ExDistilBERT：辞書拡張できるモデル蒸留によるドメインに特化した言語モデル

高　鵬挙[1]　山崎　智弘[1]　伊藤　雅弘[1]
[1]株式会社東芝　研究開発センター
知能化システム研究所　アナリティクスＡＩラボラトリー
{pengju1.gao,tomohiro2.yamasaki,masahiro20.ito}@toshiba.co.jp

概要

本論文は小規模コーパスでもドメインに特化できる ExDistilBERT を提案する．蒸留手法において，独自の手法で専門用語を辞書に追加し，一般用語と専門用語を独立して計算可能な損失関数を導入し，二種類の語彙や文脈知識を同時に学習できる．

蒸留手法で課題となっている過学習による知識忘却の問題を回避できることを実験により確認し，特殊ドメインにおいて，ExDistilBERT は教師モデルの性能を大幅に超えるだけでなく，軽量化もできた．

1　はじめに

近年，大規模コーパスを用いて学習された大規模言語モデルが注目を集めている．

BERT[1] を代表として，自然言語処理 (NLP) 領域での様々なタスクで，良い性能を達成している．汎化性能が高いと Fine-tuning の訓練コストが低くできる一方、BERT は Base サイズモデルでも一億を超えるパラメータを持つため，それを実装するためには膨大な計算リソースが必要である．様々な場面で応用するため，モデルの軽量化が要求される．さらに小規模コーパスで Fine-tuning する場合，事前学習した知識を忘れて，過学習しやすいことも課題になる．

金融，インフラ，法律，化学などの様々な分野では，一般文書にはない専門用語が大量に含まれている．一般文書を使って学習した言語モデルをそれらの分野に応用する場合，未知語として扱うことや，既知の短い単語の組み合わせに分割することが多い．日本語では，後者の場合が多く，長い専門用語がより小さい断片に分割されて，元の意味を保持できなくなることや，分割によって処理量が増えるなどの恐れがある．

モデル軽量化において，よく使われている手法には知識蒸留 (Knowledge Distillation)[2]，枝刈り (Pruning)，量子化 (Quantization) など [3] が存在するが，元モデルの性能を超えることができない．そこで我々は，モデル蒸留手法に辞書拡張機能を追加し，教師モデルの知識を学習すると同時に，スムーズに新たな単語を追加することができる ExDistilBERT を提案する．すなわちモデル軽量化と特殊ドメインへの適応性を両立できる．

結果として，ExDistilBERT は特殊ドメインの性能が教師モデルより向上することを示す．同時に，学習速度や推論時間も改善し，過学習問題も回避できることを示す．

2　関連研究

本節は本研究と関連性の強い事前学習言語モデル，モデル蒸留，辞書拡張の研究について説明する．BERT[1] は，Transformer[4] ユニットによって構成される．BERT は Masked Language Model (MLM) と Next Sentence Prediction(NSP) を使って，大規模コーパス用いた教師なしの事前学習法として提案され，NLP の様々なタスクにおいて小規模な教師データを用いた Fine-tuning によって，非常に良い性能を示した．

モデル軽量化においてよく使われる知識蒸留 (Knowledge Distillation)[2] では学生モデルが教師モデルの出力分布を近似するよう学習させる．NLP 領域では，蒸留手法を用いた軽量版 BERT の DistilBERT[5] がよく使われている．新納 [6] らは DistilBERT を用いて，ドメインに特化したモデルを構築したが，Fine-tuning しないと，教師モデルを超える性能を取得することが難しいことがわかった．

自然言語処理技術を特殊ドメインで活用するニーズの高まりから，様々な分野に特化した言語モ

デルの研究がある．英語において，生物ドメインのコーパスを使って，特化した BERT を訓練した BioBERT[7]，科学文章から専門用語を抽出し，科学ドメインに特化した SciBERT[8]，生物医学に関するデータベース PubMed を使って，生物医学ドメインに特化した PubMedBERT[9] などが存在する．日本語においても，医学分野 [10]，金融分野 [11]，法律分野 [12] などでドメインに特化した BERT が汎用日本語 BERT を超える性能を示した．ただし，ゼロから BERT を事前学習するには，大規模コーパスと計算リソースが必要となる．さらにより特化したドメインにに合致する，大規模コーパスを用意できない問題も存在する．

3 提案手法

本研究では，辞書拡張できるモデル蒸留の手法 ExDistilBERT を提案し，モデルを軽量化すると同時に，事前に大規模コーパスで訓練した汎用モデルで課題となっていた特殊ドメインの適応性を解決する．提案手法は，大規模汎用モデルを追加学習し，辞書拡張する従来手法と比べて，小規模コーパスを用いた条件で，過学習の恐れを低減することを確認する．

本論文は追加した単語を新単語と呼ぶ，教師モデル辞書に含んでいる単語を一般用語と呼ぶ．ExDistilBERT の概念図を図 1 に示す．新単語と一般用語を分けて，エンベディングや損失関数など処理を行うことで，新単語の知識と教師モデルの知識を同時に学習できる．

3.1 エンベディングと単語マスク

図 1 の token embedding において，拡張した辞書で，センテンスを分割し，単語をエンベディングする．

一般的にモデル蒸留は教師モデルと学生モデルの辞書を同じものにする必要があるが，辞書拡張のため，教師モデルに入力不可の新単語が存在する．図 1 の token masking において，教師モデルと学生モデルに対する，新単語と一般用語を分けて，特殊なマスク手段を設計した．

新単語のマスク

- 教師モデルの辞書には新単語が含んでいないため，全部マスクして，モデルに入力する．
- 学生モデルの辞書には新単語が含んでいるため，確率 $P(0 < P \leqq 1)$ でマスクされる

$P = 1$ の時，学生モデルの入力も新単語が全部マスクされるため，教師モデルの入力と同じになる．もし追加した単語の数が非常に多い場合，P で学生モデルの入力にマスクされた単語の比率を制御できる．新単語の範囲情報 $\mathbf{M}_{newtoken}$ はマスクベクターの形で損失関数の計算を利用する．

一般用語のマスク　新単語以外の部分において，BERT の MLM タスクと同じく，ランダムで単語のマスクと入れ替えを行う．入り替えの単語は教師モデルの辞書からピックアップしたものである．

3.2 損失関数計算

本節は新単語と一般用語を分けて，損失関数を計算する仕組みを説明する．

図 1 の右半分，二つの入力を教師モデル (t) と学生モデル (s) に入力し，最後レイヤの隠れ状態ベクター Hidden state，$H_{t,s}$ と単語予測の Logits，$L_{t,s}$ を取得する．教師モデルの知識を学習するため，一般用語の範囲 (selected position) だけに類似度などを計算する．範囲を設定は以下の式 1，2 で行っている．

$$H'_{t,s} = \{H_{t,s}(i) | \mathbf{M}_{newtoken}(i) = 0\} \tag{1}$$

$$L'_{t,s} = \{L_{t,s}(i) | \mathbf{M}_{newtoken}(i) = 0\} \tag{2}$$

以下の三つの尺度，Cosine 類似度，式 3；カルバック・ライブラー情報量，式 4；MSE 損失 (Mean squared error)，式 5，を用いて，損失関数を計算し，教師モデルの知識を学習させる．

$$\mathcal{L}_{Cosine}(H'_t, H'_s) = \frac{H'_t \cdot H'_s}{\|H'_s\| \|H'_t\|} \tag{3}$$

$$\mathcal{L}_{\mathrm{KL}}(L'_t, L'_s) = \sum_i L'_t(i) \log \frac{L'_t(i)}{L'_s(i)} \tag{4}$$

$$\mathcal{L}_{MSE}(L'_t, L'_s) = \frac{1}{n} \sum_{i=1}^{n} \left(L'_t(i) - L'_s(i)\right)^2 \tag{5}$$

次は，BERT と同じく，学生モデルの Logits　L_s とラベル L_{label} を用いて，マスクされた単語を推測する損失関数 $\mathcal{L}_{MLM}$ を計算する．

最終的な損失関数は，式 6 で以上の各損失の重み付き和を計算することで取得する．

$$\begin{aligned} \mathcal{L}_{final} = \alpha \mathcal{L}_{Cosine} + \beta \mathcal{L}_{\mathrm{KL}} + \gamma \mathcal{L}_{MSE} + \delta \mathcal{L}_{MLM}, \\ \text{where } \alpha, \beta, \gamma \geq 0, (\alpha, \beta, \gamma) \neq (0, 0, 0), \delta > 0 \end{aligned} \tag{6}$$

このような仕組みで，一般用語を学習する同時に，専門用語の知識を学習ことで，ドメインに特化した言語モデルを構築する．

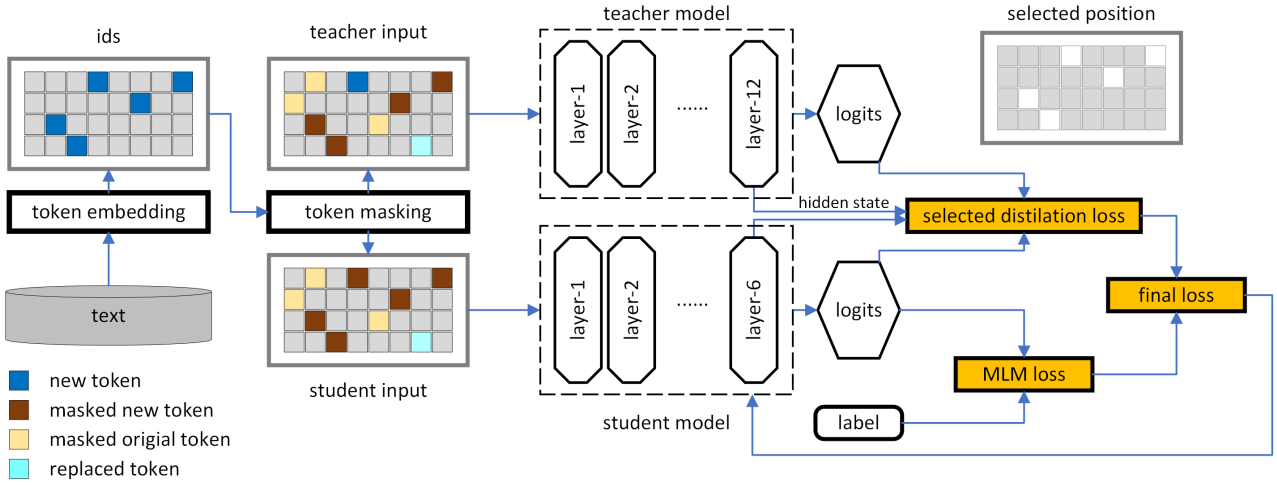

図 1　ExDistilBERT の概念図

表 1　新単語長さ分布と単語例 (新単語数 2000 の場合)

長さ	数	単語例
2	368	本件, 客様, 発送, 本日, 切替, 用紙, 客先, 取替, 取付, 異音, 送出, 新品
3	681	不具合, 不適合, 可能性, 再起動, 本事象, 再現性, 設定値, 一時的, 確認後
4	648	機体交換, 動作確認, 客様対応, 長期使用, 動作不良, 自動停止, 不具合品
5+	314	添付ファイル, 不具合現象, 作業管理者, 事象発生時, 点検・修理, 調査結果報告

表 2　データセットのセンテンス長さと規模

データセット	文書数	文数	イベント数	文平均長さ
$Data_A$	715	7941	8465	51.8
$Data_B$	7960	61038	46195	30.1

表 3　各モデルハイパーパラメータ設定

モデル	レイヤ数	Epoch 数	学習率	Batchsize
ExDistilBERT	6	2	5×10^{-5}	10
BERT_distillation	6	2	5×10^{-5}	10
BERT_addtoken	12	2	5×10^{-5}	10

4　評価実験

本節は本研究のデータ設定，実験設定と結果について説明する．

4.1　学習データと評価データ

モデル蒸留と辞書拡張コーパスは東芝社内のインフラドメインのテキスト文書データ (約 170 万文) を用いた．追加した単語の自動抽出は専門用語自動抽出システム [13][14] を使用した．教師モデルに存在せずかつ出現頻度高いの単語をピックアップした．新単語長さ分布と単語例は表 1 で示す．

イベント抽出データセットとして長文主体の文書 $Data_A$ と短文主体の文書 $Data_B$ の 2 種類を用意して，蒸留したモデルのエンベディング能力を評価する．イベント抽出タスクは，ドキュメント文書から，「配管に亀裂」や「水位が低下」のように原因や結果となりうる表現を抽出するタスクである．トラブル表現を正確に抽出するには，インフラドメインの文脈を理解することができる言語モデルを構築することが重要と考えられる．データセットのセンテンス長さと規模を表 2 で示す．抽出性能は完全一致の F 値と主要部一致 (イベント末尾 5 形態素のいずれかが重複する [15]) の F 値で評価している．

4.2　実験設定

本論文のモデル学習設定について説明する．本研究のベースラインモデルと ExDistilBERT の教師モデルは，東北大学 乾・鈴木研究室によって作成・公開された BERT モデル bert-base-japanese-v2[16] を使っている．追加した単語の数は 2000 個であり，HuggingFace's Transformers[17] の上に，PyTorch の実装を構築した．従来の手法と比較するため，ExDistilBERT と同じコーパスを用いて，BERT モデルから蒸留した BERT_distillation，同じ辞書で拡張した BERT_addtoken のモデルも学習した．各モデルのハイパーパラメータ設定は表 3 に示す．

訓練環境は RTX6000 一枚で行う．ExDistilBERT と BERT_distillation において，$\alpha = 3, \beta = 1, \gamma = 0.2, \delta = 1$ の設定をした．

4.3　評価結果

MLM タスク可視化，単語エンベディング能力，計算リソースにおいて，評価を行った．

表 4 MLM タスク出力の可視化 (推測された単語 Top5)

例文	本日、客先で [MASK] を実施しました	最短での [MASK] を教えてください.
BERT[16]	ツアー, キャンペーン, イベント, サービス, デモ	移動, 距離, 時間, 会話, 道
ExDistilBERT	動作確認, 再現試験, 確認試験, 試験, 再現テスト	納期, 対応状況, 日程, 対応方法, スケジュール

表 5 イベント抽出タスクにおいて，モデル評価結果

モデル	$Data_A$		$Data_B$	
	完全	主要部	完全	主要部
baseline				
BERT[16]	.619	.864	.629	.865
ours				
ExDistilBERT	**.671**	**.890**	.645	.866
comparison methods				
BERT_addtoken	.651	.861	**.661**	**.874**
BERT_distillation	.619	.851	.625	.862
other DistilBERTs				
DistilBERT_Laboro[18]	.599	.859	.619	.868
DistilBERT_Namco[19]	.528	.758	.580	.820

4.3.1 Masked Language Model(MLM) の可視化

ベースライン BERT モデルと比べて ExDistilBERT がドメインに特化した文脈理解能力を獲得しているかを確認するため，MLM タスクの出力を可視化した．[MASK] トークンの推測結果として得られた Top5 の単語を表 4 に示す．表 4 に示す通り，DistilBERT の方が動作確認や納期や試験といったインフラドメインでよく使われる専門用語を優先的に推測できた．

4.3.2 単語のエンベディング性能評価

本節では，単語エンベディングの性能評価を提案手法と既存手法で比較する．4.2 節説明したベースライン BERT モデル, ExDistilBERT, BERT_distillation と BERT_addtoken を含めて，既存の汎用蒸留モデルの DistilBERT_Laboro[18] と DistilBERT_Namco[19] も追加し評価を行った．評価した結果は表 5 で示す．

ベースライン BERT や単純なモデル蒸留手法 BERT_distillation と比べて，辞書拡張した ExDistilBERT と BERT_addtoken，$Data_A$ と $Data_B$ において性能向上した．専門用語の拡張効果が確認できた．$Data_A$ において ExDistilBERT は BERT_addtoken の性能を超えて，ベースライン BERT より完全一致の F 値 5 ポイント以上を向上した．BERT 丸ごとで学習して辞書拡張のは，パラメータ数が大規模である．相対的に小さい規模のコーパスで追加学習するとき，過学習して，元の知識を忘れてしまい，性能劣化の恐れがあることから，学習ハイパーパラメータの最適化という課題を残っている．$Data_B$ において，ExDistilBERT は BERT_addtoken を超えていない．短い文に対して，性能劣化の影響が少ないと推測された．BERT_distillation はできるだけベースラインの BERT に近づくのが学習目標となっており，ベースライン BERT と大体同じレベル性能を取得した．二つ公開された DistilBERT は，性能劣化の恐れがある．両者は今回のインフラドメインのコーパスではなく，一般的な日本語コーパスでモデル蒸留したもので，教師モデルからの知識勉強が不足の可能性が高いと推測された．

表 6 各モデルの学習時間, 推論時間とパラメータ数
(学習時間：2Epoch，推論時間：$Data_A$ の評価)

モデル	学習時間	推論時間	パラメータ数
ExDistilBERT	5.0 h	6.47 s	70.4×10^6
BERT_addtoken	21.4 h	8.39 s	111.2×10^6
BERT_distillation	5.8 h	4.26 s	70.4×10^6

4.3.3 計算リソース

4.2 節のモデル学習時間と評価実験において推論時間，パラメータ数を表 6 に示す．ExDistilBERT について，学習時間は従来の蒸留手法 BERT_addtoken と同じレベルであるが，BERT を辞書拡張 BERT_addtoken より 4 倍の学習速度になった．推論時間も BERT_addtoken より約 23%減少した．パラメータ数は BERT_addtoken より約 37%減少した．ExDistilBERT の方が計算リソースの観点からも優れていることが確認できた。

5 おわりに

本論文は小規模コーパスでもドメインに特化する ExDistilBERT を提案した．辞書拡張できるモデル蒸留の手法より，一般用語の知識はモデル蒸留によって教師モデルから継承すると同時に、専門用語に対して知識を学習できた．特殊ドメインの性能が教師モデルと比べて大幅に向上した．小規模コーパスで BERT を追加学習する場合，元の知識を失う過学習の問題も回避できて，BERT の追加学習より安定の学習結果を得ることができた．

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. **arXiv preprint arXiv:1810.04805**, 2018.

[2] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. **arXiv preprint arXiv:1503.02531**, Vol. 2, No. 7, 2015.

[3] 康平山本, 素子橘, 蔵人前野. ディープラーニングのモデル軽量化技術. Oki テクニカルレビュー, Vol. 86, No. 1, pp. 24–27, 05 2019.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. **Advances in neural information processing systems**, Vol. 30, , 2017.

[5] Victor Sanh, L Debut, J Chaumond, and T Wolf. Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter. arxiv 2019. **arXiv preprint arXiv:1910.01108**, 2019.

[6] 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 (2020), pp. 1E3GS902–1E3GS902. 一般社団法人 人工知能学会, 2020.

[7] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. **Bioinformatics**, Vol. 36, No. 4, pp. 1234–1240, 2020.

[8] Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. **arXiv preprint arXiv:1903.10676**, 2019.

[9] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. **ACM Transactions on Computing for Healthcare (HEALTH)**, Vol. 3, No. 1, pp. 1–23, 2021.

[10] Yoshimasa Kawazoe, Daisaku Shibata, Emiko Shinohara, Eiji Aramaki, and Kazuhiko Ohe. A clinical specific bert developed using a huge japanese clinical text corpus. **Plos one**, Vol. 16, No. 11, p. e0259763, 2021.

[11] 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. 人工知能学会第二種研究会資料, Vol. 2021, No. FIN-027, p. 05, 2021.

[12] 宮崎桂輔, 菅原祐太, 山田寛章, 徳永健伸. 日本語法律分野文書に特化した bert の構築. 第 28 回年次大会発表論文集 (2022 年 3 月), 2022.

[13] 専門用語自動抽出システム (配布). http://www.forest.eis.ynu.ac.jp/Forest/ja/term-extraction.html.

[14] 中川裕志, 湯本紘彰, 森辰則. 出現頻度と連接頻度に基づく専門用語抽出. 自然言語処理, Vol. 10, No. 1, pp. 27–45, 2003.

[15] 伊藤雅弘, 山崎智弘. アノテーション漏れ推定を用いたエンティティ抽出. 第 27 回年次大会 発表論文集 (2021 年 3 月), 2021.

[16] Tohoku NLP Group. Pretrained japanese bert models. https://github.com/cl-tohoku/bert-japanese.

[17] Hugging Face team. State-of-the-art machine learning for pytorch, tensorflow, and jax. https://github.com/huggingface/transformers.

[18] 株式会社 Laboro.AI. Laboro distilbert. https://laboro.ai/activity/column/engineer/laboro-distilbert.

[19] バンダイナムコ研究所技術開発本部. Wikipedia 日本語版全文を学習した distilbert モデル. https://github.com/BandaiNamcoResearchInc/DistilBERT-base-jp.