論文

レポートデータに対する文章類似度指標と可視化手法の比較分析の検討 –人工レポートデータを用いて–

岩田 員典（愛知大学 経営学部）

要旨

本研究では，大学のレポート課題のように「大きなテーマは共通だが内容や論点が異なる」文章集合を対象とし，複数のテキスト埋め込みモデルによる類似度計算および可視化手法の振る舞いを比較する．実際の学生レポートのみを用いた評価では，教師ラベルの粒度やデータ数の制約によりモデル間の差が明確に現れにくいという問題がある．そこで本研究では，大テーマを共通としつつ，執筆方針や論点構成の異なるレポートを擬似的に大量生成する人工レポートデータ生成手法を構築し，その上で TF–IDF，fastText，Sentence-BERT (SBERT)，E5，OpenAI Embeddings の 5 手法を比較する．さらに，t-SNE，UMAP による次元削減結果を可視化し，各モデルがレポートの構造をどの程度クラスタとして再現できるかを検討する．実験の結果，SBERT，E5，OpenAI Embeddings は人工レポートの執筆戦略ごとのクラスタをほぼ完全に分離できる一方で，TF–IDF と fastText はクラスタ構造を十分に再現できないことが定量評価・可視化の両面から確認された．これにより，レポート自動分類や類似レポート検出においては，従来広く用いられてきた BoW/単語平均型の埋め込みよりも，文レベル意味ベクトルモデルを採用することの有効性が示された．

キーワード： LLM，疑似レポート作成，レポート分類

1. はじめに

大学教育において，レポート課題は学習者の理解度や思考の深さを把握するための重要な評価手段である．しかし，提出されるレポートの数は多く，教員がすべてを精読し，内容の類似性や論点の偏りを把握することは容易ではない．このため，レポートの自動分類や類似レポート検出に向けた自然言語処理技術（Natural Language Processing: NLP）の活用が求められている．

従来，文章の類似度計算には TF–IDF による Bag-of-Words (BoW) 表現[1]や，word2vec[2, 3]の拡張である fastText[4]などの単語分散表現を平均した文ベクトルが広く用いられてきた．しかし，これらの手法は単語の出現頻度や局所的な共起に依存しており，長文レポートのように語彙が多様で構造が複雑な文章に対し

ては，執筆方針や論点の違いを十分に捉えられないことが指摘されている．

一方，近年は Transformer ベースの大規模言語モデル（Large Language Model: LLM）[5] を用いた文レベルの埋め込みモデルが多数提案されている．Sentence-BERT (SBERT) [6] や E5 [7] などは，類似文対や検索タスクを用いた対照学習により，意味的に類似した文を近傍に写像することを目的として設計されている．これらのモデルは，英語を中心としたベンチマークにおいて従来手法を大きく上回る性能を示しているが，日本語のレポートデータに対してどの程度有効であるかについては，体系的な検証が十分ではない．

さらに，教育実践で得られるレポートデータは，授業や年度によってテーマや難易度が異なり，事前に明確なラベルを付与することが難しいという課題がある．そのため，実データのみを用いて埋め込みモデルを比較しようとすると，「そもそも教師ラベルの粒度が粗すぎて差が出ない」「レポートの質がばらつきすぎて評価が安定しない」といった問題に直面する．

本研究では，これらの問題に対して次のアプローチをとる．まず，大テーマ（例：“生成 AI と社会”）を共通としつつ，複数の異なる執筆戦略（サブテーマや論点構成のパターン）を持つレポートを大規模に生成する人工レポートデータセットを構築する．このとき，文体や語彙をある程度ランダム化しつつ，戦略ごとの一貫性（クラスタ構造）が明確になるように設計する．次に，この人工レポートデータに対して 5 種類の埋め込みモデルを適用し，ペアワイズ類似度に基づく分類性能と，t-SNE [8]，UMAP [9] による可視化結果を用いて，各モデルの振る舞いを比較する．

2. 問題設定

本研究では，以下のようなレポート集合を対象とする．

- すべてのレポートは同一の大テーマ T（例：生成 AI と社会的影響）について書かれている．
- それぞれのレポートは，K 種類の執筆戦略のいずれかに基づき，主にどの観点から議論を展開するかが異なる（例：教育，産業，倫理，技術，利用者体験など）．
- 各レポートは数段落から構成され，導入・本論・結論の構造を持つ．
- 教員にとっては，「同じ戦略に属するレポート同士は類似度が高い」「異なる戦略のレポート同士は類似度が低い」ことが望ましい．

このとき，テキスト埋め込みモデル $f(\cdot)$ を用いて得られるベクトル表現 $\mathbf{x}_i =$

$f(d_i)$ に対し，コサイン類似度

$$\mathrm{sim}_{\cos}(d_i, d_j) = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\| \, \|\mathbf{x}_j\|}$$

が，執筆戦略ラベル y_i, y_j と整合的になることが望ましい．

しかし，実際の授業では，y_i を厳密に定義してラベル付けすることが難しい．そこで本研究では，レポートそのものを人工的に生成し，K 個の戦略ラベルを制御可能な形で与えることで，埋め込みモデルの性能を定量的に比較可能な環境を構築する．

3. 提案する人工レポート生成モデル

本章では，生成 AI を利用して人工レポートを作成する手法について述べる．本研究では，LLM の GPT-4 系モデル[10]を用いて多様な観点・構造を持つレポートを自動生成した．GPT 系モデルは Few-shot / Instruction-following に基づく高品質な長文生成が可能であり[11]，入力テンプレートやプロンプト設計に応じて文章構造・論点の多様化を制御できる点から，本研究の人工レポートデータ生成に適している．

提案手法は大きく次の 3 要素から構成される．

(1) **グローバルなテンプレート**：導入・結論の共通文を固定し，その間に生成される本文部分だけを変化させることで，レポート構造を一定に保ちながら内容の多様性を確保する．

(2) **執筆戦略の設計**：GPT-4 に対して複数の「執筆方針」（例：社会・文化的観点，技術的観点，制度設計的観点など）を指定し，レポート生成時の論点・内容展開が異なるように制御した．この手法は，LLM によるレポート生成研究や構造化プロンプト設計に関する先行研究[12]を参考にしている．

(3) **多様性付与のための摂動パラメータ**：temperature や randomness パラメータを変動させることで，文体・語彙選択・論点展開に揺らぎ（stochasticity）を付与し，分類性能評価に利用できる多様な人工レポート群を生成した．これにより，埋め込みモデル（特に OpenAI Embeddings[13]）に対して，内容類似度の評価を行うベンチマークデータセットを人工的に構成できる．

以上の 3 要素を組み合わせることで，多様性と一貫性を併せ持つ人工レポートデータセットを効率的に生成し，文書埋め込みモデルの評価実験に利用した．

3.1. 基本テンプレート

すべてのレポートは，共通の導入文と共通の結論文を持つテンプレートに基づ

く．導入では，テーマ T に関する一般的背景を説明し，結論では，AI 技術の可能性と課題を総括する．このように導入と結論を共通化することで，本論部分の違い（すなわち執筆戦略の違い）が埋め込み空間にどのように反映されるかを明確に観察できるようにした．さらに，本研究では導入文・結論文の複数の候補を用意し，後述する摂動パラメータに応じてランダムに選択することで，テンプレートそのものにも軽微な多様性を持たせている．

3.2. 執筆戦略の設計

大テーマ T に対して，$K = 10$ 種類の執筆戦略を定義した．各戦略は，主に焦点を当てる観点や議論の流れが異なるように設計している．一例として以下を挙げる．

- Strategy 1: 教育・学習支援への影響を中心とする．
- Strategy 2: 産業構造や雇用への影響に焦点を当てる．
- Strategy 3: 倫理・法制度上の課題を詳細に検討する．
- Strategy 4: モデル構造・性能評価など技術的観点を重視する．
- Strategy 5: クリエイティブ分野（芸術・表現）への影響を論じる．
- Strategy 6–10: 利用者体験，アクセシビリティ，国際格差，リスクコミュニケーションなど，別の観点をそれぞれ中心に据える．

各戦略に対して，生成 AI に与えるプロンプトでは，「どの観点を中心に議論するか」「どのような具体例を含めるか」「他の観点にどの程度触れるか」を自然言語で指定した．これにより，同じ戦略に属するレポート同士は類似した論点構成を持ちつつ，語彙や表現はランダム性を保つようにしている．

3.3. 多様性付与のためのパラメータ化

人工レポートデータが単調になりすぎると，どの埋め込みモデルでも容易にクラスタ分けできてしまい，モデル間の差が見えなくなる．そこで以下の 3 つのパラメータを導入し，多様性を制御した．

- P_1：**特徴観点数**　各レポートで必ず触れるべき話題の数を表す．例えば $P_1 = 12$ とした場合，教育・産業・倫理・技術など複数の観点を本論のどこかで扱うようにプロンプトで指定する．これにより，語彙レベルでは戦略ごとの差が出にくくなり，埋め込みモデルが論点構造そのものに基づいてクラスタを捉えられるかを評価できる．
- P_2：**文体レベルの混合比**　高品質・中

品質・低品質の 3 種類の文体をどの比率で混在させるかを表す．例えば $P_2 = 0.5 : 0.3 : 0.2$ とした場合，50% を論理構成の整った高品質文，30% を冗長だが意味は通る中品質文，20% を語彙や論理がやや破綻した低品質文として生成する．低品質文は，あえて接続詞の誤用や話題の飛躍を含めることで，実際の学生レポートに近いノイズを導入することを意図している．

P_3：**各戦略あたりのレポート数**　各執筆戦略から生成するレポート数であり，本研究では $P_3 = 20$ として 10 戦略から合計 200 本のレポートを生成した．

これらのパラメータの値は，予備実験により，TF–IDF や fastText ではクラスタが十分に分離せず，一方で SBERT や E5，OpenAI Embeddings では戦略ごとのクラスタが明確に現れる設定として選択した．

4. 埋め込みモデルと評価設計

本章では，比較対象とする埋め込みモデルと，類似度評価および可視化方法について述べる．

4.1. 比較対象とした埋め込みモデル

本研究では以下の 5 手法を比較対象とした．

TF–IDF　各レポートを形態素解析後に単語単位で分割し，scikit-learn の `TfidfVectorizer` を用いて TF–IDF ベクトルを作成した．これは BoW モデルとして広く用いられるベースラインである．

fastText　Bojanowski らによる fastText[4] の事前学習済み日本語モデルを用い，各単語の分散表現を取得した．文章ベクトルは，出現単語ベクトルの単純平均としている．fastText は文字 n グラムに基づく subword 情報を利用するため，未知語に対しても分散表現を推定できる利点を持つ．

SBERT　Sentence-BERT[6] は，BERT をシャムネットワーク構造で fine-tune した文埋め込みモデルである．本研究では多言語版の事前学習モデルを利用し，レポート全体を 1 ベクトルに写像した．

E5　Wang らによる E5[7] は，大規模なテキストペアデータセットを用いた弱教師あり対照学習により学習された汎用テキスト埋め込みモデルである．検索・クラスタリング・分類など多様なタスクで高い

性能を示している．

OpenAI Embeddings

OpenAI が提供する `text-embedding-3-large` モデルを用い，各レポートの埋め込みを取得した[13]．本モデルは多言語かつ長文への対応を特徴とし，検索・類似度計算・推薦などへの応用を想定して設計されている．

これらの手法は近年のテキスト埋め込み研究において広く利用されており，LLM 時代の embedding の性能比較や設計指針については Morris らによる包括的なサーベイが存在する[14]．

いずれの手法においても，レポート全文から 1 本の固定長ベクトルを得ることを前提とし，コサイン類似度によりレポート間の類似度を計算した．

4.2. 類似度評価指標

文書埋め込みモデルの性能を評価するため，本研究では ROC-AUC（Receiver Operating Characteristic - Area Under Curve），Average Precision（AP：平均適合率），PR 曲線，Mean Reciprocal Rank (MRR)，および Recall@k を用いる．これらの評価指標は，情報検索の分野で標準的に利用されるものであり[1]，ランキング性能や検索精度を多面的に評価できる．

4.2.1. 混同行列と基本指標（TP, FP, TN, FN）

文章類似度判別を含む 2 クラス分類問題では，分類器の出力を混同行列（confusion matrix）に整理することで分類性能を体系的に評価できる．混同行列は，実際のクラスと分類器の予測結果の対応を表す 2×2 の表であり，表 1 に示す 4 つの要素から構成される．

表 1: 混同行列

		予測	
		Positive	Negative
実際	Positive	TP	FN
	Negative	FP	TN

ここで，TP などの各要素は以下の値を表す．

- ***TP*** ：実際に Positive のものを Positive と正しく予測した数
- ***FP*** ：実際は Negative であるにもかかわらず，誤って Positive と予測した数（偽陽性）
- ***TN*** ：実際が Negative のものを Negative と正しく予測した数
- ***FN*** ：実際は Positive であるにもかかわらず，誤って Negative と予測した数（偽陰性）

これら 4 つの値を用いることで，分類器の性質を多面的に評価できる．

4.2.1.1. 具体例

たとえば本研究のように，文章ペアが「内容的に類似している」かどうかを判定する分類器を考える．実際の正例（Similar）が 20 件，負例（Dissimilar）が 80 件存在し，分類器が表 2 のように予測したとする：

表 2: 分類器の予測結果

		予測	
		Positive	Negative
実際	Positive	18	2
	Negative	10	70

このとき，

$$TP = 18,\ FN = 2,\ FP = 10,\ TN = 70$$

となる．

4.2.2. Recall（再現率）

Recall（再現率）は，実際に Positive である事例をどれだけ見逃さずに検出できたかを測る指標であり，次式で定義される．

$$\text{Recall} = \frac{TP}{TP + FN} \tag{1}$$

上記の例では，

$$\text{Recall} = \frac{18}{18 + 2} = 0.90$$

となり，実際に類似している文章ペアの 90% を正しく検出できていることを意味する．文章類似度分析のように，見逃し（False Negative）を避けたい場合に特に重要となる指標である．

4.2.3. Precision（適合率）

Precision（適合率）は，分類器が Positive と判断した事例のうち，どれだけが正しいかを測る指標であり，以下で与えられる．

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

例では，

$$\text{Precision} = \frac{18}{18 + 10} \approx 0.643$$

となり，分類器が「類似している」と判断した文章ペアの約 64% が，実際に類似していたことを示す．これは誤検出（False Positive）を抑える能力を表す指標であり，類似判定を厳密に行いたい場合に重要となる．

4.2.4. Precision と Recall の関係

一般に, Recall を高めると FP が増加し Precision が低下する傾向がある．一方，Precision を高めようとすると Positive 判定が慎重になり FN が増加しやすく，Recall が低下する．したがって両者はト

レードオフの関係にあり，タスクの目的に応じてどちらを重視するかを選択する必要がある．

本研究では，文章レポートの内容的類似性を評価するため，Positive（類似）・Negative（非類似）判定を用いた指標（Recall, Precision）に加え，ROC-AUC，Average Precision（AP），MRR，Recall@k といったランキング指標も併用し，文書埋め込みモデルの多面的な性能比較を行う．

4.2.5. ROC-AUC

ROC 曲線は，判定閾値を変化させたときに得られる縦軸の True Positive Rate（TPR）と横軸の False Positive Rate（FPR）の関係を可視化したものである．TPR と FPR は以下の式で定義される．

$$\mathrm{TPR} = \frac{TP}{TP + FN}, \quad \mathrm{FPR} = \frac{FP}{FP + TN}$$

ここで，TPR は正例クラスに対する再現率（Recall，式 (1)）と等価である．

そして，ROC-AUC は，ROC 曲線の下の面積である．正例ペアのスコアが負例ペアより高いほど ROC-AUC は 1 に近づき，完全識別であれば ROC-AUC=1.0，ランダム識別器の場合は 0.5 となる．したがって，ROC-AUC は閾値に依存しない分類性能指標として広く用いられる．

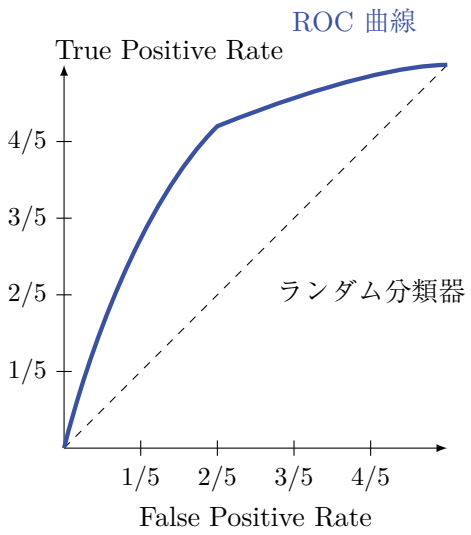


図 1: ROC 曲線の例（正例スコア＞負例スコアの場合）

4.2.5.1. ROC-AUC の例

正例スコアが

$$[0.90, 0.85, 0.80],$$

負例スコアが

$$[0.50, 0.40, 0.30]$$

であれば，どの閾値を選んでも 正例 > 負例 の関係が崩れないため，ROC-AUC = 1.0 になる．

逆に，両者の分布が重なっていれば ROC-AUC は 0.5 前後となり，区別能力の低さを示す．

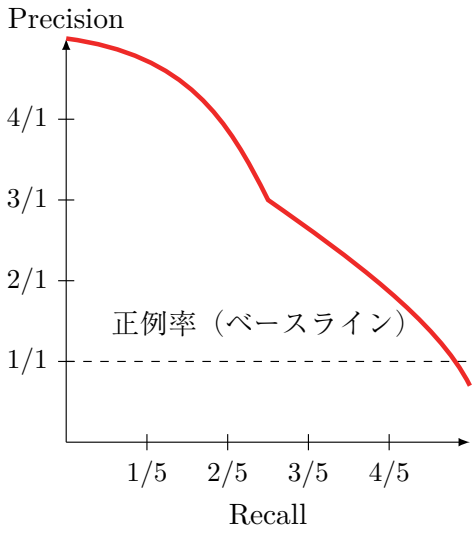


図 2: PR 曲線の例（上位に正例が集中するケース）

4.2.6. Average Precision (AP)

AP は Precision-Recall 曲線（PR 曲線）の下の面積（AUPRC）に等しく，類似度ランキングの上位に正例を多く含められるほど値が大きくなる．

AP は以下で定義される：

$$\mathrm{AP} = \sum_{k=1}^{n} (R_k - R_{k-1}) P_k,$$

ここで，P_k は Precision，R_k は Recall である．

4.2.6.1. AP の例

次のように類似度スコアで並べたとする：

順位	スコア	正例/負例
1	0.95	正例
2	0.90	正例
3	0.60	負例
4	0.55	正例
5	0.20	負例

正例は 3 件のため，Recall は

$$1/3,\ 2/3,\ 1$$

と増加する．Precision は

$$1,\ 1,\ 3/4$$

となり，これらから AP $\approx$ 0.92 が得られる．正例が上位に集中するほど AP が高くなる．

4.2.7. Mean Reciprocal Rank (MRR)

MRR は「最初に正例が出現する順位」に基づく評価指標である．

類似度ランキングを

$$d_{i,(1)}, d_{i,(2)}, d_{i,(3)}, \ldots$$

とするとき，最初の正例の順位を

$$r_i = \min\{k \mid y_{d_{i,(k)}} = y_i\}$$

とし，逆数

$$\mathrm{RR}_i = \frac{1}{r_i}$$

を計算する．これを全クエリで平均したものが MRR である．

4.2.7.1. MRR の例

順位	ラベル
1	3
2	2
3	7
4	5（正例）
5	1

クエリのラベルが 5 なら最初の正例は 4 位であるため，

$$\mathrm{RR} = \frac{1}{4} = 0.25.$$

1 位に正例がくれば RR=1 であり，MRR は「どれだけ上位に正例を置けたか」を測る．

4.2.8. Recall@k の定義

類似度にもとづくランキング評価では，あるクエリ文書 i に対して類似候補集合を類似度の高い順に並べ，上位 k 件に同一クラス（あるいは正例）文書が含まれるかどうかを測定する Recall@k が広く用いられる [1)]．

いま，文書集合 $\mathcal{D} = \{1, \ldots, N\}$ のうち，文書 i と同じクラスに属する正例集合を $\mathcal{P}(i)$ とする．ここで，$i \notin \mathcal{P}(i)$ である．また，文書 i に対して計算した類似度スコアを降順に並べたランキング列を

$$r_i(1), r_i(2), \ldots, r_i(N)$$

と書く．ただし $r_i(1)$ は文書 i に最も類似した文書を表す．このとき Recall@$k(i)$ は

$$\mathrm{Recall@}k(i) = \begin{cases} 1 & \text{if } \exists j \le k \text{ s.t. } r_i(j) \in \mathcal{P}(i), \\ 0 & \text{otherwise} \end{cases}$$

で与えられる．

すなわち，文書 i のランキング上位 k 件の中に「同じクラスの文書が 1 件でも存在するか」を判定する指標である．全データ集合での Recall@k は，すべての文書について平均をとった

$$\mathrm{Recall@}k = \frac{1}{N} \sum_{i=1}^{N} \mathrm{Recall@}k(i)$$

で定義される．

直感的には，ユーザが求める「関連文書（正例）」をランキングの上位 k 件以内で取り出せている割合を意味し，情報検索やレコメンダシステム，文書類似度評価における基本的な性能指標である．

4.2.8.1. Recall@k の例

先ほどの例では，

$$\text{Recall@1} = 0, \quad \text{Recall@5} = 1.$$

上位 1 件では正例がないが，上位 5 件には正例が存在するためである．

Recall@1 $\leq$ Recall@5 $\leq$ Recall@10 となり，k を大きくするほど Recall は増加する．

4.2.9. 指標の位置づけ

以上 4 つの指標には次の特徴がある：

- ROC-AUC：スコア分布の分離度を評価（閾値に依存しない）
- AP：ランキング上位の正例密度を強く評価
- MRR：最初の正例の位置（序盤の正解率）を評価
- Recall@k：特定の上位幅での正解率を評価

本研究では，「同一戦略レポートの検索性能」と「類似度ランキングの品質」を多角的に評価するため，これらすべてを採用した．

4.3. 次元削減と可視化

埋め込み空間におけるクラスタ構造を直感的に理解するため，各モデルのベクトルに対して次元削減を行い，2 次元散布図として可視化した．可視化には非線形手法である t-SNE[8] と UMAP[9] を用いた．

t-SNE は局所的な隣接関係の保持に優れる一方で，大域的な構造が歪むことが知られている[8]．UMAP は多様体学習に基づく次元削減手法であり，t-SNE と同程度の可視化品質を保ちつつ，大域構造の保持と計算効率に優れると報告されている[9]．本研究では，両者の長所・短所を踏まえつつ，クラスタの分離度合いと形状を比較した．

5. 実験結果

本章では，$K = 10$ 戦略，$P_3 = 20$ により生成した合計 200 本の人工レポートを対象とした実験結果を示す．

5.1. 数値評価

表 3 に，各埋め込みモデルの ROC-AUC，Average Precision，MRR，Recall@1/5/10 の結果を示す．ここで TF–IDF および fastText は BoW/単語平均型のベースラインであり，SBERT，E5，OpenAI Embeddings は文レベル埋め込みモデルである．

表 3 から，TF–IDF は ROC-AUC が約 0.59，fastText でも約 0.71 に留まり，ランキング指標も 3 手法に比べて大きく劣ることが分かる．特に TF–IDF の

表 3: 人工レポートデータに対する各埋め込みモデルの類似度評価結果

Method	ROC-AUC	AP	MRR	R@1	R@5	R@10
TF–IDF	0.5938	0.1469	0.5189	0.3833	0.6792	0.8167
fastText	0.7112	0.2077	0.7204	0.6083	0.8750	0.9500
SBERT	0.9998	0.9977	1.0000	1.0000	1.0000	1.0000
E5	0.9947	0.9593	1.0000	1.0000	1.0000	1.0000
OpenAI Emb.	0.9998	0.9982	1.0000	1.0000	1.0000	1.0000

Recall@1 は 0.38 であり，クエリと同一戦略のレポートを最上位に推薦できる割合は 4 割未満である．

一方，SBERT，E5，OpenAI Embeddings の 3 手法は ROC-AUC, AP, MRR, Recall@1/5/10 のいずれにおいても 1.0 近い値を示し，本実験条件においてはほぼ完全に執筆戦略を識別できている．特に SBERT と OpenAI Embeddings は，すべてのランキング指標で 1.0 を記録し，任意のレポートをクエリとしたときに必ず同一戦略のレポートを最上位に位置付けている．

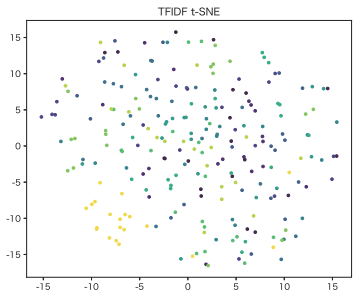


図 3: TF–IDF の t-SNE

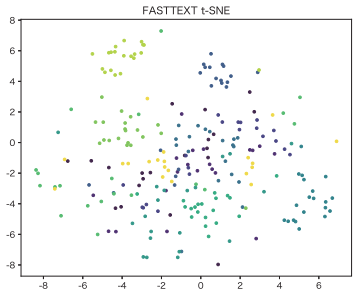


図 4: fastText の t-SNE

5.2. 可視化結果

図 3 〜 12 に，TF–IDF，fastText, E5, SBERT, OpenAI Embeddings の t-SNE および UMAP による 2 次元可視化結果の例をそれぞれ示す．なお，プロットの色は執筆戦略ラベルを表す．

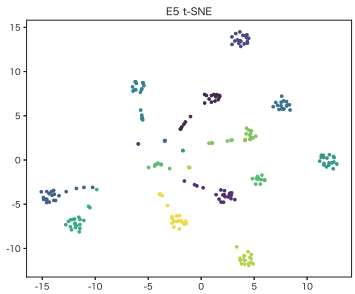


図 5: E5 の t-SNE

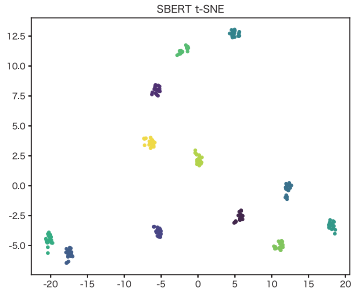


図 6: SBERT の t-SNE

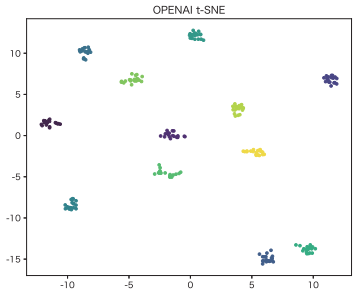


図 7: OpenAI Embeddings の t-SNE

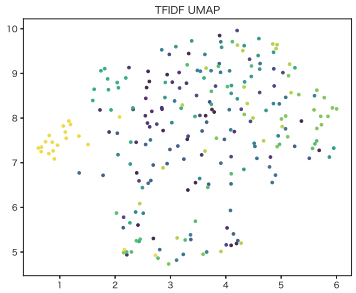


図 8: TF–IDF の UMAP

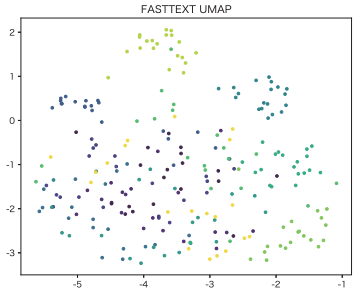


図 9: fastText の UMAP

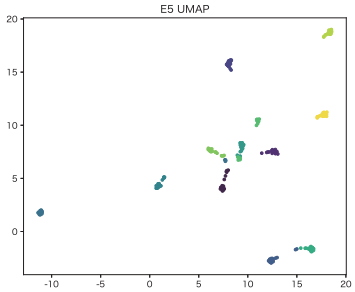


図 10: E5 の UMAP

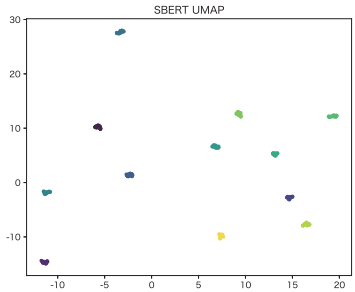


図 11: SBERT の UMAP

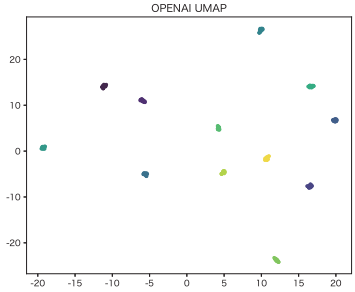


図 12: OpenAI Embeddings の UMAP

SBERT，E5，OpenAI Embeddings のいずれにおいても，t-SNE・UMAP ともに 10 個のクラスタがほぼ完全に分離している．同一戦略に属する点は高密度の塊を形成し，異なる戦略のクラスタとの重なりはほとんど見られない．

これに対して，fastText と TF–IDF では挙動が大きく異なる．fastText の t-

SNE/UMAP では，一部の戦略に対応する薄いクラスタが見られるものの，多くの点が混在しており，明確な 10 クラスタ構造は確認できない．

TF–IDF の可視化では，特定の戦略に対応する小さなクラスタが現れる場合もあるが，全体としては戦略ラベルがほぼランダムに散在している．これは，本研究の人工レポートデータにおいて全戦略で共通の語彙群を意図的に混在させている設計と一致している．

6. 考察

本章では，上記の結果に基づき，BoW/単語平均型埋め込みと文レベル埋め込みモデルの違いについて考察する．

6.1. BoW/単語平均型埋め込みの限界

TF–IDF は単語の出現頻度に基づくベクトル表現であり，単語の順序や長距離の依存関係を考慮しない．本研究の人工レポートデータでは，すべてのレポートが共通導入・共通結論を持ち，さらに本論でも共通語彙が意図的に混在するように設計されている．このため，戦略の違いは主に「どの観点をどの順番でどの程度の深さで論じるか」という談話構造の差として現れる．しかし TF–IDF では，このような構造的な違いが十分に表現されず，単語分布の些細な差に基づいて類似度が決まってしまう．

fastText は subword 情報を利用して単語分散表現を獲得する点で TF–IDF よりも強力であるが，文章レベルの表現としては単語ベクトルの平均に過ぎない．長文レポートでは多様な話題が混在するため，平均操作によって各話題に対応するベクトルが相殺され，結果として文書間の差が縮小してしまう．本研究の結果でも，fastText は TF–IDF よりは高い性能を示したものの，説明したいクラスタ構造を完全に再現するには至らなかった．

6.2. 文レベル埋め込みモデルの優位性

SBERT は，類似文判別タスクに基づき文ペアを近傍に写像するように学習されたモデルである [6]．E5 は大規模な検索ログ等から構成された CCPairs コーパスを用い，クエリと関連文書の埋め込みが近くなるように弱教師あり対照学習を行っている [7]．これらのモデルは，単語レベルではなく文レベルの意味関係を直接的に最適化しているため，「教育的観点からの批判的検討」「技術的詳細の提示」など，執筆戦略を特徴づける高次の意味構造を捉えることができる．

OpenAI Embeddings は，より大規模なデータとモデル容量に基づき学習されており，多言語・長文に対しても安定した意味表現を提供するよう設計されている．本研究の実験では，SBERT および

E5 とほぼ同等，あるいはそれ以上の性能を示し，人工レポートの執筆戦略を完全に復元することが確認された．

これらの結果は，レポート自動分類や類似レポート推薦といった教育データ分析において，BoW/単語平均型の従来ベースラインに比べて，文レベル埋め込みモデルを採用することの実用上の利点を示している．特に，本研究のように語彙レベルでクラス差が小さい設定では，両者の差が顕著に現れる．

6.3. 人工レポートデータを用いた評価の意義と限界

本研究の特徴は，実際の学生レポートではなく，生成 AI による人工レポートを用いて埋め込みモデルを比較した点にある．人工レポートデータを用いることで，下記のような利点が得られる．

(1) 戦略ラベルを厳密に制御できる，
(2) データ数やノイズレベルをパラメータとして調整できる，
(3) 再現性の高い実験が可能になる，

一方で，人工レポートはあくまで生成モデルが作り出した文章であり，実際の学生レポートに見られる誤字脱字や独特の表現，知識の偏りなどを完全には再現していないという限界がある．また，本研究で用いた生成プロンプトやモデルは，埋め込みモデルとは独立であるものの，同じ大規模言語モデルの系統に属しているため，ある程度のバイアスが共有されている可能性も否定できない．

したがって，人工レポートデータに基づく評価結果をそのまま実践現場に適用することは慎重であるべきだが，埋め込みモデルの挙動を比較するためのテストベッドとしては有用であると考えられる．今後は，人工レポートと実際の学生レポートを組み合わせたハイブリッドな評価環境の構築が課題である．

7. まとめ

本研究では，大テーマが共通で執筆方針が異なるレポート集合を想定し，生成 AI を用いて人工レポートデータを構築した上で，5 種類の埋め込みモデル (TF–IDF, fastText, SBERT, E5, OpenAI Embeddings) の類似度計算および可視化の挙動を比較した．

実験の結果，BoW/単語平均型の TF–IDF と fastText は，執筆戦略に対応するクラスタ構造を十分に再現できず，ROC-AUC やランキング指標でも性能が限定的であることが示された．一方で，SBERT，E5，OpenAI Embeddings の 3 手法は，人工レポートの戦略ラベルをほぼ完全に区別し，t-SNE・UMAP による 2 次元可視化でも明確な 10 クラスタ構造を形成した．

これらの結果から，大学レポートの自動分類・類似レポート検索といった教育応用においては，BoW/単語平均型の従来手法ではなく，埋め込みベクトルモデルを用いることが有効であると結論づけられる．今後の課題として，実際の学生レポートデータへの適用と，教員による評価結果との比較，および人工レポートデータ生成プロセスそのものの検証と改良が挙げられる．

謝辞

本稿を執筆するにあたり本学国際コミュニケーション学部教授 梅垣敦紀先生に作成していただいた upLaTeX のクラスファイル[15]を利用した．ここに感謝の意を表す．

参考文献

1） Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. Introduction to Information Retrieval. https://nlp.stanford.edu/IR-book/, 2008.

2） Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. https://arxiv.org/abs/1301.3781, 2013.

3） Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*, 2013.

4） Bojanowski Piotr, Grave Edouard, Joulin Armand, and Mikolov Tomas. Enriching Word Vectors with Subword Information. https://cir.nii.ac.jp/crid/1363388843924591872, 12 2017.

5） Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All You Need. https://arxiv.org/abs/1706.03762, 2017.

6） Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. https://arxiv.org/abs/1908.10084, 2019.

7） Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text Embeddings by Weakly-Supervised Contrastive Pre-training. https://arxiv.org/abs/2212.03533, 2024.

8） Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. http://jmlr.org/papers/v9/vandermaaten08a.html, 2008.

9） Leland McInnes, John Healy, and James Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. https://arxiv.org/abs/1802.03426, 2020.

10） OpenAI. GPT-4 Technical Report. https://arxiv.org/abs/2303.08774, 2023.

11） Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, et al. Language Models are Few-Shot Learners. https://arxiv.org/abs/2005.14165, 2020.

12） Mercy Ranjit, Gopinath Ganapathy, Ranjit Manuel, and Tanuja Ganu. Retrieval Augmented Report Generation using OpenAI GPT models. https://arxiv.org/abs/2305.03660, 2023.

13） OpenAI. OpenAI Text Embeddings. https://platform.openai.com/docs/guides/embeddings, 2024.

14） John X. Morris, Volodymyr Kuleshov, Vitaly Shmatikov, and Alexander M. Rush. When Text Embedding Meets Large Language Model. https://arxiv.org/abs/2412.09165, 2025.

15） 梅垣敦紀. 愛知大学情報メディアセンター紀要「COM」への upLaTeX での投稿. https://saturn.aichi-u.ac.jp/img/center/pdf/com49-2.pdf, 2025.