人材領域特化の LLM 教師付き長文 BERT 埋め込みモデルの構築

原 龍昊 [1] 長谷川 崇 [1]Dat P.T. Nguyen[1]
[1] 株式会社ビズリーチ
{longhao.yuan, takashi.hasegawa, dat.nguyen}@bizreach.co.jp

概要

大規模言語モデル（LLM）は高精度なテキスト理解を実現する一方，大量の文書処理では推論コストと応答速度が課題となり，外部 API 利用時にはデータ送信に伴うセキュリティ懸念も生じる．

本研究では，LLM を教師とした知識蒸留（Knowledge Distillation）により，求人マッチングの機会創出に特化した高性能かつ軽量な BERT 埋め込みモデルを構築する手法を提案する．提案手法では，企業内で蓄積された求人文書のテキストペアに対して LLM を用い意味的類似度スコアを付与し，生成された大規模ソフトラベルデータセットで長文処理に特化した BERT モデルを訓練する．評価実験の結果，提案モデルは総合性能で最高値を達成し，OSS 埋め込みモデルを大幅に上回り，OpenAI Embedding API とも同等以上の性能を示すことを確認した．

1 はじめに

近年，ChatGPT に代表される大規模言語モデル（LLM）は，高精度なテキスト理解を実現し，多様なタスクで高い性能を発揮している．一方で，膨大なパラメータ数に起因する推論コストの高さや応答速度の低下，さらに外部 API 利用時にはデータ送信に伴うセキュリティ上の懸念も指摘されている．

一方，人材業界では求人文書やレジュメ（職務経歴書）などの大量のテキストデータが日々蓄積されており，これらの意味的類似度を効率的に計算する技術が求められている．特に，推薦システムにおける新規ユーザや新規求人へのマッチングでは，テキスト内容に基づく特徴量が重要となる．

このような問題に対し，これまでは TF-IDF や LDA（Latent Dirichlet Allocation）[1] などの統計的手法や，汎用 BERT モデルが用いられてきた．しかし，統計的手法では文脈を十分に捉えられず，汎用 BERT モデルでは業界特有の専門用語や表現への対応が難しいという課題がある．

本研究では，LLM を教師とした知識蒸留（Knowledge Distillation）[2] により，人材マッチング領域の支援に特化した軽量で高速な長文 BERT 埋め込みモデルを構築した．具体的には，企業内で蓄積された求人文書のテキストペアに対し，LLM を用いて意味的類似度スコアを付与した．この大規模なソフトラベルデータセットを用いて，長文処理に対応した BERT 埋め込みモデルを訓練した．評価実験では，複数のモデルサイズを構築し，OSS 埋め込みモデルおよび OpenAI Embedding API との比較を実施した．その結果，提案モデルは総合評価で最高値を達成し，OSS モデルを大幅に上回り，OpenAI Embedding API と同等以上の性能を実現した．

本研究の主な貢献は以下の 3 点である．

- ドメイン特化型軽量 BERT 埋め込みモデルの LLM 知識蒸留による構築フレームワーク
- サンプリング戦略と LLM 自動ラベリングを統合した効率的な意味的類似度（Semantic Textual Similarity, STS）データセット生成手法
- 軽量モデルによる商用 API 同等性能の達成とその有効性の実証

2 関連研究

2.1 知識蒸留

知識蒸留 [2] は，大規模な教師モデルの知識を小規模な生徒モデルに転移させる技術である．近年では，ChatGPT などの LLM により生成されたアノテーションを利用し，小規模モデルを教師 – 生徒フレームワークで学習させる研究が報告されている [3, 4]．本研究では，LLM を用いてテキストペアの意味的類似度スコアを生成し，これを教師信号として BERT 埋め込みモデルを訓練した．

2.2 文埋め込みモデル

Sentence-BERT[5] は，Siamese ネットワーク構造を用いて BERT をファインチューニングすることで高品質な文埋め込みを生成する手法である．大規模なテキストペアデータセットを用いた対照学習 (Contrastive Learning) による訓練も注目されている [6]．本研究では，Sentence-BERT の枠組みを採用し，LLM が生成したスコアで BERT モデルを訓練した．

2.3 人材業界への応用

人材業界では，BERT を用いた求人文書のキーワード抽出 [7] や求人とレジュメのマッチング [8] などの研究が行われている．原ら [9] は，日本語求人文書のマルチラベル分類においてドメイン特化 BERT の有効性を示した．本研究では，求人文書を用いた埋め込みモデルの学習に焦点を当てた．

3 提案手法

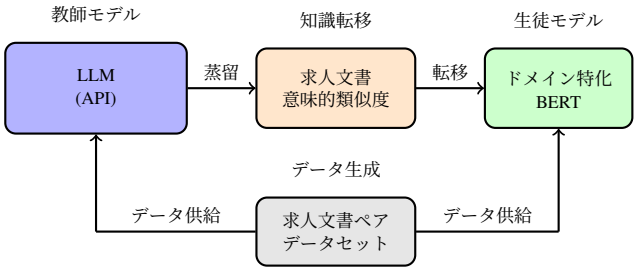

図 1 提案手法の概要図

提案手法は以下の 3 つのステップからなる．

1. データセットの構築：サンプリング戦略に基づく求人文書ペアの構築
2. テキストペアのラベリング：LLM を用いた意味的類似度スコアの自動生成
3. BERT モデルのファインチューニング：生成されたデータを用いた埋め込みモデルの学習

本研究の提案手法の概略は図 1 に示す．以下，各ステップの詳細を述べる．

3.1 STS データセットの構築

本研究では，企業内で蓄積された求人文書のテキストデータを用いた．求人文書には，ポジション名，募集条件，仕事内容などの情報が含まれ，これらは数百文字から数千文字に及ぶ長文である．

近年，LLM を用いたテキスト埋め込みの高精度化において，大規模なテキストペアの意味的類似度データと対照学習を組み合わせた手法が高い性能を示している [6]．テキストペア間の意味的距離を直接学習することで，意味的に近い文は近く，異なる文は遠く配置される埋め込み空間が構築され，検索や類似度推定に適した分散表現が得られる．

しかし，STS データセット構築において，ランダムサンプリングでは高類似度ペアが極端に少なくなるという課題が存在する．本研究では，以下のサンプリング戦略を応用した．

- **25%：完全ランダムサンプリング**
  全テキストから無作為にペアを生成．低類似度ペアを含むことで，モデルの識別能力を向上させる．
- **75%：類似度が高い可能性のあるペアを優先サンプリング**
  同一候補者が返信した複数の求人を組み合わせてペア化することで，業務内容やスキル要件が類似する求人ペアを構築する．

この戦略により，後続の LLM による STS ラベリングにおいて，高から低まで幅広いスコアを持つテキストペアを含むデータセットを構築した．

3.2 LLM による意味的類似度スコアの生成

大規模テキストペアへの 人間による STS スコアの付与が不可能であるため，本研究では OpenAI API を用いて STS スコアを付与した．モデルは，精度・コスト・速度を総合的に考慮し，高速かつ軽量で対話や文章生成に適した GPT-4o-mini を採用した．

スコアリングプロンプト STS スコアの付与に使用したプロンプトを図 2 に示す．スコアは 1.0 〜 5.0 の範囲で付与され，職務内容・必要スキル・業務範囲の観点で所定の評価基準と出力形式で評価した．

倫理的配慮およびバイアス対策 データの収集にあたっては，プライバシーポリシーに基づき，ユーザの同意を得た上で公開されている求人文書のみを使用した．また， OpenAI API の利用にあたっては，データがモデルの訓練に使用されない設定とし，プライバシーとセキュリティに配慮した．

LLM への入力には，年齢・性別・学歴などの個人属性情報を一切含めず，テキスト内容のみに基づいた評価を行うことで，バイアスの増幅を防いだ．また，推薦結果にバイアスが生じることを防ぐため，求人と候補者間のスカウト送信や返信，応募などの行動データは使用しない設計とした．

あなたは、求人文書同士の意味的類似度（STS: Semantic Textual Similarity）を、職務内容・必要スキル・業務範囲に着目して評価するアシスタントです。
以下の 2 つの求人文書について、「職務内容」「必要スキル」「業務範囲」の観点から意味的な類似度を評価してください。
1.0 〜 5.0 のスコア（小数点第 1 位まで）で、以下の評価基準に従って判断してください。
【評価基準】
- 1.0 〜 2.0: 類似性がほとんどない（業種や業務内容が大きく異なる）。
- 2.0 〜 3.0: 一部に共通するスキルや業務が見られるが、全体としては別の職種に相当する。
- 3.0 〜 4.0: 主な業務内容やスキルに明確な重なりがある。中程度の類似性。
- 4.0 〜 5.0: 表現の違いを除けば、職務内容・スキル・業務範囲はほぼ同じ。高い類似性。

【出力形式】
スコアのみを、「[[rating]]」の形式で出力してください（例：[[4.3]]）。スコア以外のコメントや説明は出力しないでください。
[求人文書 1]
[求人文書 2]
以上を踏まえて、2 つの求人文書の意味的な類似度は：

図 2　STS スコアリングプロンプト

データ品質の評価　LLM スコアリングの妥当性を確認するため，出力されたスコアの 1.0 – 2.0，2.0 – 3.0，3.0 – 4.0，4.0 – 5.0 の 4 区間から各 5 件，計 20 件を均等に抽出した．抽出したペアを人材業界のアノテータ 3 名が独立にスコアリングし，アノテータの平均スコアと LLM スコアを比較した．

アノテータの平均スコアと LLM スコア間の Pearson 相関係数は 0.932，MAE（平均絶対誤差）は 0.387 であった．人間判断との線形一致性は極めて高く，STS（1.0 〜 5.0）のスケールを考慮すると MAE も許容範囲である．この結果から，LLM による STS スコアは十分信頼できると判断した．

教師データセットの規模　80,000 組の求人文書テキストペアに対して STS スコアを付与することで，教師データセットを構築した．そのうち，70%を train，15%を validation，15%を test として分割し，ベンチマークデータセットとして利用した．

3.3　BERT モデルのファインチューニング

テキストペアとその STS スコアを含むデータセットを用いて，事前学習済み BERT 系モデルを Sentence-BERT 方式でファインチューニングした．

モデル構成と事前学習　本研究で用いる BERT モデルは，大量の求人データを高速に処理するために設計された，総パラメータ数が約千万規模の軽量モデルである．最大入力長は 2048 tokens とし，実際の利用環境における長文処理能力を確保した．また，Transformer 層数の異なる 12M / 30M / 45M（Million，百万）の 3 種類のモデルサイズを設計し，アプリケーション要件に応じて選択できるようにした．

分かち書きには SudachiPy[10] を採用し，BERT モデル自体は Transformers ライブラリを用いて実装した．事前学習は，社内の大規模テキストデータと Wikipedia 日本語コーパスを組み合わせ，Masked Language Modeling（MLM）に基づく教師なしの事前学習手法によって実施した．この事前学習済み BERT を SentenceTransformer フレームワーク[1)]に組み込み，文埋め込み生成モデルとして用いた．

ファインチューニング　各データサンプルはテキストペアとその STS スコアから構成される．元のスコアは 1.0 〜 5.0 の連続値で付与されていたため，類似度学習に適した 0 〜 1 の範囲に正規化した．

テキストペアの意味的近さを反映した埋め込み空間を学習するため，損失関数には Sentence-BERT で広く利用されている CosineSimilarityLoss を採用した．この損失は，文埋め込み間のコサイン類似度を正解スコアに近づけるよう最適化するものであり，STS タスクに適している．

4　実験

4.1　比較方法

先述の STS データセットを用いてファインチューニングした 3 つのモデルサイズ（12M，30M，45M）の BERT の性能を検証するため，以下に示す外部モデルと併せて比較評価を行った．

- **OSS 埋め込みモデル**：日本語に特化した公開埋め込みモデルとして，名古屋大学が公開する `cl-nagoya/ruri-small`[2)]（68M パラメータ）と，PKSHA Technology が公開する `pkshatech/GLuCoSE-base-ja-v2`[3)]（133M パラメータ）を評価対象とした．これらは日本語コーパスで事前学習されたモデルである．
- **OpenAI Embedding API**：商用の大規模多言語埋め込みモデルである `text-embedding-3-large` を評価した．本モデルは多様なドメインで訓練された汎用モデルである．標準次元（3048）に

1）　https://www.sbert.net/
2）　https://huggingface.co/cl-nagoya/ruri-small
3）　https://huggingface.co/pkshatech/GLuCoSE-base-ja-v2

加え，商用環境での実運用を想定した 256 次元および 512 次元のモデルも評価対象とした．

4.2 実験設定

提案モデルの有効性を検証するため，STS 計測に関するタスクとより実用的なマッチングに関するタスクの 2 つの観点から評価を行った．実験には，STS データセットのテストデータを用いた．

STS 計測に関するタスク 教師とした LLM による STS をモデルがどの程度再現できているかを評価するための実験を行った．まず，各モデルによりテストデータの文埋め込みを算出し，テキストペア間のコサイン類似度を計測した．次に，モデルから得られたコサイン類似度と LLM が付与した STS スコアとの相関を算出した．評価指標には Spearman の順位相関係数を用いた．

マッチングに関するタスク（推薦タスク） 実際のビジネス環境では，**企業の採用担当者に候補者を推薦するタスク**[4]が発生する．このようなタスクに対するモデルの有用性を検証するため，企業内リクルーターによる過去のスカウト送信履歴を用いた推薦実験を行った．まず，ローカル環境において，求人文書および候補者レジュメから文埋め込みを算出し，埋め込み間のコサイン類似度を求人と候補者のマッチ度として計測した．次に，各求人に対して全候補者をマッチ度に基づいてランキングし，ランキング上位 K 件以内に実際にスカウトメールが送信された候補者が含まれる割合を Hit@K 指標として算出し，マッチング性能を評価した．

表 1 各モデルの評価結果

モデル	Emb. Size	STS タスク	推薦タスク	Avg.
（提案）BERT-45M-job-sts	576	91.66	46.09	**68.88**
（提案）BERT-30M-job-sts	512	**91.67**	44.24	67.95
OpenAI-Emb3-Large-3048	3048	85.70	**51.46**	68.58
（提案）BERT-12M-job-sts	256	90.41	40.03	65.22
OpenAI-Emb3-Large-512	512	84.38	49.04	66.71
OpenAI-Emb3-Large-256	256	81.74	45.47	63.60
ruri-small-68M	768	71.26	13.81	42.54
GLuCoSE-base-ja-v2-133M	768	61.38	26.48	43.93

4.3 実験結果と考察

スケールが異なる各評価タスクの指標を 0 〜 100 に正規化し，その平均値を総合スコア（Avg.）として算出した．全 8 モデルの結果を表 1 に示す．

総合スコア（Avg.）では，提案モデル BERT-45M-job-sts が 68.88 で最高値を達成し，OpenAI-Emb3-Large-3048（68.58）と僅差で上回った．提案モデルは上位 3 位中 2 モデルを占め，OSS モデル（ruri-small，GLuCoSE-base）を大きく上回った．領域特化型の学習が，意味的類似度推定および候補者推薦において有効であることが確認された．

タスク別の性能を見ると，STS タスクでは提案モデルが圧倒的な優位性を示した．BERT-30M-job-sts と BERT-45M-job-sts はそれぞれ 91.67 と 91.66 で最高値を記録し，OpenAI-Emb3-Large-3048（85.70）を大きく上回った．これは，LLM 教師付き学習により，求人文書の意味的類似度を高精度に捉える埋め込み表現が獲得されたことを示している．

一方，推薦タスクでは OpenAI-Emb3-Large-3048 が 51.46 で最高性能を示し，提案モデル BERT-45M（46.09），BERT-30M（44.24），BERT-12M（40.03）を上回った．OpenAI Embedding API は多様なドメインで訓練されているため，求人と候補者間の潜在的な適合性を捉える能力に優れていると考えられる．

提案モデルのモデルサイズに関しては，45M（68.88），30M（67.95），12M（65.22）の順で性能が向上し，モデル容量の増加が性能向上の主要因であることが確認された．一方，OpenAI Embedding API では埋め込み次元の縮小（3048 → 512 → 256）による性能低下は比較的小さく，次元圧縮に対する頑健性が示された．

5 まとめ

本研究では，LLM を教師とした知識蒸留により，求人マッチングの機会創出に特化した BERT 埋め込みモデルを構築する手法を提案した．

複数のモデルサイズを構築し，OSS 埋め込みモデルおよび OpenAI Embedding API と比較した結果，提案モデルが総合スコアで最高性能を達成し，OpenAI Embedding API とも同等以上の性能を示した．特に，意味的類似度推定タスクにおいて提案モデルが顕著な優位性を示し，LLM 教師付き学習による領域特化型モデル構築の有効性が確認された．

提案モデルは軽量で，ローカル環境での高速推論が可能なため，外部 API への依存を避けつつ高性能を実現できる点で実務的な価値の高さも示唆している．

4） ビズリーチにおける企業への推薦（レコメンド）は候補の方の提示にとどまり、直接人材を紹介することはありません

参考文献

[1] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, Vol. 3, pp. 993–1022, 2003.

[2] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1503.02531*, 2015.

[3] Jiazheng Li, Lin Gui, Yuxiang Zhou, David West, Cesare Aloisi, and Yulan He. Distilling ChatGPT for Explainable Automated Student Answer Assessment. *arXiv preprint arXiv:2305.12962*, 2023.

[4] Mingxuan Xia, Haobo Wang, Yixuan Li, Zewei Yu, Jindong Wang, Junbo Zhao, and Runze Wu. Prompt Candidates, then Distill: A Teacher-Student Framework for LLM-driven Data Annotation. *arXiv preprint arXiv:2506.03857*, 2025.

[5] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks. *arXiv preprint arXiv:1908.10084*, 2019.

[6] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple Contrastive Learning of Sentence Embeddings. *arXiv preprint arXiv:2104.08821*, 2021.

[7] Hussain Falih Mahdi, Rishit Dagli, Ali Mustufa, and Sameer Nanivadekar. Job Descriptions Keyword Extraction Using Attention Based Deep Learning Models with BERT. In *2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pp. 1–6. IEEE, 2021.

[8] Changmao Li, Elaine Fisher, Rebecca Thomas, Steve Pittard, Vicki Hertzberg, and Jinho D Choi. Competence-level Prediction and Resume & Job Description Matching Using Context-aware Transformer Models. *arXiv preprint arXiv:2011.02998*, 2020.

[9] 原龍昊, 林勝悟, Dat P.T. Nguyen. 人材業界固有の表現を考慮した求人票のマルチラベル分類. 言語処理学会第 30 回年次大会発表論文集, pp. 1746–1750, 2024.

[10] Kazuma Takaoka, Sorami Hisamoto, Noriko Kawahara, Miho Sakamoto, Yoshitaka Uchida, and Yuji Matsumoto. Sudachi: A Japanese Tokenizer for Business. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.