Recommendation System for Biomedical Literature

Xin Lan
School of Public Health, Yale University
x.lan@yale.edu

Luoyi Tian
Graduate School of Arts and Sciences, Yale University
luoyi.tian@yale.edu

Yiran Liu
Graduate School of Arts and Sciences, Yale University
yiran.liu@yale.edu

Ruoxi Teng
Graduate School of Arts and Sciences, Yale University
ruoxi.teng@yale.edu

Abstract

We present a semantic recommendation system for biomedical literature using the RELISH v1 dataset of 180 000 query–candidate pairs. We compare six encoder architectures: (1) MPNet-based SBERT (baseline), (2) SBERT fine-tuned with triplet loss, (3) BART encoder, (4) feed-forward autoencoder, (5) variational autoencoder, and (6) domain-specialized BioBERT. Retrieval is performed via FAISS approximate nearest-neighbor search, and performance is measured by MAP@5, MRR@5, and NDCG@5. The SBERT baseline achieves the highest perfomance, while full fine-tuning boosts NDCG@5 at the expense of overall precision—highlighting a trade-off between top-hit accuracy and broader ranking consistency. Other variants (BioBERT, AE, VAE, BART) underperform, suggesting that contrastively trained sentence transformers are particularly well-suited for capturing biomedical semantics under data constraints. Our results demonstrate that lightweight, contrastive sentence embeddings can effectively complement keyword-based retrieval, offering both high precision and semantic richness. However, limited dataset size and domain shift constrain fine-tuning gains. Future work should explore domain-adaptive pretraining to further enhance recommendation fidelity and generalizability.

1 Introduction and Motivation

As biomedical research papers multiply, relying solely on keywords is insufficient—simple searches miss synonyms, paraphrases, and technical jargon. By embedding titles and abstracts into "meaning vectors," recommendation systems can capture context and meaning, suggesting truly related articles.

2 Background and Related Work

Existing studies in biomedical literature access hinge on two complementary strategies: retrieval—typically realized by keyword-based systems that build inverted indices to screen arti-

Preprint. Under review.

cles—and recommendation—which automatically suggests similar papers based on user profiles, click history, or content similarity[1,2]. Keyword retrievers remain ubiquitous in search engines and databases, whereas recommenders fill gaps by interpreting user interactions—for example, PubMed's "Similar Articles" feature logs clicks on a given paper as implicit relevance feedback, then surfaces semantically related work[1,2]. Recommendation algorithms range from graph-based ranking (PMRA-link, which applies PageRank/HITS to content-similarity networks[3]; bibliographic coupling in PBC, which uses shared citations[4]) to lightweight ML classifiers (MScanner[5], MedlineRanker[6]) and semi-supervised content filters like PURE[7] that let experts seed candidate sets. More advanced learning-to-rank approaches (Crow-rank[8]) leverage crowdsourced judgments to optimize rank orders, while integrated platforms (BioReader[9], LitSuggest[10]) layer iterative user feedback and workflow tools for real-time personalization. The latest shift to transformer-based embeddings (BioBERT[11], SBERT[12]) delivers deep, context-aware representations that capture paraphrases and domain-specific jargon, enabling next-generation biomedical recommendation engines to complement traditional retrieval with highly precise, semantically rich suggestions.

3 Methods / Model

3.1 Dataset and Preprocessing

RELISH v1 is a crowd-sourced biomedical "similar article" dataset covering around 180 k articles across domains. We use the RELISH v1 corpus[13] as training data. From its JSON, we extract query PMIDs and their "relevant," "partial," and "irrelevant" candidates. We batch-fetch metadata via NCBI E-Utilities (200 PMIDs/request), extracting title, abstract, and keywords. Text is cleaned by collapsing whitespace, removing non-alphanumerics (preserving "." and ","), and lowercasing keywords. We concatenate into a `full_text` field, then split queries 80%/10%/10% (train/val/test). Finally, we build `InputExample` pairs $\big([\text{query}, \text{candidate}], y\big)$ with $y \in \{2, 1, 0\}$ for relevant, partial, and irrelevant. See Appendix A for details.

3.2 Models

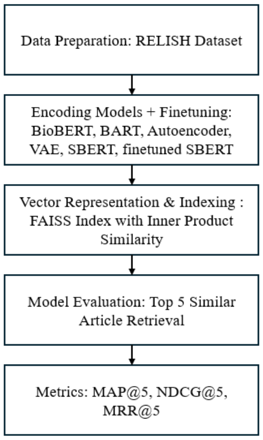
Data Preparation: RELISH Dataset
Encoding Models + Finetuning: BioBERT, BART, Autoencoder, VAE, SBERT, finetuned SBERT
Vector Representation & Indexing : FAISS Index with Inner Product Similarity
Model Evaluation: Top 5 Similar Article Retrieval
Metrics: MAP@5, NDCG@5, MRR@5

Figure 1: Model development process for biomedical article recommendation.

We conducted model development following the process shown in Figure 1. We evaluated three sentence-transformer models and two autoencoder based model for generating our article embeddings:

- `all-mpnet-base-v2`: a 768-dimensional MPNet-based SBERT encoder providing state-of-the-art performance on diverse downstream benchmarks.
- **Seq2Seq encoder (BART)**: a transformer encoder–decoder (`facebook/bart-base`) where we use the encoder's first-token output as a fixed-length embedding.
- **Autoencoder (AE)**: a lightweight feed-forward text autoencoder built on top of mean-pooled BERT embeddings (768→256→768) trained to minimize reconstruction error.

- **Variational Autoencoder (VAE)**: similar architecture to the AE but with a stochastic latent layer (μ,σ) to encourage smooth, robust embeddings under limited data.
- **BioBERT**: the domain-specialized `dmis-lab/biobert-base-cased-v1.1` model, using mean-pooled token embeddings for representation.

3.3 Fine-tuning Strategies

Standard fine-tuning on `all-mpnet-base-v2`: update all parameters with triplet loss (margin=0.2), 3 epochs, batch=16, LR=1e-5, 10% linear warmup on `all-mpnet-base-v2`

3.4 Evaluation Protocol

We train on the 80% split, select the checkpoint maximizing MAP@5 on validation, and test by encoding each query to retrieve top-k candidates ($k \in \{5, 10, 15\}$) via FAISS `IndexFlatIP` over ℓ_2-normalized vectors. We quantify retrieval quality using three established metrics:

- **Mean Average Precision (MAP)** computes the mean of per-query average precisions, capturing both precision and recall across the ranked list. For a query q with R_q total relevant documents and precision $P(k)$ at rank k,

$$\mathrm{AP}(q) \;=\; \frac{1}{R_q}\sum_{k=1}^{n} P(k)\,\mathrm{rel}(k), \quad \mathrm{MAP} \;=\; \frac{1}{|Q|}\sum_{q=1}^{|Q|}\mathrm{AP}(q).$$

- **Mean Reciprocal Rank (MRR)** averages the reciprocal rank of the first relevant result across queries, emphasizing early correct hits. If rank_i is the position of the first relevant document for query i, then

$$\mathrm{MRR} \;=\; \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{\mathrm{rank}_i}.$$

- **Normalized Discounted Cumulative Gain (NDCG)** handles graded relevance by discounting gains logarithmically by position and normalizing by the ideal ranking. For top p positions:

$$\mathrm{DCG}_p \;=\; \sum_{i=1}^{p}\frac{2^{\mathrm{rel}_i}-1}{\log_2(i+1)}, \quad \mathrm{NDCG}_p \;=\; \frac{\mathrm{DCG}_p}{\mathrm{IDCG}_p}.$$

We compute MAP@5, MRR@5, and NDCG@5 over the test split.

3.5 Computing Infrastructure

All training and inference were run in Google Colab on a single NVIDIA Tesla T4 and A100 GPU with CUDA 11.2. We used Ubuntu 20.04, Python 3.8, PyTorch 1.11, HuggingFace Transformers 4.21, Sentence-Transformers 2.2.0, and FAISS 1.7.1, in 32-bit float mode.

3.6 Hyperparameters & Finetuning

- Batch size: 16
- Learning rate: 1×10^{-5}
- Triplet margin: 0.2
- Epochs: 3
- Warmup: 10% linear

4 Results

Table 1 shows test performance for all variants.

Table 1: Comprehensive performance on RELISH (test split).

Model Variant	MAP@5	MRR@5	NDCG@5
`all-mpnet-base-v2 (baseline)`	77.23%	93.88%	74.62%
`fine-tuned all-mpnet-base-v2`	75.47%	90.97%	78.37%
`BioBERT`	56.80%	81.86%	55.50%
`Variational Autoencoder`	38.60%	65.92%	38.93%
`BART`	16.74%	37.62%	18.78%
`Autoencoder`	27.28%	58.46%	30.76%

Key observations.

- Among baseline models, `all-mpnet-base-v2 (baseline)` leads in performance.
- *Full fine-tuning:* boosts NDCG@5 but lowers MAP@5/MRR, signaling overfitting.

5 Discussion

5.1 Negative Results & Analysis

When we fully fine-tuned all-mpnet-base-v2 on the RELISH dataset, we noticed that NDCG@5 improved, but MAP@5 and MRR@5 dropped. This shows that while the model improved ranking the most relevant article in the top position, it struggled to rank all relevant articles consistently in the top 5. One possible reason is that fine-tuning can help the model adapt to dataset-specific patterns, which boosts performance on a few top results. However, this also risks overfitting the embedding space too much and hurting its general ability to rank papers correctly overall. Since the pre-trained models are already trained on larger and diverse datasets, too much fine-tuning might actually undo some of that general usefulness.

5.2 Model Comparison

SBERT significantly outperforms other models in this biomedical article recommendation task primarily due to its architecture and training objective, which are specifically designed for capturing sentence-level semantic similarity. Unlike models like BioBERT and BART, which are primarily pretrained for token-level tasks, SBERT fine-tunes BERT using a Siamese or triplet network structure with contrastive loss. This enables it to produce dense vector embeddings that directly reflect sentence-level meaning, making it highly effective for tasks like semantic search and similarity ranking. Additionally, SBERT embeddings are optimized for inner product comparisons—aligning well with FAISS-based nearest neighbor search used in your system. This explains why even the base SBERT model surpasses BioBERT, and why fine-tuning brings only marginal additional gains: SBERT already encodes general semantic similarity very effectively out of the box.

5.3 Domain-Specific Challenges

RELISH includes a wide variety of biomedical fields, many of which use different terminologies, abbreviations, and writing styles. This may make it difficult for models to generalize. Also, the SBERT models we used are not biomedical domain-specific models, so they may lack the specialized vocabulary and contextual understanding needed to fully capture the nuances of biomedical literature.

6 Limitations & Future Work

- **Dataset bias:** RELISH over-represents omics/genomics; performance may vary in under-represented subdomains.
- **Limited dataset size:** RELISH's small corpus constrains fine-tuning effectiveness.
- **No statistical tests:** we did not report confidence intervals or run multiple seeds.
- **Compute constraints:** only a single GPU was used; larger models or multi-GPU training were not explored.

- **Future directions:** incorporate domain-adaptive pretraining, hard negative mining, and evaluate cross-corpus generalization.

References

[1] Fiorini, N., Leaman, R., Lipman, D.J. & Lu, Z. (2018) How user intelligence is improving PubMed. *Nat. Biotechnol.* **36**(10):937–945.

[2] Tran, N., Alves, P., Ma, S. & Krauthammer, M. (2009) Enriching PubMed related article search with sentence level co-citations. *AMIA Annu. Symp. Proc.*:650–654.

[3] Lin, J. (2008) PageRank without hyperlinks: Reranking with PubMed related article networks for biomedical text retrieval. *BMC Bioinformatics* **9**:1–12.

[4] Liu, R.L. (2015) Passage-based bibliographic coupling: An inter-article similarity measure for biomedical articles. *PLoS ONE* **10**(10):1–22.

[5] Poulter, G.L. et al. (2008) MScanner: A classifier for retrieving Medline citations. *BMC Bioinformatics* **9**:1–12.

[6] Fontaine, J.F. et al. (2009) MedlineRanker: Flexible ranking of biomedical literature. *Nucleic Acids Res.* **37**:141–146.

[7] Yoneya, T. & Mamitsuka, H. (2007) PURE: a PubMed article recommendation system based on content-based filtering. *Genome Informatics* **18**:267–276.

[8] Lingeman, J.M. & Yu, H. (2016) Learning to Rank Scientific Documents from the Crowd. *arXiv preprint arXiv:1611.01400.*

[9] Simon, C. et al. (2019) BioReader: a text mining tool for performing classification of biomedical literature. *BMC Bioinformatics* **19**(Suppl. 13):165–170.

[10] Allot, A. et al. (2021) LitSuggest: a web-based system for literature recommendation and curation using machine learning. *Nucleic Acids Res.* **49**(W1):W352–W358.

[11] Lee, J. et al. (2020) BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **36**:1234–1240.

[12] Reimers, N. & Gurevych, I. (2019) Sentence-BERT: Sentence Embeddings using Siamese BERT-networks. *Proc. EMNLP-IJCNLP*:3982–3992.

[13] Zhang, L. et al. (2022) A comparative evaluation of biomedical similar article recommendation. *J. Biomed. Inform.*

A Appendix: Data Preprocessing Details

- Remove special characters except "." and ",".
- Collapse multiple whitespace.
- Concatenate title, abstract, and keywords.
- Drop entries with missing abstracts or titles.

B Appendix: Group Member Contribution

Each member of our group contributes equally to the project and was involved in all parts of the report and the presentation. The specific contribution are listed as follows: Yiran Liu: Autoencoder model, introduction in presentation; Xin Lan: SBERT model with finetuning, conclusion in presentation; Ruoxi Teng: Variational Autoencoder Model, results in presentation; Luoyi Tian: BART and BioBERT model, methods in presentation;

A Video link

`https://yale.zoom.us/rec/share/7rLO-Ptyk6b-Nil2HUyEhu_`
`gh8bmmIpubjlyMdQBllB4QnljDp-PbGWRvvQwubtl.WGpkvq565Vc7J5rx`

A Appendix: Code Repository

https://github.com/Ruoxi-Teng/CPSC-452-Deep-Learning-Applications