情報圧縮を用いた訓練データの重複削減

堤田 恭太　村瀬 文彦　三谷 陽
株式会社デンソー
{kyota.tsutsumida.j7m, fumihiko.murase.j6b, akira.mitani.j5g}@jp.denso.com

概要

公開されている汎用的なニューラル言語モデルを利用する際，当該タスクやドメインの文書を用いた継続事前学習を行って，下流タスクの精度向上を図ることがある．また，訓練データから重複を除去することで，学習効率が向上することが知られている．そこで本研究では，情報圧縮を用いて訓練データ中の重複を削減する手法を提案する．実データを用いた類似文書検索タスクにて，完全一致を除去するナイーブな手法や，ランダムなデータ選択などと比べて，提案法がより高い検索精度を実現する言語モデルを構築できることを示した．

1　はじめに

BERT モデル [1] の登場以降，大規模な事前学習を行った汎用的な公開の BERT 型モデルが，文書分類や文書検索など幅広いタスクに活用されている．特定の分野や企業で利用する際は，下流タスクでの精度向上を目的として，当該ドメインの文書を用いた継続事前学習 [2] を行うことがあり，医療系分野の BioBERT[3] や，科学技術分野の SciBERT[4] などが知られている．日本語のモデルとしては，Wikipedia コーパス[1] の日本語記事を事前学習した東北大 BERT モデル[2] などが広く利用されている．

また近年の研究で，訓練データから重複を除去することにより，過学習のリスクを低減して学習時の安定性を高め，同一のデータに触れる回数を減らすことで学習効率が向上するという報告がある [5]．重複の削減に限らず，訓練データをより簡潔に保つことで，学習効率の向上に加え，データ管理のコスト低減も期待できるとされる [6]．

そこで本研究では，汎用的な公開モデルである東北大 BERT モデルに対して，社内文書など特定ドメインの文書を継続事前学習する際に用いる，訓練データの重複削減手法を提案する．より具体的には，暫定的な訓練データの集合に対し，追加する候補の文書が付加する情報量を指標化し，閾値以上に情報量が増加する文書を加え，閾値未満の文書を除去する逐次的な処理によって，訓練データ中の重複するデータを削減する．これにより，文書内の表記揺れや語順の違いなどを吸収しつつ，多様な専門用語を含む文書を残しながら，重複する訓練データを除去することができる．

本稿の構成は以下の通りである．第 2 章では関連研究として，訓練データの重複削減と，情報圧縮を用いた自然言語処理の先行研究について述べる．第 3 章では，提案法について説明する．第 4 章では，実験について述べる．製造機器の不具合について原因や対処を記録した実データ，および，類似した過去の不具合事例の文書を検索する類似文書検索タスク，比較手法，結果について述べる．最後に，第 5 章でまとめと今後の課題について述べる．

2　関連研究

2.1　訓練データの重複削減

第 1 章でも述べた通り，言語モデルの学習においては，訓練データの重複はモデルの性能と学習効率に重大な影響を与えることが知られており，重複を削減する技術は **deduplication** と呼ばれている [5]．重複データの存在は，モデルが既に学習したデータに繰り返し接触することで学習効率が低下し，特定のパターンに過剰に適合（overfitting）するリスクを高めるため，汎化性能を損なうことも指摘されている．Albalak らの報告 [7] によれば，訓練データに含まれる単語の頻度などの統計的な性質より，データの多様性を重視する操作であると整理されている．また同報告によれば，訓練データの選択・削減には，重複の削除だけでなく，日本語や英語など学習させたい言語を選ぶフィルタリング，学習に適さないもの (決済画面や広告，不自然な文など) を除去す

1） https://huggingface.co/datasets/legacy-datasets/wikipedia
2） https://github.com/cl-tohoku/bert-japanese

るヒューリスティクス，有害文書の除去など様々な手法が挙げられている．社内文書を訓練データとして用いる場合，特に文書が作業記録としての性質を持つものでは，ニュース記事等と異なり，同一の内容も多く含まれているため，重複の削除は重要な前処理と考えられる．

既存の重複削減の手法としては，URL や，本文で一定以上の文字列の一致を Suffix Array を用いて検出する手法 [5]，段落や文字列の n-gram レベルでハッシュ値を算出して他の文書との類似度を測る手法 [8] などがある．これらの手法は主に，LLM の学習に用いる大規模な Web コーパスなどへの適用を想定し，同一の記事や，他の記事の引用部分を章や節レベルで特定することで重複除去を行っている．このように文字列の一致を基準とすると，語順の変化や表記揺れにより検出力が低下すると考えられる．そこで，ハッシュ関数を独自に学習したり [9]，複数の小規模なモデルの学習を行ってその尤度差を基準とする手法 [6] が考えられているが，そのモデルの学習およびチューニングが必要となる．

一方，本研究で取り組む社内文書の類似文書検索タスクは，製造機器の不具合やその対処について自由記述された作業記録の文書を対象としている．定型文的なフレーズも多く含まれるが，文書が数十から百文字程度と比較的短く，語順の違いや表記揺れ等が多い．そのため，他の文書との一致箇所を特定するより，小さな差を吸収しつつ，珍しい不具合事例に現れる専門用語も訓練データに積極的に残す目的で，提案法は各文書によって増加する情報量を基準とする方法とした．

2.2 情報圧縮距離 (Normalized Compression Distance, NCD)

情報圧縮距離（Normalized Compression Distance, 以下 NCD）

[10] は，Kolmogorov 複雑性理論に基づき，データ間 (x, y) の情報の共通性を類似度とする尺度である．NCD は次式で定義される．

$$\mathrm{NCD}(x,y) = \frac{C(xy) - \min\big(C(x), C(y)\big)}{\max\big(C(x), C(y)\big)}. \qquad (1)$$

ここで，$C(x)$ はデータ x を gzip などの汎用的な圧縮手法で圧縮した際のデータサイズを表し，$C(x)$ は，x を記述するために必要な最小の情報量である Kolmogorov 複雑性 $K(x)$ の実用的な近似として扱われる．また，$\min\big(C(x), C(y)\big)$ は，$C(x)$ と $C(y)$ のうち最小のもの，$\max\big(C(x), C(y)\big)$ は最大のものをそれぞれ表す．同じ文字列を含む文書間 (x, y) では，それらを結合した文書 xy の圧縮時のデータサイズ $C(xy)$ がより小さくなるため，この度合いを類似度とすることができる．Jiang ら [11] は，分類したい文書と，クラスラベル付きの訓練データ中の文書との NCD が最小となる (最も類似した) クラスへ分類する手法を提案し，分類器のための特徴抽出やモデルの学習**なし**に，高い精度で文書分類が実現できることを示した．

しかしながら，訓練データの重複削減に NCD を用いた研究は我々の調査する限り報告されていない．そこで本研究では，特徴抽出や学習なしに訓練データの重複削減を行う手法として，NCD を改良した手法を提案する．

3 提案法

提案法の基本的なアイデアは，通常 2 つの文書間の類似度を測るために用いる NCD を，BERT などのニューラル言語モデルの継続事前学習に用いる訓練データ集合 $\mathcal{T}$ と，訓練データの候補である文書の集合 $\mathcal{C}$ から取り出した i 番目の候補文書 $c_i \in \mathcal{C}$ について算出し，その類似度を測ることである．これにより，既存の訓練データ集合に含まれていない専門用語を含む候補文書では値が大きく，逆に訓練データ集合に既に存在する場合は値小さくなるため，訓練データ集合に加えるべき候補文書を選択的に残すことができ，逆に重複を特定，除去することでデータ削減を行うことができる．この手法は，訓練データ集合と候補を逐次的に比較し，閾値以上となる候補を訓練データ集合へ追加することで最終的な訓練データ集合を構築するアルゴリズムとなっている．詳細な手順は，次節および Algorithm1 に示す．例えば，訓練データ集合 $\mathcal{T}$ が，

- 「使用劣化 寿命 コンベアベルト切れ」
- 「センサー故障 LS 不良」[3)]

とある場合，訓練データの候補集合 $\mathcal{C}$ から取り出した候補 c についてそれぞれ，逐次的に訓練データ集合 $\mathcal{T}$ と比較し，

- c_1「コネクタ断線 吸着せず」 → $\mathcal{T}$ へ追加
- c_2「センサー故障 LS 不良」 → 重複のため除去
- c_3「コネクタ断線 吸着せず」 → 既に追加された c_1 と重複のため除去

のように訓練データ集合への追加・除去が進む．

3) LS は，リミットスイッチの略記

3.1 提案法の詳細

提案法では，訓練データ集合 $\mathcal{T}$ と，その候補文書の集合 $\mathcal{C}$ から取り出した候補文書 c_i（以下，単に「候補」）との情報の重複度合いを NCD の枠組みで算出し，重複が少ない (未知の単語や専門用語を多く含む) と判定された候補を訓練データ集合に加える．具体的には，候補 c_i を，訓練データ集合 $\mathcal{T}$ に追加した場合に増える情報量 s_i を $\mathrm{Score}(\mathcal{T}, \mathrm{c_i})$ として次式で定義する．

$$s_i = \mathrm{Score}(\mathcal{T}, c_i) = \frac{C(\mathcal{T}c_i) - \max\big(C(\mathcal{T}),\, C(c_i)\big)}{\min\big(C(\mathcal{T}),\, C(c_i)\big)}. \quad (2)$$

ここで，NCD と同様に，$C(x)$ は文書 x を gzip などで圧縮した際のデータサイズとする．また，$\mathcal{T}c_i$ は，現在の訓練データ集合 $\mathcal{T}$ に候補 c_i を追加した集合，$C(\mathcal{T}c_i)$ は圧縮時のデータサイズを表す．このとき，訓練データ集合 $\mathcal{T}$ に含まれる文書と候補 c_i が類似していると，集合 (結合した文書) $\mathcal{T}c_i$ の圧縮時のデータサイズ $C(\mathcal{T}c_i)$ の増分が小さいため，s_i が小さくなる．逆に類似していない場合，s_i は大きくなる．ここで，訓練データ集合 $\mathcal{T}$ は徐々に大きくなるため，$C(\mathcal{T}) > C(c_i)$ が成り立ち，

$$s_i = \mathrm{Score}(\mathcal{T}, c_i) = \frac{C(\mathcal{T}c_i) - C(\mathcal{T})}{C(c_i)}, \quad (3)$$

となる．この Score は，候補 c_i のもつ情報が，現在の訓練データ集合 $\mathcal{T}$ によってカバーされて**いない**分の情報量の，元々の文書が持つ情報量に対する割合を測ったものと解釈できる．割合として解釈することで，s_i は文書長に依らずに 1 つの閾値 θ を用いて取捨選択できるようになる (Alg.1, 4 行目)．これを候補文書の集合 $\mathcal{C}$ 全てに適用するか，最終的な訓練データ集合 $\mathcal{T}$ が必要件数 K に達するまで繰り返し，重複を削減した訓練データ集合を構築する．

Algorithm 1 提案法における訓練データ集合の構築

Require:
訓練データ集合 $\mathcal{T}$（初期状態）
候補文書の集合 $\mathcal{C} = \{c_1, c_2, \dots\}$
閾値 θ（重複判定用）
必要件数 K（学習に用いるデータの目標件数）

Ensure:
学習に用いる訓練データ集合 $\mathcal{T}$

1: $N_t \leftarrow |\mathcal{T}|$ ▷ 現在の訓練データ数をカウント
2: **for all** 候補文書 $c_i \in \mathcal{C}$ **do**
3: $\mathcal{T}c_i \leftarrow \mathcal{T} \cup \{c_i\}$ を結合した文書とし次式を計算：

$$s_i = \mathrm{Score}(\mathcal{T}, c_i) = \frac{C(\mathcal{T}c_i) - \max\big(C(\mathcal{T}),\, C(c_i)\big)}{\min\big(C(\mathcal{T}),\, C(c_i)\big)}$$

4: **if** $s_i \geq \theta$ **then** ▷ 情報の重複が少ない
5: $\mathcal{T} \leftarrow \mathcal{T} \cup \{c_i\}$ ▷ 候補 c_i を追加
6: $N_t \leftarrow N_t + 1$
7: **if** $N_t \geq K$ **then**
8: **break** ▷ 必要件数で終了
9: **end if**
10: **else** ▷ $s_i < \theta$ の場合は削除
11: **skip** c_i
12: **end if**
13: **end for**
14: **return** $\mathcal{T}$ ▷ 最終的な訓練データ集合を返す

4 実験

本章では，実験用の実データ，訓練データの削減の比較手法，類似文書検索タスクと評価方法について述べ，最後に実験結果を述べる．

4.1 実験データのラベルとタスク

製造機器の不具合やその対処について記録した弊社実データである保全履歴データから，不具合の現象および原因についての自由記述を用いる．記述は数十から百文字程度と比較的短く，語順の違いや表記揺れ等が多く含まれている．クエリとなる文書を用いて，過去の不具合事例を検索する類似文書検索タスク用途で人手のラベル付けが行われており，クエリとなる正例 10 件，検索のターゲットとなる正例 218 件と負例 777 件 (ランダム選択時の正解率約 22%)，ラベルなしデータ約 24 万件のデータセットとなっている．実験では，このラベルなしデータを用いてモデルを継続事前学習する際の，学習を効率化するデータ削減手法を検索タスクにて比較評価した．また，自由記述に対する前処理として，適さないデータ (極端に短い定型文，「定期検査」等) や不要な空白文字等を除去し，NFKC での文字列の正規化処理を行った．

4.2 比較手法と提案法の設定

前述の通り，ラベルなしデータ約 24 万件を各手法で 1 万件に削減し，削減手法の比較評価を行った．BERT モデルの継続事前学習のコストは，およそデータ件数に比例して増加するため，1 万件を

10epoch 学習しても，総学習時間は全量を用いた場合の半分以下に抑えられる．比較手法および提案法の設定は次の通りである．

1. **BERT**: 東北大 BERT の base モデル v2 [4] を，継続事前学習していない状態で用いた．
2. **random**: ランダムに選択して用いる手法．関連研究で述べた通り，重複削減は元のデータセットの統計的な性質より多様化を好む戦略であるが [7]，ランダムでは不具合の頻度などの統計的な性質を活用できる．
3. **uniq**: 完全一致するデータを除去するナイーブな重複削減の手法．先行研究 [5] でも同一記事の特定が行われていたが，多発する事例は自由記述でも完全一致しており，それらを削除してからランダムに選択した．
4. **prop.**: 第 3 章で述べた提案法．uniq の完全一致に加えて，表記揺れや語順の変化を吸収しながら重複を除去できる．K は 1 万，閾値 θ は 1 万件程度が残る 0.4，圧縮アルゴリズムは Jiang ら [11] と同様に gzip を用いたが，コード表の最適化がデータ追加により進んだ場合などで，ごくまれに Score$(\mathcal{T}, c)$ が負値となるため，負値の場合は閾値以下でも例外的に訓練データに追加することとした．提案法の計算時間は，一般的なデスクトップ PC でも約 24 万件を 20–25 分程度で処理でき，モデルの学習時間 (数時間から 1 日程度) と比べて十分小さい．

4.3 継続事前学習と類似文書検索タスク評価

各手法で削減した訓練データは，RoBERTa[12] での継続事前学習用のデータとして用いてモデルを構築，前述のタスクでの検索精度の比較を行った．学習のハイパーパラメータは東北大 BERT の設定や，他の研究事例を参考に経験的に決定して共通とし，最適化器は AdamW[13] を用い，$\beta_1 = 0.9$, $\beta_2 = 0.999$, 学習率 1.0e-5，訓練データ 1 万件の 10epoch でステップ数 10 万で行っている． 検索タスクの評価は，まず文書の BERT の各トークンの出力を平均化する，mean pooling を用いて文書ベクトル化し，faiss[14] を用いたデータベースにターゲット文書のベクトルを格納，クエリ文書も同様にベクトル化してデータベースをベクトルのコサイン類似度を基準として最近傍探索し，より上位 k に正例が含まれる場合に高い検索精度となる，平均適合率 (map@k)[15] を用いてモデルの良さを評価した．

4） https://huggingface.co/tohoku-nlp/bert-base-japanese-v2

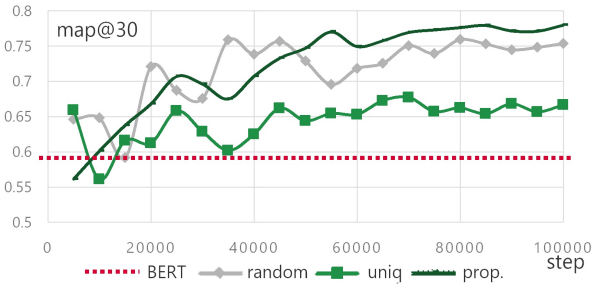

図 1 各手法の 5,000step 毎のタスク精度 (map@30) の推移

表 1 類似文書検索タスクでの評価結果

map	@1	@10	@30	@50
BERT	0.80	0.71	0.59	0.57
random	0.90	0.82	0.75	0.71
uniq	0.80	0.71	0.66	0.63
prop.	**1.00**	**0.83**	**0.78**	**0.73**

4.4 実験結果と考察

表 1 に，各手法の map@k での類似文書検索タスクの精度を示す．最も高い値を太字で示したが，提案法によるデータ削減を行ったモデルが最も良い精度となった．また，継続事前学習を行った各手法が，学習を行っていない BERT より高い精度となっている．また，図 1 は，各手法の 5,000 step 毎のタスク精度 (map@30) の推移である．前半は random が優れるが，学習の後半には提案法の prop. が優れる結果となっている．学習中，提案法および uniq は，random より学習時のパラメータの更新量が小さくなる傾向が見られており，これはデータが多様化したことで，最適化器 AdamW のモーメンタム項による加速が抑えられ，より慎重に学習が進められたためと考えている．

5 まとめ

本研究では，情報圧縮を用いた訓練データの削減手法を提案した．提案法は，訓練データ集合に候補の文書を追加した場合に増加する情報量を基準としてデータを逐次的に追加・削減することで，語順の変化や表記揺れを吸収しつつつ，学習なしに類似文書検索タスクの精度を向上する訓練データを構築できる．実データを用いた実験にて，ナイーブな比較手法と比べ，高い検索精度を実現する言語モデルを構築できることを示した．

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. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, **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.

[2] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 8342–8360, Online, July 2020. Association for Computational Linguistics.

[3] 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.

[4] Iz Beltagy, Kyle Lo, and Arman Cohan. SciBERT: A pretrained language model for scientific text. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, **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. 3615–3620, Hong Kong, China, November 2019. Association for Computational Linguistics.

[5] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, **Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 8424–8445, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[6] Jun Suzuki, Heiga Zen, and Hideto Kazawa. Extracting representative subset from extensive text data for training pre-trained language models. **Information Processing & Management**, Vol. 60, No. 3, p. 103249, 2023.

[7] Alon Albalak, Yanai Elazar, Sang Michael Xie, Shayne Longpre, Nathan Lambert, Xinyi Wang, Niklas Muennighoff, Bairu Hou, Liangming Pan, Haewon Jeong, Colin Raffel, Shiyu Chang, Tatsunori Hashimoto, and William Yang Wang. A survey on data selection for language models. **Transactions on Machine Learning Research**, 2024. Survey Certification.

[8] Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, H. Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant M. Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew J. Johnson, Blake A. Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Edward Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. Scaling language models: Methods, analysis & insights from training gopher. **CoRR**, Vol. abs/2112.11446, , 2021.

[9] A.Z. Broder. On the resemblance and containment of documents. In **Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No.97TB100171)**, pp. 21–29, 1997.

[10] Ming Li, Xin Chen, Xin Li, Bin Ma, and P.M.B. Vitanyi. The similarity metric. **IEEE Transactions on Information Theory**, Vol. 50, No. 12, pp. 3250–3264, 2004.

[11] Zhiying Jiang, Matthew Yang, Mikhail Tsirlin, Raphael Tang, Yiqin Dai, and Jimmy Lin. "low-resource" text classification: A parameter-free classification method with compressors. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, **Findings of the Association for Computational Linguistics: ACL 2023**, pp. 6810–6828, Toronto, Canada, July 2023. Association for Computational Linguistics.

[12] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. **CoRR**, Vol. abs/1907.11692, , 2019.

[13] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In **International Conference on Learning Representations**, 2019.

[14] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. 2024.

[15] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. **Introduction to information retrieval**, Vol. 39. Cambridge University Press Cambridge, 2008.