事前学習済みモデルに基づく検索モデルにおけるドメイン適応手法の比較と相乗効果の検証

飯田大貴 [1,2]　岡崎直観 [1]
[1] 東京工業大学　[2] 株式会社レトリバ
{hiroki.iida@nlp.c., okazaki@c.}titech.ac.jp

概要

BERT 等の事前学習済みモデルを用いた検索モデルは，教師データのドメイン外で使用すると検索精度が低下することが知られている．特に，学習データと対象データの語彙が大きく異なる場合は，大幅な精度低下につながる．そのため，検索モデルのドメイン外での検索精度を向上させる手法が複数提案されている．本論文では，統一された実験設定において各手法の性能やその組み合わせによる相乗効果の有無を検証した．その結果，事前学習済みモデルのドメイン適応，BM25 など語彙の一致に基づく検索モデルを併用すること，及びその組み合わせが複数の事前学習済みモデルを用いた検索モデルで有効であることが明らかになった．さらに BM25 のみならず複数の検索モデルを併用することで，現時点での世界最高性能を上回る結果を得た．

1　はじめに

BERT [1] を始めとする事前学習済みモデルを用いて，教師データとなる検索データセットで検索モデルを訓練することにより，BM25 [2] などの語彙の一致に基づく検索モデルを大きく上回る検索精度を達成できることが明らかになった [3]．代表的な検索モデルとしては，クエリと文書を密ベクトルに変換し，その内積を関連度スコアとする密ベクトル検索 [4]，疎ベクトルに変換する SPLADE [5]，クエリの各トークンベクトルに対して，それぞれ文書の全トークンベクトルとの内積を計算し，その類似度が最大のものの和をとる ColBERT[6] などが挙げられる．しかしながら，どの検索モデルを訓練する場合も，大量の教師データが必要であるため，適用できるドメインが限定される．また，大量の教師データが存在するドメインで訓練した検索モデルは，語彙が大きく異なるドメインに適用した場合に，検索精度が大きく低下することが知られている．

そのため，ドメイン外において教師なしで検索モデルの精度向上を試みる研究が進められており，擬似クエリを使用した密ベクトル検索の教師なしドメイン適応 [7, 8]，事前学習済みモデルの継続事前学習を用いた教師なしドメイン適応 [9]，IDF を通じた重み調整によるドメイン適応 [9]，語彙一致検索との併用 [7, 10, 11] などが提案されている．

しかしながら，これらのドメイン適応手法を複数用いた場合の効果は明らかになっていない．また，あるドメイン適応手法がどの検索モデルに対して効果的に働くのか，俯瞰的に分析した事例はない．

そこで，本研究では，まず**各ドメイン適応手法がどのような検索モデルに対して効果があるか**を明らかにするために，事前学習済みモデルを用いた代表的な検索モデルである，密ベクトル検索，SPLADE，ColBERT に対して，擬似クエリを使用したドメイン適応，事前学習済みモデルのドメイン適応，重み調整によるドメイン適応，及び語彙一致検索（BM25）との併用を適用した．その結果，どの検索モデルでも大きな効果を示すドメイン適応方法は，事前学習済みモデルのドメイン適応と語彙一致検索との併用であることが分かった．

次に，**ドメイン適応手法を複数適用した場合に相乗効果を示すか**を明らかにするために，ドメイン適応手法を複数適用した場合について実験を行なった．その結果，事前学習済みモデルのドメイン適応と語彙一致検索との併用を一緒に用いる場合が複数の検索モデルで有効であること，この二つを適用した場合に擬似クエリを使用する手法と重み調整は性能向上にほとんど寄与してないことが分かった．

語彙一致検索の併用は複数の検索モデルのアンサンブルと見なせる．そこで，**検索モデルによるアンサンブルは更なる検索精度の向上をもたらすか**を検証した．事前学習済みモデルのドメイン適応と語

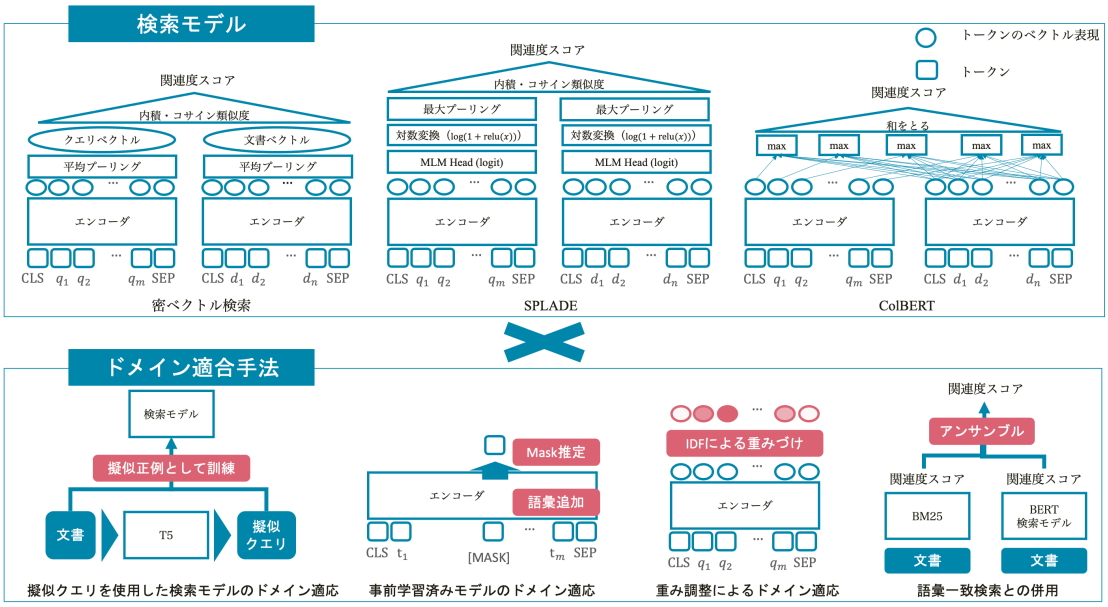

図 1　実験で使用した検索モデルとドメイン適応手法の概要図

彙一致検索との併用を適用すると共に，密ベクトル検索，SPLADE，ColBERT のアンサンブルを行い，現時点での世界最高の性能 [9, 12] を上回る結果を得た．

まとめると，本論文の貢献は以下の通りである．

- 異なる検索モデルのアンサンブルが，ドメイン適応に対して有効であることを示した．
- 事前学習済みモデルのドメイン適応が，アンサンブルと相乗効果をもたらすことを示した．

2　実験手法

本研究では，事前学習済みモデルに基づく検索モデルを語彙が大きく異なるドメインに適用する際に，ドメイン適応手法の有効な組み合わせを明らかにすることにある．以下では，今回用いたドメイン適応手法を紹介する．

擬似クエリを使用した検索モデルのドメイン適応として，**GPL** [8] を用いる．GPL では，元ドメインにおける正例のクエリ・文書ペアを教師データとし，T5 [13] を用いて文書を入力としたクエリ生成器を学習する．そして，対象ドメインにおいて各文書からクエリを生成し，そのペアを正例とする．次に，生成したクエリで学習対象とする検索モデルで検索を行い，負例を生成する．その後，クロスエンコーダの検索モデル[1] [3] を用いてスコアリングをする．クロスエンコーダは，クエリと文書を結合トークン（BERT の場合は [SEP]）を用いて連結して，検索モデルとしてファインチューニングした事前学習済みモデルに入力し，クエリと文書の関連度スコアを推定する方式である．最後に，クロスエンコーダのスコアを用いて，知識蒸留で検索モデルの訓練を行う．

事前学習済みモデルのドメイン適応として，**AdaLM** [14] を用いる．AdaLM は，対象ドメインのコーパスを用いて，語彙拡張とマスク付き言語モデルによる継続事前学習を行う手法である．これは，Iida and Okazaki [9] において，検索タスクでの有効性が示されている．

重み調整によるドメイン適応において，IDF を用いた．本稿ではこの手法を **IDF 重み**と呼ぶ．密ベクトル検索では重みつき平均によって重みを反映させている．SPLADE はエンコーダの語彙次元の疎ベクトルにテキストをエンコードする．そのため，文書側のベクトルの各要素に直接 IDF の重みを付けた．ColBERT については，クエリのベクトルを IDF で重み付けした．なお，検索文書中に存在しない語彙については，重みを 1 とした．

語彙一致検索との併用において，語彙一致検索には，BM25 を採用した[2]．本論文ではこの手法を **BM25 併用**と呼ぶ．併用方法は，各検索モデルと BM25 のそれぞれ上位 100 件の文書のスコアの和とした[3]．和を計算する際，片方の検索結果にしか出

1）　cross-encoder/ms-marco-MiniLM-L-6-v2 を使用した．

2）　具体的には，pyserini [15] を用いた．

3）　積を計算する方法 [16] も提案されているが，和を計算す

表 1 各検索モデルにドメイン適応手法を適用した結果．適用なしの場合については，数値は NFCorpus, TREC-COVID, SCIDOCS, Scifact データセットの nDCG@10 の平均値である．ドメイン適応手法を適用した結果は適用なしの場合からの差（向上幅）である．各モデルで最も性能が向上したものを太字にしている．

ドメイン適応手法	密ベクトル	SPLADE	ColBERT
適用なし	0.433	0.452	0.451
GPL	+0.033	+0.016	+0.010
AdaLM	+0.042	**+0.031**	**+0.031**
IDF 重み	+0.005	+0.020	+0.005
BM25 併用	**+0.049**	+0.020	+0.024

現しない文書のスコアとして，もう片方の検索結果 100 位の文書のスコアを用いた．

事前学習済みモデルを利用した検索モデルのアンサンブルでは，語彙一致検索との併用の場合と同様に，上位 100 件の文書のスコアの和を計算した．

3 結果

本節では，事前学習済みモデルを用いた各検索モデルに対して単一のドメイン適応手法を用いた場合の結果，複数の手法を組み合わせて用いた場合の結果，及び有効なドメイン適応手法を用いて複数の検索モデルをアンサンブルした結果を述べる．データセットは，ドメイン外での検索性能を計測する際に広く使われている BEIR [17] を用いた．BEIR は複数のデータセットから構成されており，その中でも NFCorpus [18]，TREC-COVID [19]，SCIDOCS [20]，Scifact [21] を用いた．また，3.3 節では BioASK [22] も用いた．これらは，重みつき Jaccard 係数で計測した場合，検索モデルを学習した際のデータセットである MS MARCO [23] から最も離れたデータセットである [17]．その他の実験設定の詳細は付録 A に記した．

3.1 各ドメイン適応手法はどのような検索モデルに対して効果があるか

各ドメイン適応手法の性能を表 1 に記す．全てのドメイン適応手法が検索の精度を改善していることが分かる．特に，BM25 併用と AdaLM は性能を大きく向上させている．BM25 併用は密ベクトル検索で最も検索精度を向上させた．これは，Formal ら [24] が指摘するように，事前学習済みモデルを用いた検索モデルが教師データ中の低頻度な単語の重要度を低く捉えてしまうという状況を緩和するためと考えられる．BM25 併用を用いることで，単独では最も検索精度が低かった密ベクトル検索が他の検索モデルを上回っている．

る方が良好な結果を示した．

表 2 各検索モデルに，AdaLM と一緒に他のドメイン適応手法を適用した結果．数値は NFCorpus，TREC-COVID，SCIDOCS，Scifact データセットの nDCG@10 の平均値である．他のドメイン適応手法を適用した結果は AdaLM を適用した場合からの差（向上幅）である．各モデルで性能が最も向上したものを太字にしている．

ドメイン適応手法	密ベクトル	SPLADE	ColBERT
AdaLM	0.475	0.484	0.483
+GPL	-0.006	-0.010	-0.002
+IDF 重み	-0.003	+0.002	+0.005
+BM25 併用	**+0.034**	**+0.014**	**+0.012**

他のドメイン適応手法の中では，AdaLM が SPLADE と ColBERT に対して最も効果的であった．擬似クエリを用いる GPL は，密ベクトル検索で有効であるが，他のモデルにおいてあまり改善していない．また，IDF 重みは SPLADE では高い効果を示すが，他のモデルでは効果が薄かった．

3.2 ドメイン適応手法を複数適用した場合に相乗効果があるか

次に，表 1 の実験で全般的に改善効果が最も高かった AdaLM を選び，さらに他のドメイン適応手法を適用した場合の結果を表 2 に記す．この実験では，BM25 併用が全てのモデルにおいて検索精度の向上をもたらした．よって，AdaLM には単語重要度の補正以外の効果があると考えられる．例えば，SPLADE や ColBERT のモデルを考えると，ドメイン特有の単語の埋め込みの学習が進んでいることが期待される．

擬似クエリを用いる GPL は，全ての検索モデルで検索精度が低下した．しかしながら，データセットにおける性能を詳しく調べると，その効果はデータセットによってばらつきがあった．密ベクトル検索と ColBERT では，TREC-COVID で大幅に性能が低下していたため，平均での検索精度の低下が見られたが，それ以外のデータセットでは改善が見られた．また，SPLADE では，NFCorpus と Scifact で検索精度が低下している．このように，事前学習済み言語モデルのドメイン適応を行った場合は，その効果はデータセットやモデルに依存していた．

表 1 では IDF 重みが SPLADE での性能の改善に大きく寄与していたが，表 2 ではその効果がほぼ見られなくなった．密ベクトル検索では，むしろ IDF 重みは逆効果となっている．IDF 重みは対象ドメインにおける重要語に着目する効果があると考えられるが，AdaLM でもその効果が果たされているためと考えられる．

さらに，AdaLM と BM25 併用に加えて GPL 及び IDF 重み適用した場合について実験を行なったが，ほとんど効果はなかった．結果は付録 B に記した．

3.3 検索モデルによるアンサンブルは更なる検索精度の向上をもたらすか

BM25 併用は，事前学習済みモデルを用いた検索モデルとのアンサンブルと見なすことができる．そこで，事前学習済みモデルを用いた検索モデル間でアンサンブルをすることで，検索精度の更なる改善を試みた．表 2 で最も高い検索精度に達した AdaLM と BM25 併用を適用した密ベクトル検索に対し，SPLADE と ColBERT をアンサンブルした．実験結果を表 3 に記す．密ベクトル検索と SPLADE をアンサンブルした場合，密ベクトル検索と ColBERT をアンサンブルした場合のどちらも，検索の精度に改善が見られる．よって，AdaLM 及び BM25 併用によるドメイン適応と事前学習済みモデルを用いた検索モデルのアンサンブルは検索精度のさらなる改善をもたらした．一方，密ベクトル検索，SPLADE，ColBERT を全てアンサンブルした場合は，密ベクトル検索と ColBERT をアンサンブルした場合と同程度の検索精度に留まった．

次に，既存の最高性能の検索モデルと比較する．密ベクトル検索の手法である COCO-DR [12] は，我々の知る限り NFCorpus, TREC-COVID, SCIDOCS, Scifact データセットの nDCG@10 の平均で最も高い値に到達している．また，SPLADE に AdaLM，IDF 重み，BM25 併用を適用した CAI [9] は，先ほどの 4 件のデータセットに BioASK を加えた 5 件のデータセットの nDCG@10 の平均で最高値を示している．表 3 においてこれらの検索モデルと比較すると，AdaLM 及び BM25 併用を適用した検索モデルのアンサンブルが 4 データセットでは COCO-DR と同等の値を示しており，5 データセットでは ColBERT とアンサンブルした場合に最も高い値に達した．

複数検索モデルのアンサンブル単独による効果を検証するため，AdaLM を適用しない場合について，各検索モデルのアンサンブルを行った．ドメインによる違いを観察するために，教師データと同じドメインである MS MARCO でも実験を行っている．MS MARCO において最も検索精度の高かった ColBERT に対して，密ベクトル検索と SPLADE をアンサンブルした．これに加えて，BM25 併用を適用した実験も行った．その結果を表 4 に示す．教師データのドメイン外である 4 データセットの平均において，

表 3 AdaLM と BM25 併用を適用した密ベクトル検索に AdaLM を適用した SPLADE と ColBERT をアンサンブルした結果．数値は NFCorpus，TREC-COVID，SCIDOCS，Scifact の四データセットにおける nDCG@10 の平均とさらに BioASK を加えた五データセットにおける nDCG@10 の平均．それぞれ最も性能が良いものを太字にしている．

検索モデル	4 データセット	5 データセット
CAI [9]	0.499	0.513
COCO-DR [12]	0.515	0.501
密ベクトル+AdaLM +BM25 併用	0.509	0.507
+SPLADE	**0.516**	0.515
+ColBERT	**0.516**	**0.522**
+SPLADE+ColBERT	**0.516**	0.521

表 4 ColBERT に対して密ベクトル検索と SPLADE のアンサンブルを行った結果．数値は MS MARCO における nDCG@10 の値と NFCorpus，TREC-COVID，SCIDOCS，Scifact の四データセットの nDCG@10 の平均値である．

検索モデル	MS MARCO	4 データセット
ColBERT	0.749	0.452
+SPLADE	0.749	0.474
+密ベクトル	0.726	0.477
+SPLADE+密ベクトル	0.738	0.488
+SPLADE+密ベクトル +BM25 併用	0.752	0.493

アンサンブルを行うことで検索精度が改善した．また，密ベクトル検索，SPLADE，ColBERT の全ての検索モデルをアンサンブルし，BM25 併用を適用した場合に，最も高い数値となった．しかしながら，どの結果も表 3 のドメイン適応した検索モデルのアンサンブル結果を下回っている．この結果より，表 3 において，特に AdaLM が検索モデルのアンサンブルと相乗効果をもたらすと考えられる．

なお，教師データと同ドメインである MS MARCO においては，アンサンブルを行っても ColBERT の結果から改善していない．単純な和によるアンサンブルはドメイン適応において有効と推察される．

4 結論

本論文では，検索モデルのドメイン適応において，事前学習済みモデルのドメイン適応，語彙一致検索との併用，およびその組み合わせを検証し，語彙が大きく異なるデータの場合に特に効果的であることを示した．また，語彙一致検索のみならず，複数の検索モデルのアンサンブルが有効であること，その場合も事前学習済みモデルのドメイン適応と相乗効果があることを示した．今後の課題として，検索モデル相互の特性をより活用したアンサンブル手法の探求などを考えている．

謝辞

この成果は, 国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務（JPNP18002）の結果得られたものです.

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 **NAACL**, pp. 4171–4186, 2019.

[2] S E Robertson and S Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In **SIGIR**, pp. 232–241, 1994.

[3] Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with BERT. **arXiv:abs/1901.04085**, January 2019.

[4] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-Tau Yih. Dense passage retrieval for Open-Domain question answering. In **EMNLP**, pp. 6769–6781, 2020.

[5] Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. SPLADE v2: Sparse lexical and expansion model for information retrieval. **arXiv:abs/2109.10086**, 2021.

[6] Omar Khattab and Matei Zaharia. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In **SIGIR**, pp. 39–48, 2020.

[7] Ji Ma, Ivan Korotkov, Yinfei Yang, Keith Hall, and Ryan McDonald. Zero-shot neural passage retrieval via domain-targeted synthetic question generation. In **EACL**, pp. 1075–1088, 2021.

[8] Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. GPL: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In **NACL**, pp. 2345–2360, 2022.

[9] Hiroki Iida and Naoaki Okazaki. Unsupervised domain adaptation for sparse retrieval by filling vocabulary and word frequency gaps. In **AACL-IJCNLP**, pp. 752–765, 2022.

[10] Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. Sparse, dense, and attentional representations for text retrieval. **TACL**, Vol. 9, pp. 329–345, 2021.

[11] Luyu Gao, Zhuyun Dai, Tongfei Chen, Zhen Fan, Benjamin Van Durme, and Jamie Callan. Complement lexical retrieval model with semantic residual embeddings. In **ECIR**, Vol. 12656, pp. 146–160, 2021.

[12] Yue Yu, Chenyan Xiong, Si Sun, Chao Zhang, and Arnold Overwijk. COCO-DR: Combating distribution shifts in Zero-Shot dense retrieval with contrastive and distributionally robust learning. 2022.

[13] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. **JMLR**, Vol. 21, No. 140, pp. 1–67, 2020.

[14] Yunzhi Yao, Shaohan Huang, Wenhui Wang, Li Dong, and Furu Wei. Adapt-and-distill: Developing small, fast and effective pretrained language models for domains. In **Findings of ACL-IJCNLP 2021**, pp. 460–470, 2021.

[15] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In **SIGIR**, pp. 2356–2362, 2021.

[16] Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. LaPraDoR: Unsupervised Pretrained Dense Retriever for Zero-Shot Text Retrieval. In **Findings of ACL**, pp. 3557–3569, 2022.

[17] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In **NeurIPS**, 2021.

[18] Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. A Full-Text learning to rank dataset for medical information retrieval. In **ECIR**, pp. 716–722, 2016.

[19] Ellen Voorhees, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, William R. Hersh, Kyle Lo, Kirk Roberts, Ian Soboroff, and Lucy Lu Wang. Trec-covid: Constructing a pandemic information retrieval test collection. **SIGIR Forum**, Vol. 54, No. 1, feb 2021.

[20] Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel Weld. SPECTER: Document-level representation learning using citation-informed transformers. In **ACL**, pp. 2270–2282, 2020.

[21] David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. Fact or fiction: Verifying scientific claims. In **EMNLP**, pp. 7534–7550, 2020.

[22] George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, Yannis Almirantis, John Pavlopoulos, Nicolas Baskiotis, Patrick Gallinari, Thierry Artiéres, Axel-Cyrille Ngonga Ngomo, Norman Heino, Eric Gaussier, Liliana Barrio-Alvers, Michael Schroeder, Ion Androutsopoulos, and Georgios Paliouras. An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. **BMC Bioinformatics**, Vol. 16, p. 138, April 2015.

[23] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human generated machine reading comprehension dataset. In **NIPS**, Vol. 1773 of **CEUR Workshop Proceedings**, 2016.

[24] Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. Match your words! a study of lexical matching in neural information retrieval. In **ECIR**, p. 120–127, 2022.

[25] Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. S2ORC: The Semantic Scholar Open Research Corpus. In **ACL**, pp. 4969–4983, 2020.

[26] Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In **SIGIR**, pp. 113–122. 2021.

表 5 密ベクトル検索学習時のハイパーパラメータ

バッチサイズ	64
最大文書長	300
学習率	2e-5
エポック	30
Warmup ステップ数	1000

表 6 SPLADE 学習時のハイパーパラメータ

バッチサイズ	40
最大文書長	256
学習率	2e-5
エポック	30
Warmup ステップ	1000

表 7 ColBERT 学習時のハイパーパラメータ

バッチサイズ	32
最大文書長	220
学習率	3e-6
一クエリあたりの負例数	48
エポック	1
最大訓練ステップ数	500,000

表 8 GPL 使用時のハイパーパラメータ（ColBERT を除く）

一文書あたりの生成クエリ数	3
バッチサイズ	24
最大文書長	350
学習率	2e-5
訓練ステップ数	140000
Warmup ステップ数	1000

A 実験設定

今回使用した対象データは，BioASK [22]，NFCorpus [18]，TREC-COVID [19] がバイオ・医療ドメインであり，SCIDOCS [20]，Scifact [21] が科学ドメインである．そのため， AdaLM を適用するコーパスとして，バイオ・医療ドメインは PubMed [4] の要旨を用いた．科学ドメインは S2ORC [25] の要旨を用いた．

元ドメインにおける検索モデルの学習については，全ての検索モデルでクロスエンコーダ[5] [3] を知識蒸留することで訓練した．知識蒸留の方法として，密ベクトル検索と SPLADE では，あるクエリに対する正例文書と負例文書のクロスエンコーダにおけるスコアの差であるマージン二乗誤差損失 [26] を用いた．また，密ベクトル検索と SPLADE の関連度スコアは内積を用いて計算した．ColBERT については，クロスエンコーダにおけるスコアと検索モデルのスコアそれぞれに対してソフトマックス関数を適用し，KL ダイバージェンスをとった．GPL 学習時にも同様に知識蒸留を行った．

4） https://pubmed.ncbi.nlm.nih.gov/

5） モデルには，cross-encoder/ms-marco-MiniLM-L-6-v2 を使用している．

表 9 各検索モデルに，AdaLM，BM25 併用と一緒に他のドメイン適応手法を適用した結果．数値は NFCorpus, TREC-COVID, SCIDOCS, Scifact データセットの nDCG@10 の平均値である．他のドメイン適応手法を適用した結果は AdaLM と BM25 併用を適用した場合からの差（改善幅）である．

ドメイン適応手法	密ベクトル	SPLADE	ColBERT
AdaLM+BM25 併用	0.509	0.497	0.494
+GPL	-0.016	0.000	-0.001
+IDF	-0.003	-0.001	+0.004
+GPL+IDF 重み	-0.017	-0.006	+0.001

各検索モデルの事前学習済みモデルには BERT[6] を用いた．学習時のハイパーパラメータを表 5，6，7 に記す．また，密ベクトル検索と SPLADE に対して，GPL を適用した際のハイパーパラメータを表 8 に記す．なお，ColBERT は GPL 適用時も表 7 と同様である．また，一文書あたりのクエリ数は表 8 と同様である．

B AdaLM・BM25 併用に加えてドメイン適応を行なった結果

AdaLM と BM25 併用に加えて，GPL，IDF を適用した場合の実験結果を表 9 に記す．GPL はどの手法においても，検索精度の向上に寄与していない．密ベクトル検索ではその検索精度が大きく低下しており，各データセットの詳細を調べても，改善が見られたのは SCIDOCS のみであった．密ベクトル検索において，GPL による改善効果の多くが AdaLM と BM25 併用で実現されていたと推察される．IDF 重みについては，検索モデルによって差があり，密ベクトル検索及び SPLADE は検索精度が下がっている．BM25 は TF-IDF によるランキングアルゴリズムであるため，BM25 併用によってすでに IDF 重みと同様の効果が得られたことが要因と考えられる．一方，ColBERT では改善している．AdaLM による埋め込みの改善が，語彙 BM25 併用による効果以外にも改善をもたらしたと考えられる．

6） bert-base-uncased を使用している．