Vol.2020-IFAT-139 No.4
2020/7/31

学術分野に特化した事前学習済み日本語言語モデルの構築

壹岐 太一[1,2,a)]　金沢 輝一[1]　相澤 彰子[1,2]

概要：言語モデルの事前学習は多くの自然言語処理タスクに対して有効な手法である．近年では，領域固有の学習済みモデルによって汎用的な言語モデルを補うことの重要性が示されている．本稿では学術情報データベース・サービスである CiNii の論文抄録を用いて BERT の事前学習を行い，文書分類タスクを通して効果を検証した．事前学習はスクラッチからの事前学習と既存の汎用モデルへの追加事前学習の 2 通りのシナリオで実験した．検証の結果，両者とも学術文献に関する文書分類タスクにおいて汎用モデルを上回る性能を達成し，特化した事前学習の有効性が示された．構築した言語モデルは公開を予定しており，学術文献に関する自然言語処理での活用が期待される．

Construction of a Pre-trained Language Model for Japanese Academic Documents

Abstract: Pre-training language models is an effective method for various natural language processing tasks. Recent work suggests the importance of domain-specific pre-trained models as a compensation for language models for generic domains. In this paper, we pre-trained BERT on paper abstracts from CiNii, an academic database service in Japan, and validated our method by fine-tuning for document classification. We experimented with two scenarios: pre-training from scratch and additional pre-training to an existing general-purpose language model. Both of them achieved higher performance than a general-purpose language model in document classification for academic publication and this shows the effectiveness of domain-specific pre-training. We plan to release our models to accelerate NLP research for the academic publication.

1. はじめに

近年，深層学習を用いた自然言語処理が成果を上げている．この手法の根本的な問題点は大規模なラベル付きのデータ資源が必要となる点である．クラウドソーシングの広まりによってラベル付データの収集は易化してきているものの，学術分野におけるラベル付けは専門家を必要とする場合が多く依然として難易度が高い．

ラベル付きデータの不足を解消する手法として，下流タスクに先んじて多量のラベルなし文で汎用的なテキスト表現を学習する，言語モデルの事前学習が注目されている．ULMFiT[1], ELMo[2], GPT-2[3], BERT[4], RoBERTa[5], ALBERT[6] を代表とするこの手法は，ラベル付け作業のコストが高い場合に特に有効であると期待される．実際に英語においては SciBERT（科学）[7]， BioBERT（生物医学）[8]，ClinicalBERT（カルテ）[9]，IMHO（インターネット上の意見）[10]，BERTweet（ソーシャルネットワークサービス; SNS）[11] といった特化型言語モデルが活用され始めている．日本語においてはレシピや SNS，ビジネスニュースのコーパスを用いた特化型言語モデルの試みはあるものの，我々の知る限り，日本語学術文献の大規模コーパスを用いた言語モデルの研究はない．

このような背景から，我々は学術文献に特化した日本語の学習済み言語モデルの構築を目指している．本稿では，第一歩として学術情報データベース・サービスである CiNii の論文抄録を用いて日本語言語モデルの事前学習を試みた．

既存の特化型言語モデルの研究を踏まえ，BERT モデルのスクラッチからの学習と汎用型言語モデルへの追加事前学習の 2 つのシナリオで言語モデルを構築し，学術/非学術文献（抄録）における性能を汎用型言語モデルと比較した．その結果，（1）両者とも学術文献に関して汎用型言語

[1] 国立情報学研究所
National Institute of Informatics , Chiyoda-ku, Tokyo 101–8430 Japan
[2] 総合研究大学院大学
Graduate University for Advanced Studies, SOKENDAI , Shonan Village, Hayama, Kanagawa 240–0193 Japan
[a)] iki@nii.ac.jp

モデルに対する性能の改善がみられた（2）スクラッチモデルは学術文献に関して最良の性能を達成し，追加事前学習モデルはある程度の汎用性を保ちつつ学術文献に関する性能を向上した．本研究で得られた学術文献に特化した言語モデルは公開を予定する．

2. 関連研究

2.1 BERT

近年の代表的な言語モデルである BERT（Bidirectional Encoder Representations from Transformers）[4] は大規模コーパスを用いて Transformer 構造 [12] を教師なし学習し，文脈を考慮した言語表現を効率的に獲得する．

学習時のロス計算には，文の一定数のトークンをマスクして入力し文脈から元のトークンを復元する masked language model（MLM）タスクと，2 文を入力しそれらが前後の関係になっているかを判定する next sentence prediction（NSP）が用いられる．ただし，NSP タスクはその有効性には議論があり，RoBERTa は NSP タスクを用いておらず，ALBERT は類似タスクで置き換えを提案している．そのため本稿においては MLM タスクを中心に議論する．

2.2 英語における特化型言語モデル

言語モデルを下流タスクに近いデータ（語彙や文法）で学習することはタスク性能の向上に繋がると考えられており，英語では特化型言語モデルの提案が進んでいる．特化型言語モデルの学習には新しいモデルをスクラッチで学習，汎用型言語モデルに追加事前学習，特化型言語モデルに追加事前学習など様々な手法が用いられる．表 1 に英語における BERT を用いた特化型言語モデルの概要を例示する．

2.3 タスクに関連するデータを用いる追加事前学習

タスクに関連するデータを用いる追加事前学習の有効性を複数の研究が報告している．Howard と Ruder は LSTM を用いた汎用的な言語モデル構造 ULMFiT を提案したが，その学習では（1）汎用言語モデル学習，（2）タスクドメイン言語モデル学習，（3）タスクファインチューニングの 3 段階で学習することが前提とされる [1]．Han らは BERT に関して，汎用的な分野でしか十分な教師データが用意できない場合において（1）タスク分野でのラベルなしデータを用いた追加事前学習，（2）汎用的な分野での教師あり学習の 2 段階学習を提案し，固有表現抽出タスクで有効性を確認した [13]．追加事前学習の有効性はクラス分類タスクにおいても示されている [14]．

タスク関連データを用いる事前学習の日本語に関する研究には，複合コーパスを用いる小川らの研究 [15] やモデル蒸留を行う新納らの研究 [16] がある．小川らは日本語版 Wikipedia から良質な記事を厳選した WikiText-JA コーパスを作成し，タスク関連コーパスと組み合わせることによって BERT 事前学習を効率化する手法を提案した．ゲーム実況とレシピの固有表現抽出において手法を検証し，日本語版 Wikipedia 全文を使う場合に比べて同等以上の性能が得られることを示した．新納らは，モデル蒸留によってパラメータ数を抑えた DistilBERT[17] を用いて，領域に特化した BERT モデルを効率的に構築する手法を提案した．蒸留したモデルは，広い領域で平均すると教師モデルの性能を超えることは難しいが，対象領域が十分に絞られていれば教師モデルを超えることが可能であり，モデル蒸留がコストと性能を両立する特化型モデル構築手法になり得ることを示した．

3. 本研究で用いるデータ

データの統計を表 2 に示す．以下，詳細を述べる．

3.1 言語モデル学習用データ

国立情報学研究所が運用する CiNii[*1] Articles は学協会刊行物・大学研究紀要・国立国会図書館の雑誌記事索引データベースなど，学術論文情報を検索の対象とする論文データベース・サービスであり，2020 年 3 月末現在では約 2,200 万件の論文が検索可能である．

本研究ではそれらの論文の一部である約 360 万件に関して日本語の抄録を抽出し，言語モデルの学習データとして用いる．抄録データは延べ約 6 億単語を含み，同じく抄録を学習対象とした BioBERT のデータの約 13%にあたる．

抄録データには，複雑な数式のマスク（単純な数式は残す），空白の調整，SGML 表記の復号といった正規化を施し，できる限り通常の文に近付けて用いる．

3.2 評価用データ

2 つのコーパスの分類タスクによって言語モデルを評価する．第一に学術文献での性能評価に，本研究で新たに作成した CiA25 コーパスを用いる．第二に学術分野への特化が汎用的な性能にどのような影響を与えるか確認するため，livedoor ニュースコーパスを用いる．

3.2.1 CiA25 コーパス

CiNii Articles の論文情報から CiA25 コーパスを作成した．このコーパスは 1970 年以降の抄録とその抄録が投稿された学会の情報を含む．25 の学会ラベルがあり，それぞれについて 2,500 件，合計で 62,500 件の抄録を収録する．学会の内訳は医学系（7 学会），化学系（4），情報学系（4），農学系（3），獣医学系（2），その他（5）である（学会は用意したデータ中で件数が多いもの上位 25 団体を採用した）．本稿ではこのコーパスを学会あたり 2,000 抄録，500 抄録の割合で学習，テストの 2 つのサブセットに分けて使用する．前処理として，40 文字未満の抄録は除外し，抄録に言

[*1] https://ci.nii.ac.jp/

表 1　英語における BERT を用いた特化型言語モデルの例.

	分野	ベースモデル	学習データ (大きさ)
BERT[4]	汎用	BERT (スクラッチ)	英語版 Wikipedia (25 億単語), BookCorpus (8 億単語)
SciBERT[7]	科学	BERT (スクラッチ)	Semantic Scholar[a] (32 億トークン)
BERTweet[11]	SNS コメント	BERT (スクラッチ)	Twitter の英語の tweet （16 億単語）
BioBERT v1.1[8]	生物医学	BERT	PubMed[b] (45 億単語)
Clinical BERT[9]	カルテ	BERT	MIMIC-III v1.4[c]
Clinical BioBERT[9]	カルテ	BioBERT	MIMIC-III v1.4[c]

[a]Semantic Scholar: 論文全体を使用．主な使用された分野は生物医学（82%），計算機科学（18%）.
[b]PubMed: 生物医学の論文抄録.　[c]MIMIC-III v1.4: 約 200 万カルテを含むデータセット.

表 2　本研究で用いるデータの統計．単語数は MeCab（IPA 辞書）で分かち書きして算出.

	事前学習データ	CiA25	ニュースコーパス
文書数	3,623,792	62,500	7,367
延単語数	618,700,839	13,699,081	4,815,111
最長文書長 (単語)	4,080	1,146	5,992
平均文書長 (単語)	170.7	219.2	653.6
平均単語長 (文字)	1.7	1.7	1.8
文書クラス数	-	25	9

語モデル学習用データと同様の正則化を施した．元データには稀に末尾に英語訳が付属している場合があるが，英語訳は可能な限り除去した．このコーパスと言語モデル学習用データの間に重複はない.

3.2.2 livedoor ニュースコーパス

livedoor ニュースコーパス[*2]（ニュースコーパスと略記）はインターネット上の 9 箇所のソースから収集した合計 7,367 記事を収録する日本語コーパスである．それぞれのソースの記事はほぼ同数含まれる．9 箇所のソースはニュース全般，スポーツ，IT 関連，家電，映画，女性向けコラム，モバイル関連，男性向け情報，女性向けニュースであり，学術分野とは異なる傾向の文書から構成される．本研究では平均記事数の 20%にあたる 164 記事を記事ソースごとに抽出してテストサブセットとして使用する.

4. 実験

4.1 言語モデルの事前学習に用いた学習環境

言語モデルの事前学習環境は 2 つの Intel(R) Xeon(R) Gold 6240 CPU @ 2.60GHz，約 80GB の RAM，4 枚の GPU（Tesla V100 32GB）を有するサーバーである．特に断りがない場合，4 枚全ての GPU を使用した.

4.2 言語モデルの設定

次に，本研究で用いた 4 種類の言語モデルを説明する.

4.2.1 東北大 BERT

一つ目のモデルは東北大学 乾・鈴木研究室日本語 BERT モデル[*3]である．このモデルは Wikipedia の記事によって学習した汎用型の言語モデルである．複数モデルが公開されているが，本研究では cl-tohoku/bert-base-japanese-whole-word-masking モデルを HuggingFace's Transformers[18] 経由で利用した．このモデルのトークナイゼーションは MeCab[19]（IPA 辞書と mecab-ipadic-NEologd[20] 辞書を併用）で入力文を分かち書きしたあと，WordPiece[21] で語彙をサブワードに分割する．語彙数は 32,000，トークンの最大入力数は 512 である．また，モデルパラメータ数の設定は標準的な BERT-base を用いている.

4.2.2 CiNiiBERT-APT

東北大 BERT を基に 3.1 小節に記した CiNii の抄録を用いて追加事前学習を行った言語モデルである．トークナイゼーションは，辞書の更新による齟齬を避けるため IPA 辞書のみで分かち書き[*4]，東北大 BERT の WordPiece 辞書によるサブワード分割を行った．東北大 BERT の重みを引き継いでいるため，語彙数は 32,000，トークンの最大入力数は 512 である．追加事前学習に使用したタスクは MLM タスクのみである．HuggingFace's Transformers の学習用スクリプトを使用し，ミニバッチサイズ 80 で 135k ステップ（学習データセット約 3 周分）学習した．学習率のスケジュール等その他の設定はデフォルトを使用した．追加事前学習に要した時間は約 60 時間であった.

4.2.3 CiNiiBERT-SCR

3.1 小節に記した CiNii の抄録のみを用いてスクラッチから学習した言語モデルである．トークナイゼーションには事前学習と同一のデータで学習した sentencepiece[22] を用いた．語彙数は 32,006 で他のモデルとほぼ同一だが，計算資源の都合上，トークンの最大入力数を 128 とした．同様の理由から隠れ層次元，フィードフォワード層の次元，アテンションヘッド数，レイヤー数についても BERT-base より小さな値を用いた（具体的な値は付録 A.1 を参照）．値は全体的に規模を縮小するよう設定し，結果的に総パラメータ数は約 4300 万となった．これは BERT-base のおおよそ 39%である．事前学習タスクは標準的な BERT と同様に MLM タスクと NSP タスクの両方を使用した.

[*2] https://www.rondhuit.com/download.html#ldcc
[*3] https://github.com/cl-tohoku/bert-japanese
[*4] 本研究では pip mecab-python3==0.996.5 を指定して同梱の辞書を用いる．2020 年 6 月 29 日の最新版 mecab-python3==1.0.0 には辞書が同梱されないため注意を要する.

BERT-multi-gpu リポジトリ[*5]の学習スクリプトを使用し，ミニバッチサイズ 256 で 100k ステップの学習を行った．事前学習に要した時間は約 80 時間であった．

4.2.4 CiNiiBERT-SCR-512

CiNiiBERT-SCR の最大入力数の小ささによる影響を緩和し比較の公平さを上げるため，CiNiiBERT-SCR のトークンの最大入力数を 512 に拡張したモデルである．入力数を増やすためには（最大入力長，埋め込み次元）の型を持つ position embeddings の第一次元の数を増やす必要がある．本研究では**事前学習後**に，第一次元方向をスプライン補間で 4 倍に拡大することによって最大 512 トークンの入力を可能にした[*6]．なお，重みに変更を加えるモデルであるためファインチューニングする場合にのみ用いた．

4.3 評価タスク

4.3.1 MLM ロス

事前学習完了後，学習の状態を確認するため，まず MLM ロスの計算を行った．文書のマスク生成は学習時と同一の処理を用い，15%の確率で候補トークンを選択し，マスク入力を作成した．元文は 2 つのコーパスのテストサブセット全サンプルを使用した．乱数シードを変更して 3 回の試行を行い，異なるマスクにおける平均値を算出した．

4.3.2 クラスタリングによる文章埋め込みの定性的評価

この評価は，ファインチューニングを行わず，学習済みモデルの文書埋め込みを直接評価することを目的とした．文書埋め込みの作成方法は複数考えられる．本研究では，ある文書の文書埋め込みを「全ての出力ベクトルのトークン数による平均」とした．各クラスから 100 サンプルを取り出し，言語モデルで文書埋め込みを計算後，T-SNE[23] を用いてその次元を 2 次元に削減した．T-SNE の実装は scikit-learn[24] の sklearn.manifold.TSNE[*7]を用いた．

4.3.3 クラス分類タスクへのファインチューニング

学術文献の CiA25 と非学術文献のニュースコーパスのクラス分類でファインチューニング時の性能を評価した．言語モデルの `[CLS]` トークンに対応する出力上にクラス数を出力次元とする線型変換を取り付け，モデル全体をクラスラベルに関する Cross Entropy ロスで最適化した．最適化器には ADAM を用い，ハイパーパラメータはミニバッチサイズを 6，最大エポック数を 5 で固定し，学習率のみ 8e-6，1e-5，2e-5，5e-5 の 4 通りで探索を行った．あるハイパーパラメータセットについて検証ロスが最も小さかったエポックの重みを採用し，ハイパーパラメータセット間で検証ロスが最小の重みをモデルの最終的な重みとした．各コーパスの学習サブセット全体からランダムに 20%を選んで学習から除外し，検証ロスの計算に用いた．

*5 https://github.com/guotong1988/BERT-multi-gpu.git
*6 補間には scipy.ndimage.zoom をデフォルト設定で用いた．
*7 https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html

表 3 2 つのコーパスにおける MLM ロス．

	CiA25	ニュースコーパス
東北大 BERT	3.32	2.56
CiNiiBERT-APT	**0.89**	**1.77**
CiNiiBERT-SCR (参考値)	2.60	3.65

表 4 CiNii 抄録出版社名分類タスク（CiA25）のテスト結果 適合率，再現率，F 値はマクロ平均．

	適合率	再現率	F 値	正解率
東北大 BERT	0.866	0.862	0.861	0.862
CiNiiBERT-APT	0.881	0.877	0.877	0.877
CiNiiBERT-SCR	0.887	0.886	0.885	0.886
CiNiiBERT-SCR-512	**0.889**	**0.887**	**0.887**	**0.887**

表 5 ニュースコーパス分類タスクのテスト結果．適合率，再現率，F 値はマクロ平均．

	適合率	再現率	F 値	正解率
東北大 BERT	**0.930**	**0.929**	**0.927**	**0.929**
CiNiiBERT-APT	0.928	0.927	0.926	0.927
CiNiiBERT-SCR	0.873	0.870	0.869	0.870
CiNiiBERT-SCR-512	0.903	0.900	0.900	0.900

4.4 結果

4.4.1 MLM ロス

表 3 に 2 つのコーパスに関するそれぞれの言語モデルの MLM ロスを示す．CiNiiBERT-SCR は東北大 BERT や CiNiiBERT-APT とトークナイゼーションが異なり，同一の問題を解いていないため参考値として掲載した．モデルごとに 2 つのコーパスの MLM ロスを比較すると，東北大 BERT では CiA25 よりニュースコーパスが小さく，CiNiiBERT-APT と CiNiiBERT-SCR では逆転している．これは抄録の学習により CiNiiBERT モデルの学術文献への特化が進んだことを示す．なお，CiNiiBERT-APT の MLM ロスはニュースコーパスにおいても東北大 BERT より改善がみられた．これは CiNiiBERT-APT を MLM ロスのみで学習したためであると考えられる．

4.4.2 クラスタリングによる文章埋め込みの定性的評価

図 1 は CiA25 テストサブセットの文書埋め込みの T-SNE によるクラスタリングを示す．類似領域の学会を同一色で色分けして示した．3 つの可視化結果に大きな差は見られず，汎用型の東北大 BERT においても類似領域の学会がクラスタを形成することが見て取れる．いずれのモデルもファインチューニングをせずとも，ある程度は領域の識別が可能な埋め込み表現を獲得できていることが確認できる．

4.4.3 クラス分類タスクへのファインチューニング

表 4 に CiA25 におけるクラス分類タスクのテスト結果を示す．スクラッチで学習し最大トークン入力数を 512 に拡張した CiNiiBERT-SCR-512 が F 値，正解率の点で最良だった．それより 1 ポイント程度下回るが追加事前学習

をした CiNiiBERT-APT も，汎用型モデルである東北大 BERT の F 値，正解率を上回った．この比較から構築した言語モデルは学術分野に関係する下流タスクにおいて有効であることがわかる．特に，CiNiiBERT-SCR/-512 はパラメータ数を少なく抑えた上で，学術分野に関する高い性能を達成している．

次に，表 5 はニュースコーパスにおけるクラス分類のテスト結果を示す．このテストでは東北大 BERT が CiNiiBERT を上回った．これは CiNiiBERT が学術文献に特化したためと考えられる．抄録だけで学習した CiNiiBERT-SCR は 512 文字入力の場合でも東北大 BERT に対して約 3 ポイント性能が低下している．一方，CiNiiBERT-APT は元のモデルである東北大 BERT とほぼ同等の性能を保っている．CiA25 に含まれない分野のデータも扱う状況では CiNiiBERT-APT が有効であると予想される．

CiNiiBERT-SCR と CiNiiBERT-SCR-512 の比較から，最大トークン入力長を増やすことは性能の向上に繋がることがわかる．特に，3 節の表 2 に示したように，サンプルテキストの平均長が CiNiiBERT-SCR の最大入力長[*8]である 128 を大きく上回るニュースコーパスにおいては約 3 ポイント F 値，正解率が改善し大きな効果を上げている．

5. おわりに

本稿では，CiNii Articles の抄録を用いて，スクラッチ・追加事前学習の 2 つのシナリオで日本語学術文献に特化した BERT モデルを構築した．文書埋め込みのクラスタリング，下流タスクへのファインチューニングで評価し，汎用型 BERT モデルに対する学術文献での有効性を示した．また，シナリオによってモデルの性質が異なるという知見を得た．スクラッチモデルは抄録のクラス分類で最高性能であり，追加事前学習に基づくモデルは抄録ではスクラッチモデルに劣るものの非学術文献でも汎用型モデルと同等の性能を維持するバランスの取れたモデルとなった．

今後は，本稿では未検証である他のソースの学術文献を用いた評価を進めると同時に，特化型言語モデルに最適な学習を検討することが課題である．また，文生成が可能な構造の使用を検討することで，翻訳や生成型要約といったタスクへと学習済み言語モデルの活用範囲を広げたい．

謝辞 本研究は，JST CREST-JPMJCR1513「構造理解に基づく大規模文献情報からの知識発見」の支援を受けたものである．

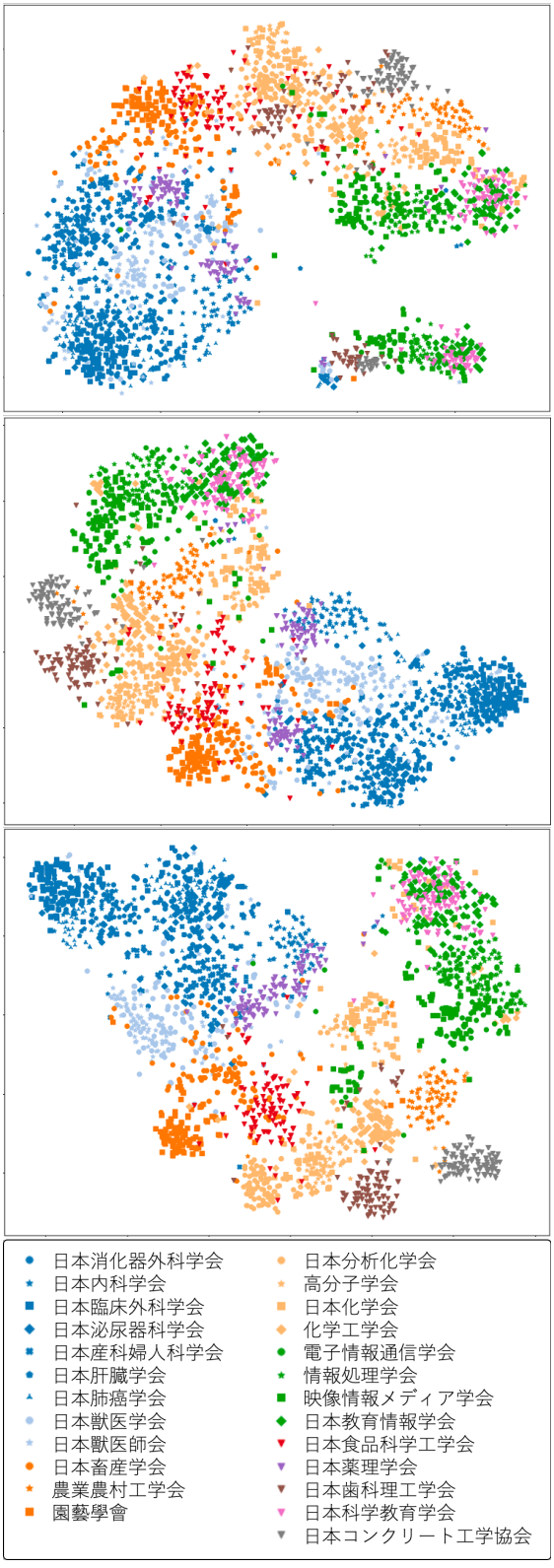

図 1 CiNii 抄録の T-SNE クラスタリング．上から東北大 BERT，CiNiiBERT-APT，CiNiiBERT-SCR．

参考文献

[1] Howard, J. and Ruder, S.: Universal Language Model Fine-tuning for Text Classification, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 328–339 (2018).

[2] Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. and Zettlemoyer, L.: Deep Contextualized Word Representations, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

[*8] 厳密には入力はトークン単位だが，差が大きいため最大入力長を超えるテキストが多いと推測される．

Technologies, Volume 1 (Long Papers), pp. 2227–2237 (2018).

[3] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I.: Language Models are Unsupervised Multitask Learners.

[4] Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, *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 (2019).

[5] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach, *arXiv preprint arXiv:1907.11692* (2019).

[6] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P. and Soricut, R.: ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, *International Conference on Learning Representations* (2019).

[7] Beltagy, I., Lo, K. and Cohan, A.: SciBERT: A Pretrained Language Model for Scientific Text, *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. 3606–3611 (2019).

[8] Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H. and Kang, J.: BioBERT: a pre-trained biomedical language representation model for biomedical text mining, *Bioinformatics*, Vol. 36, No. 4, pp. 1234–1240 (2020).

[9] Huang, K., Altosaar, J. and Ranganath, R.: Clinicalbert: Modeling clinical notes and predicting hospital readmission, *arXiv preprint arXiv:1904.05342* (2019).

[10] Chakrabarty, T., Hidey, C. and McKeown, K.: IMHO Fine-Tuning Improves Claim Detection, *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. 558–563 (2019).

[11] Nguyen, D. Q., Vu, T. and Nguyen, A. T.: BERTweet: A pre-trained language model for English Tweets, *arXiv preprint arXiv:2005.10200* (2020).

[12] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. and Polosukhin, I.: Attention is All you Need, *Advances in Neural Information Processing Systems*, pp. 5998–6008 (2017).

[13] Han, X. and Eisenstein, J.: Unsupervised Domain Adaptation of Contextualized Embeddings for Sequence Labeling, *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. 4229–4239 (2019).

[14] Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D. and Smith, N. A.: Don't Stop Pretraining: Adapt Language Models to Domains and Tasks, *arXiv preprint arXiv:2004.10964* (2020).

[15] 小川 晃，友利 涼，亀甲博貴，森 信介: WikiText-JA 構築による BERT 事前学習の効率化，言語処理学会 第 26 回年次大会 発表論文集，pp. 1173–1176 (2020).

[16] 新納浩幸， 白静， 曹鋭， 馬ブン: Fine-Tuning による領域に特化した DistilBERT モデルの構築，人工知能学会全国大会論文集 第 34 回全国大会 (2020)，一般社団法人 人工知能学会，pp. 1E3GS902–1E3GS902 (2020).

[17] Sanh, V., Debut, L., Chaumond, J. and Wolf, T.: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, *arXiv preprint arXiv:1910.01108* (2019).

[18] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M. and Brew, J.: HuggingFace's Transformers: State-of-the-art Natural Language Processing, *ArXiv*, Vol. abs/1910.03771 (2019).

[19] Kudo, T.: Mecab: Yet another part-of-speech and morphological analyzer, *http://mecab. sourceforge. jp* (2006).

[20] 佐藤敏紀，橋本泰一， 奥村学: 単語分かち書き辞書 mecab-ipadic-NEologd の実装と情報検索における効果的な使用方法の検討，言語処理学会第 23 回年次大会 (NLP2017)，言語処理学会，pp. NLP2017–B6–1 (2017).

[21] Sennrich, R., Haddow, B. and Birch, A.: Neural Machine Translation of Rare Words with Subword Units, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725 (2016).

[22] Kudo, T. and Richardson, J.: Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing, *arXiv preprint arXiv:1808.06226* (2018).

[23] Maaten, L. v. d. and Hinton, G.: Visualizing data using t-SNE, *Journal of machine learning research*, Vol. 9, No. Nov, pp. 2579–2605 (2008).

[24] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. et al.: Scikit-learn: Machine learning in Python, *the Journal of machine Learning research*, Vol. 12, pp. 2825–2830 (2011).

付　　録

A.1 BERT のスクラッチ学習時の設定

使用した bert_config は次のとおり．

```
{
    "attention_probs_dropout_prob": 0.1,
    "hidden_act": "gelu",
    "hidden_dropout_prob": 0.1,
    "hidden_size": 384,
    "initializer_range": 0.02,
    "intermediate_size": 2560,
    "max_position_embeddings": 128,
    "num_attention_heads": 6,
    "num_hidden_layers": 12,
    "type_vocab_size": 2,
    "vocab_size": 32006
}
```

また，学習データ作成処理では以下の設定を使用．

```
--do_lower_case=True
--do_whole_word_mask=True
--max_seq_length=128
--max_predictions_per_seq=5
--masked_lm_prob=0.15
--ramdom_seed=12345
--dupe_factor=3
```