hokuto Ototake, Yuzu Uchida, Keiichi Takamaru, Madoka Ishioroshi, Masaharu Yoshioka, Tomoyoshi Akiba, Yasuhiro Ogawa, Minoru Sasaki, Kenichi Yokote, Kazuma Kadowaki, Tatsunori Mori, Kenji Araki, Teruko Mitamura, and Satoshi Sekine. Overview of the ntcir-16 qa lab-poliinfo-3 task. *Proceedings of The 16th NTCIR Conference*, 6 2022.

[4] 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.

[5] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*, 2020.

[6] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451, 2020.

[7] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pp. 4003–4012, Marseille, France, May 2020. European Language Resources Association.

[8] Mandy Guo, Zihang Dai, Denny Vrandečić, and Rami Al-Rfou. Wiki-40B: Multilingual language model dataset. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pp. 2440–2452, Marseille, France, May 2020. European Language Resources Association.

[9] 鈴木雅弘, 坂地泰紀, 平野正徳, 和泉潔. 金融文書を用いた事前学習言語モデルの構築と検証. 人工知能学会第 27 回金融情報学研究会 (SIG-FIN), 2021.

[10] Keiyu Nagafuchi, Rin Sasaki, Seiya Oki, Yasutomo Kimura, and Kenji Araki. Ouc at the ntcir-16 qa lab-poliinfo-3 budget argument minig. *Proceedings of The 16th NTCIR Conference*, 6 2022.