Enhancing Content-Based Recommender System with Modern Sentence Embedding Methods

Võ Hoàng An, Trần Thị Mỹ Duyên, Nguyễn Văn Kiệt
University of Information Technology, Ho Chi Minh City, Vietnam
Vietnam National University, Ho Chi Minh City, Vietnam
{21520555, 21522017}@gm.uit.edu.vn
{kietnv}@uit.edu.vn

Abstract

In today's digital age, recommender systems plays a crucial role in improving user experiences. However, recommending content for new users and products with limited information, known as the "cold start" problem remains a challenge. This study explores content-based recommender systems as a solution and experiments with different embedding methods, including baseline ones like TFIDF and Word2Vec, and advanced transformer-based models like Sentence-BERT (SBERT), Sentence-T5 (ST5), and Sentence-GPT (SGPT). Using Google's renowned Goodreads dataset, we aim to assess and compare these methods' effectiveness in enhancing recommender systems and performing topic model tasks. Transformer-based models, especially SBERT, ST5, and SGPT, outperform our baseline in both recommendation and topic model tasks. Notably, SBERT achieves the highest recall@50 score of 0.4574 in the recommendation task, ST5 achieves the highest C_v score of 0.6422, and SGPT achieves the highest NPMI score of 0.1487 in topic quality evaluation. These results demonstrate that modern sentence embedding methods can improve content-based recommender system performance and generate higher-quality topics.

1 Introduction

In the rapidly advancing era of technology, internet users are confronted with an overwhelming amount of information, making the selection of relevant information a challenging and decision-diminishing process. To address this issue, considerable efforts have been made by researchers to develop recommendation systems capable of tailoring information based on user preferences, ranging from video recommendations (Covington et al., 2016), music choices (Patra et al., 2017), movie reviews (Kumar et al., 2015) to product recommendations.

Within the realm of recommendation systems, algorithms can be broadly categorized into content-based and collaborative filtering methods. Content-based algorithms create user and item profiles, often in vector form, while collaborative filtering analyzes past user behavior to identify similarities in preferences. The integration of Deep Learning techniques has been employed to enhance the performance of these recommendation models.

However, building effective recommendation systems faces challenges, especially when users struggle to articulate their preferences. This difficulty is exacerbated during the cold start phase, where insufficient user behavior data hampers accurate suggestions. Content-based recommender systems, addressing the cold start problem, suggest products based on the similarity between personal preferences (inferred from shopping history and reviews) and product features.

Our research focuses on improving the efficiency of recommendation systems, particularly in terms of reducing computational overhead. We employ sentence embedding methods, such as TFIDF, Word2Vec, Sentence-Bert (SBERT), Sentence-T5 (ST5), and Sentence-GPT (SGPT), to convert text into representative vectors. This facilitates the capture of product features and better understanding of user needs, especially in the cold start phase. Through rigorous experiments and visual comparisons of various methods, our goal is to identify optimal solutions to enhance the quality of content-based recommendations and evaluate their effectiveness in topic modeling. This aims to optimize user experiences and alleviate computational burdens in deploying recommendation systems.

The figure 1 illustrates how our content-based recommender system is organized. When a new user visits our recommender system and reads some books, these books are processed through an embedding module to create a user representation. This representation is then compared for the similarity with books already embedded in our database. Our system suggests the top K recommended books for this new user based on this comparison. Our contributions can be summarized as follows:

1. Conducting experiments on both recommendation and topic model tasks using sentence embedding methods.

2. Analyzing results and highlighting certain limitations of embedding methods, offering insights for future research endeavors.

The structure of our work organized as follows: Section 2 provides an overview of related work on recommendation tasks and embedding techniques. Section 3 provides information about the dataset utilized in our experiments, including details about how we processed it and presenting statistics related to this dataset. Section 4 outlines the experiment setup and result analysis. Finally, Section 5 discusses the conclusion, future works, and limitations.x

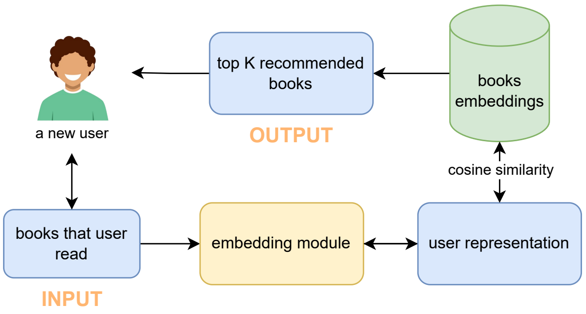


Figure 1: Framework workflow

2 Related Works

In contemporary times, the reliance on recommendations from other users for discovering suitable products and content has become a prevailing trend across various domains such as movies, images, and books (Resnick and Varian, 1997). Recommender systems (RS) are increasingly crucial, evidenced by numerous studies in this field, including the work of (Chen et al., 2019), who proposed a top-K recommender system on YouTube, and (Xu and Hu, 2023), who enhanced product recommendations in e-commerce by utilizing pre-trained language models and fine-tuning.

The Collaborative Filtering approach (CF) stands out as the most prevalent technique in RS (Yang et al., 2016), generating recommendations for users based on the similarities between their preferences and interests with those of similar users in the past. However, a significant challenge is the lack of information from users and products, referred to as the cold start problem (Hasan and Khatwal, 2022). Various approaches address this issue, such as (Nguyen et al., 2014) leverage user ratings as contextual information and propose the LinUCB algorithm. Another study is the cross-domain recommendation by (Bi et al., 2020), which transfers information from a source domain to a target domain to alleviate information sparsity. This method combines cross-domain mechanisms and heterogeneous information networks to provide personalized recommendations for new users in the insurance domain.

Regarding content-based recommender system in the context of the cold start problem, there are studies employing techniques such as Topic Models in Content-Based News Recommender Systems. (Mooney and Roy, 2000) describe a book recommender system based on information extraction and a machine-learning algorithm for text categorization. Additionally, (Salmi et al., 2021) presents a Content-Based Recommender Support System for Counselors in a Suicide Prevention Chat Helpline: Design and Evaluation Study.

A majority of prior research has leveraged word embedding techniques to embed item metadata by averaging individual words to form sentence embeddings. Several noteworthy studies have adopted word embeddings, such as the work by (Mediani et al., 2023) which employs word embeddings for enhancing educational content recommendations. Additionally, (Birunda and Devi, 2021) explored the application of contextualized word embeddings. But differing from word embeddings, sentence embeddings can capture the entire context of a text, providing better context preservation. (Wang and Kuo, 2020) introduced a two-sentence embeddings approach with both non-parameterized and parameterized models. Many studies utilize BERT, as seen in the work of (Cygan, 2021) and (Juarto and Girsang, 2021), to create sentence embeddings. (Mendes de Melo et al., 2022) proposed a content-based recommender system to support COVID-19, utilizing sentence embeddings to ensure the quality of recommendations in chat conversations.

3 Dataset

Our research focuses on analyzing Google's Goodreads dataset, particularly the "comic-graphic" category, to assess and improve recommender system performance. This dataset provides a rich variety of information about books, users, and reviews, offering a unique perspective for our study.

To enhance the recommender system, we employed preprocessing techniques such as lowercase conversion, tokenization, stopwords removal, whitespace trimming, and stemming/lemmatization.

After preprocessing, we obtained two datasets: MetaBooks and UsersHistory.

MetaBooks: includes book metadata like titles, authors, and descriptions. We also generated a text_features column by concatenating title, description, and author name. The statistic of MetaBooks is shown in table 1. The length distribution of text_features column (see figure 2) is right-skewed, with a long tail extending towards longer documents. The text_features has a Zipfian (Newman, 2005), (see figure 3) distribution of word frequencies, meaning that there are a few words that appear very frequently and many more words that appear only a few times.

UsersHistory: contains user-book interactions, including user_id, book_id, and ratings ranging from 1 to 5. This dataset, along with MetaBooks, provides a solid foundation for evaluating methods and addressing the recommender system problem.

MetaBooks (preprocessed)	
Total documents	28565
Vocabulary size	73526
Max words of document	1071
Min words of document	4
Average number of words per document	98

Table 1: Statistics of MetaBooks

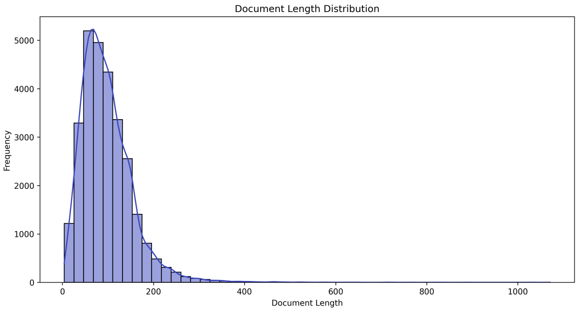


Figure 2: Documents Length Distribution

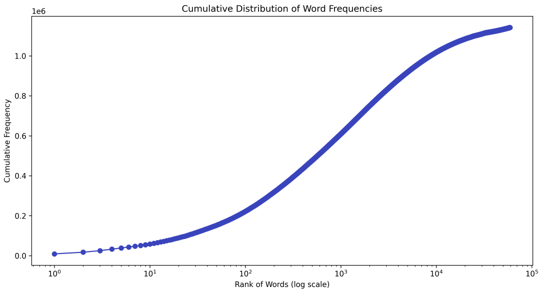


Figure 3: Cumulative Distribution of Word Frequencies

4 Experiments and Results

4.1 Methods

This study compares classical embedding models, such as TFIDF and Word2Vec serve as our baseline, with modern transformer-based architecture models. Further details provided below.

TFIDF (Term Frequency-Inverse Document Frequency): is a term weighting scheme used in information retrieval. It calculates a term's importance in a document by considering its frequency and rarity, helping identify the key terms for document retrieval. It assigns higher weights to terms frequent in a document but rare in the collection. A survey (Beel et al., 2015) showed that 83 % of text-based recommender systems in digital libraries used TFIDF.

Word2Vec: Word2Vec, introduced by (Mikolov et al., 2013), has become a widely recognized and effective technique in Natural Language Processing (NLP). Its simplicity and robust performance in various NLP tasks have solidified its role as a fundamental method for transforming textual features into meaningful embeddings. This approach, rooted in the idea of learning distributed representations for words, has gained prominence and continues to be a valuable tool in the field of text analysis.

Sentence-BERT (SBERT): (Reimers and Gurevych, 2019) short for Sentence Embeddings using Siamese BERT-Networks, is a method for creating meaningful representations of sentences using the powerful BERT language model. Unlike BERT (Devlin et al., 2019), which focuses on individual words, SBERT aims to capture the overall semantic meaning of a sentence in a single, dense vector.

Sentence-T5 (ST5): (Ni et al., 2022) is a state-of-the-art sentence embedding method based on the Text-to-Text Transfer Transformer (T5) model, a powerful pre-trained model for various natural language processing tasks. ST5 scales the T5 (Raffel et al., 2019) model to generate sentence embeddings, providing a unified and efficient approach for encoding sentences into dense vector representations.

GPT Sentence Embeddings (SGPT): (Muennighoff, 2022) The GPT (Brown et al., 2020) model is a powerful language processing tool, trained on vast text data to generate human-like text and accurately capture semantic and syntactic nuances. SGPT, as a sentence embedding technique, utilizes GPT's strengths to convert sentences into dense vectors, reflecting semantic meanings. SGPT enhances semantic search by accurately representing sentence similarity, useful in a variety of applications.

Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP): (McInnes et al., 2020) is a dimension reduction technique that can be used for visualisation similarly to t-SNE, but also for general non-linear dimension reduction.

HDBSCAN: is a clustering algorithm developed by (Campello et al., 2013). It extends DBSCAN (Ester et al., 1996) by converting it into a hierarchical clustering algorithm, and then using a technique to extract a flat clustering based in the stability of clusters.

K-Means (Jin and Han, 2010): is a clustering algorithm widely used in data analysis. It groups similar data points into 'k' clusters based on their features, aiming to minimize the intra-cluster variance. The algorithm iteratively assigns data points to the nearest cluster center and updates the center until convergence.

4.2 Experiments pipeline

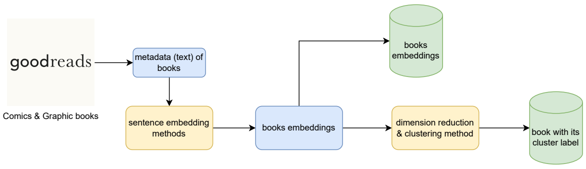

goodreads
Comics & Graphic books
metadata (text) of books
sentence embedding methods
books embeddings
books embeddings
dimension reduction & clustering method
book with its cluster label

Figure 4: Embedding Phrase

In the embedding phrase, firstly, we embedded the text_features from books into dense vectors using various embedding methods discussed earlier 4.1. Secondly, we employed UMAP + HDBSCAN for dimension reduction and clustering to cluster each book into its respective clusters. Finally, we save the user embeddings, along with the books and their cluster labels, in the dataset to support the evaluation phases.

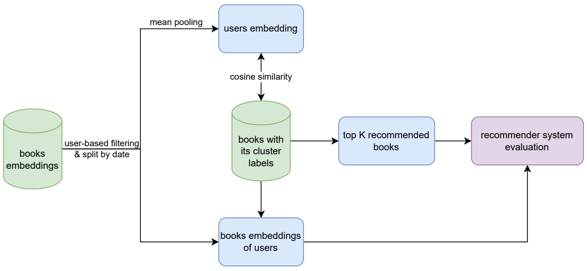

mean pooling
users embedding
cosine similarity
books embeddings
user-based filtering & split by date
books with its cluster labels
top K recommended books
recommender system evaluation
books embeddings of users

Figure 5: Evaluation Phrase (Recommendation Task)

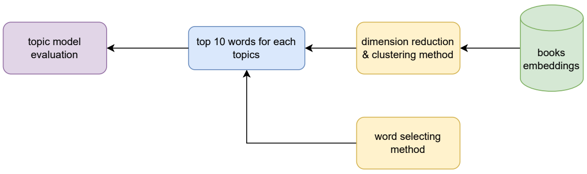

topic model evaluation
top 10 words for each topics
dimension reduction & clustering method
books embeddings
word selecting method

Figure 6: Evaluation Phrase (Topic Model Task)

For evaluation, we assess these methods for recommendation and topic model tasks. In the recommendation task, we reuse the previously stored book embeddings from the embedding phase. We then split the dataset into training and testing sets based on users and, for each user, consider the books they purchased on a specific date. Specifically, 70 percentages of the first books they bought are used for embedding methods, followed by mean pooling to create a user representation. Next, we calculate the similarity between the user representation and all books in the embeddings using cosine similarity. Finally, we compare the top K books (where K can be 5, 10, or 50) with 30 percentages of the books in the test set, based on clusters assigned in the embedding phase, and evaluate system performance using recall@K.

In the topic model task, we deviate from using the HDBSCAN clustering algorithm in the Embedding Phase. Instead, we opt for K-Means for clustering since HDBSCAN cannot determine a specific number of topics. Our goal is to limit the number of topics to fewer than 10 for practical and interpretative purposes. After applying book embeddings through UMAP + K-Means, we use TFIDF-IDFI, as suggested by (Zhang et al., 2022), for word selection. TFIDF-IDFI has been proven to be more effective than the c-TF-IDF method used in BERTopic (Grootendorst, 2022). This process helps us extract the top 10 importance words for each topic. Finally, we evaluate topic quality based on both diversity and coherence, as detailed in section 4.4

4.3 Experiments Settings

We do not specify a maximum sequence length for the TFIDF and Word2Vec methods because these approaches generate sentence embeddings by averaging the words in a sentence. In contrast, the remaining methods embed the entire sentence. It is noteworthy that the dimension of TFIDF is significantly larger than other methods. However, given that our vocabulary consists of 73,526 words and considering the operational principles of TFIDF, we have chosen the value of 8000 for effective representation. See table 2, 3 for more detail.

4.4 Evaluation Metrics

4.4.1 Recommendation Task

Recall, is calculated by the number of consumed items in the recommendation list out of the total number of items the user consumed. Authors have called recall as recall@k , where k stands for the size of the recommendation list.

$$\text{Recall@k} \ = \frac{\text{No. of relevant books in top K}}{\text{Total no. of relevant books}} \tag{1}$$

We define relevant books as those with a rating of 3 or higher in UserHistory Dataset.

4.4.2 Topic Modeling Task

To evaluate topic quality, we consider both topic diversity and coherence. We measure topic diversity using a score called Topic Uniqueness (TU), as suggested by (Nan et al., 2019). Given the top L words from each of the K topics, the TU for topic k is calculated:

$$\mathrm{TU}(k) = \frac{1}{L}\sum_{l=1}^{L}\frac{1}{\mathrm{cnt}(l,k)}, \quad k = 1,\ldots,K, \tag{2}$$

where $\mathrm{cnt}(l,k)$ is the total number of times the l^{th} top word in topic k appears in the top words across all topics.

For topic coherence, we use Normalized Pointwise Mutual Information (NPMI) (Newman et al., 2010), which involves counting word co-occurrence patterns within a sliding window. We also use Topic Coherence (C_{v}) (Röder et al., 2015), a variant of NPMI that counts word co-occurrences using one-set segmentation and cosine similarity as the similarity measure.

4.5 Results & Analysis

We report the main results in Table 4, 5, 6.

We evaluated the performance of several sentence embedding methods in a recommendation task and compared them with classic methods such as TFIDF and Word2Vec, which serve as our baseline. Our results demonstrate that modern sentence embedding methods, when utilized in the context of recommender systems, can significantly enhance the accuracy of recommendations. Specifically, we observed that the SBERT (pretrained bert-multilingual-uncased) model achieved the highest Recall@5 (0.3536), Recall@10 (0.3989), and Recall@50 (0.4574) scores, outperforming both the TFIDF and Word2Vec methods. Additionally, the ST5 and SGPT models also performed well, with recall@k score that were comparable to or better than the baselines.

In the topic quality analysis, we observed that modern methods outperformed our baselines, indicating their capability to generate more meaningful topics. Specifically, the SBERT (multilingual-uncased) model exhibited a higher TU score than the SBERT (base-uncased) model, highlighting the multilingual model's ability to identify a greater diversity of words in topics. Moreover, the SGPT model achieved the highest TU and NPMI scores, followed by the ST5 model. These pretrained models, trained on extensive datasets with a large number of parameters, demonstrated robust performance.

We also evaluated the execution time of embedding methods in the recommendation task across three document length categories based on statistics from MetaBooks: documents with lengths less than 58 (short), lengths between 58 and 126 (medium), and lengths greater than 126 (long). The results indicated that Word2Vec is the fastest, with an execution time of 0.04 across all three document lengths. TFIDF encountered out-of-memory issues (indicated by “-”) when evaluated in medium and long documents length in Google Colab (12GB). This occurred due to increased memory requirements as the dimensionality of the experiment (8000 important words in our experiment) grew. Additionally, due to their extensive parameters and complex model architecture, ST5 and SGPT demonstrated strong performance in both the recommendation task and topic model task but required more time for execution. Overall, we find that ST5 produces good results with a relatively acceptable runtime.

Methods	Max sequence length	Dim embedding	Configs
TFIDF	unlimited	8000	min_df=4, max_df=200, max_features=8000
Word2Vec	unlimited	300	pretrained: GoogleNews-vectors-negative300
SBERT	512	768	pretrained: bert-base-uncased, bert-base-multilingual-uncased
ST5	512	768	pretrained: sentence-t5-base
SGPT	512	768	pretrained: SGPT-125M-weightedmean-nli- bitfit

Table 2: Configuration of embedding methods

	Configs
UMAP	n_neighbors=15, n_components=5, metric=cosine
HDBSCAN	min_cluster=10, metric=euclidean, cluster_method=eom
K-Means	n_clusters = [5,7,9]

Table 3: Configuration of dimension reduction & clustering methods

Methods	Recall@5	Recall@10	Recall@50
TFIDF	0.2163	0.2644	0.3950
Word2Vec	0.3148	0.3475	0.3997
SBERT (bert-base-uncased)	0.3229	0.3664	0.4443
SBERT (bert-base-multilingual-uncased)	**0.3536**	**0.3989**	**0.4574**
ST5 (base)	0.3477	0.3645	0.4447
SGPT	0.3514	0.3853	0.4368

Table 4: The average recall@k is calculated by running HDBSCAN 10 times for each k-value, utilize a randomly selected subset of with 4000 users

The demo of our recommender system is shown in A

5 Conclusion and Future Works

These findings suggest that modern sentence embedding methods can be a valuable tool in enhancing the accuracy of recommender systems, and that they offer a promising alternative to classic methods such as TFIDF and Word2Vec. Leveraging current pre-trained models enables the rapid creation of high-quality item embeddings. Overall, our results highlight the potential of sentence embedding methods in the context of recommender systems and provide a foundation for future research in this area.

Methods	Avg TU	Avg NPMI	Avg C_v
TFIDF	0.8346	0.0080	0.5112
Word2Vec	0.8322	0.0930	0.5689
SBERT (bert-base-uncased)	0.8518	0.1151	0.6006
SBERT (bert-base-multilingual-uncased)	0.8420	0.1402	0.6392
ST5 (base)	0.8506	0.1477	**0.6422**
SGPT	**0.8529**	**0.1487**	0.6246

Table 5: The average topic diversity (TU), topic coherence (NPMI, C_v) of top 10 words are computed using K-Means, running 10 times for each k-value (5,6,7)

Methods	Short	Medium	Long
TFIDF	2.912	-	-
Word2Vec	0.048	0.050	0.044
Bert-base-uncased	0.316	0.536	0.838
Bert-base-multilingual-uncased	0.340	0.574	0.888
ST5	0.360	0.630	1.018
SGPT	0.422	0.700	0.960

Table 6: Time execution of methods on the recommendation task with different document lengths

In the recommendation task, we employed a single configuration for UMAP+HDBSCAN to compare various embedding methods. However, each method yielded embeddings with different dimension sizes. Therefore, In the future works, we aim to investigate the impact of dimension reduction and clustering algorithms techniques on these embedding methods. Furthermore, we plan to deploy our recommender system and conduct an online evaluation, such as A/B testing, to assess its effectiveness.

Limitations

While our study demonstrates the effectiveness of modern sentence embedding methods in enhancing recommender systems, it's important to acknowledge certain limitations. One notable concern is the potential impact of randomness in clustering algorithms as well as the no control in number of topic of HDBSCAN clustering method, which could lead to erroneous results. The inherent variability in these algorithms may introduce inconsistencies, affecting the reliability and stability of the clustering outcomes. Addressing this limitation could involve exploring strategies to mitigate the influence of randomness and enhance the robustness of the proposed approaches.

References

Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. 2015. Research-paper recommender systems: A literature survey. *International Journal on Digital Libraries*, pages 1–34.

Ye Bi, Liqiang Song, Mengqiu Yao, Zhenyu Wu, Jianming Wang, and Jing Xiao. 2020. A heterogeneous information network based cross domain insurance recommendation system for cold start users. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 2211–2220, New York, NY, USA. Association for Computing Machinery.

Selva Birunda and R.Kanniga Devi. 2021. *A Review on Word Embedding Techniques for Text Classification*, pages 267–281.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Ricardo J. G. B. Campello, Davoud Moulavi, and Joerg Sander. 2013. Density-based clustering based on hierarchical density estimates. In *Advances in Knowledge Discovery and Data Mining*, pages 160–172, Berlin, Heidelberg. Springer Berlin Heidelberg.

Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, Francois Belletti, and Ed H. Chi. 2019. Top-k off-policy correction for a reinforce recommender system. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, WSDM '19, page 456–464, New York, NY, USA. Association for Computing Machinery.

Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, New York, NY, USA.

Natalie K. Cygan. 2021. Sentence-bert for interpretable topic modeling in web browsing data.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure.

Sayed Nasir Hasan and Ravi Khatwal. 2022. Cold start problem in recommendation system: A solution model based on clustering and association rule techniques. In *2022 5th International Conference on Multimedia, Signal Processing and Communication Technologies (IMPACT)*, pages 1–8.

Xin Jin and Jiawei Han. 2010. *K-Means Clustering*, pages 563–564. Springer US, Boston, MA.

Budi Juarto and Abba Girsang. 2021. Neural collaborative with sentence bert for news recommender system. *JOIV : International Journal on Informatics Visualization*, 5:448.

Manoj Kumar, Dharmendra Yadav, Ankur Singh, and Vijay Kr. 2015. A movie recommender system: Movrec. *International Journal of Computer Applications*, 124:7–11.

Leland McInnes, John Healy, and James Melville. 2020. Umap: Uniform manifold approximation and projection for dimension reduction.

Chahrazed Mediani, Saad Harous, and Mahieddine Djoudi. 2023. Content-based recommender system using word embeddings for pedagogical resources. In *2023 5th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, pages 1–8.

Saulo Mendes de Melo, André Lima Férrer de Almeida, Lívia Almada Cruz, and Ticiana Linhares Coelho da Silva. 2022. A chat recommender system for covid-19 support based in textual sentence embeddings. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, WI-IAT '21, page 248–252, New York, NY, USA. Association for Computing Machinery.

Tomas Mikolov, Ilya Sutskever, Kai Chen, G.s Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26.

Raymond J. Mooney and Loriene Roy. 2000. Content-based book recommending using learning for text categorization. In *Proceedings of the Fifth ACM Conference on Digital Libraries*, DL '00, page 195–204, New York, NY, USA. Association for Computing Machinery.

Niklas Muennighoff. 2022. Sgpt: Gpt sentence embeddings for semantic search.

Feng Nan, Ran Ding, Ramesh Nallapati, and Bing Xiang. 2019. Topic modeling with Wasserstein autoencoders. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6345–6381, Florence, Italy. Association for Computational Linguistics.

David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 100–108, Los Angeles, California. Association for Computational Linguistics.

M. E. J. Newman. 2005. Power laws, pareto distributions and zipf's law. *Contemporary Physics*, 46(5):323–351.

Hai Nguyen, Jérémie Mary, and Philippe Preux. 2014. Cold-start problems in recommendation systems via contextual-bandit algorithms.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874, Dublin, Ireland. Association for Computational Linguistics.

Braja Gopal Patra, Dipankar Das, and Sivaji Bandyopadhyay. 2017. Retrieving similar lyrics for music recommendation system. In *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, pages 290–297, Kolkata, India. NLP Association of India.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Paul Resnick and Hal R. Varian. 1997. Recommender systems. *Commun. ACM*, 40(3):56–58.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the space of topic coherence measures. *WSDM 2015 - Proceedings of the 8th ACM International Conference on Web Search and Data Mining*, pages 399–408.

Salim Salmi, Saskia Mérelle, Renske Gilissen, and Willem-Paul Brinkman. 2021. Content-based recommender support system for counselors in a suicide prevention chat helpline: Design and evaluation study. *Journal of Medical Internet Research*, 23.

Bin Wang and C.-C. Jay Kuo. 2020. Sbert-wk: A sentence embedding method by dissecting bert-based word models. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 28:2146–2157.

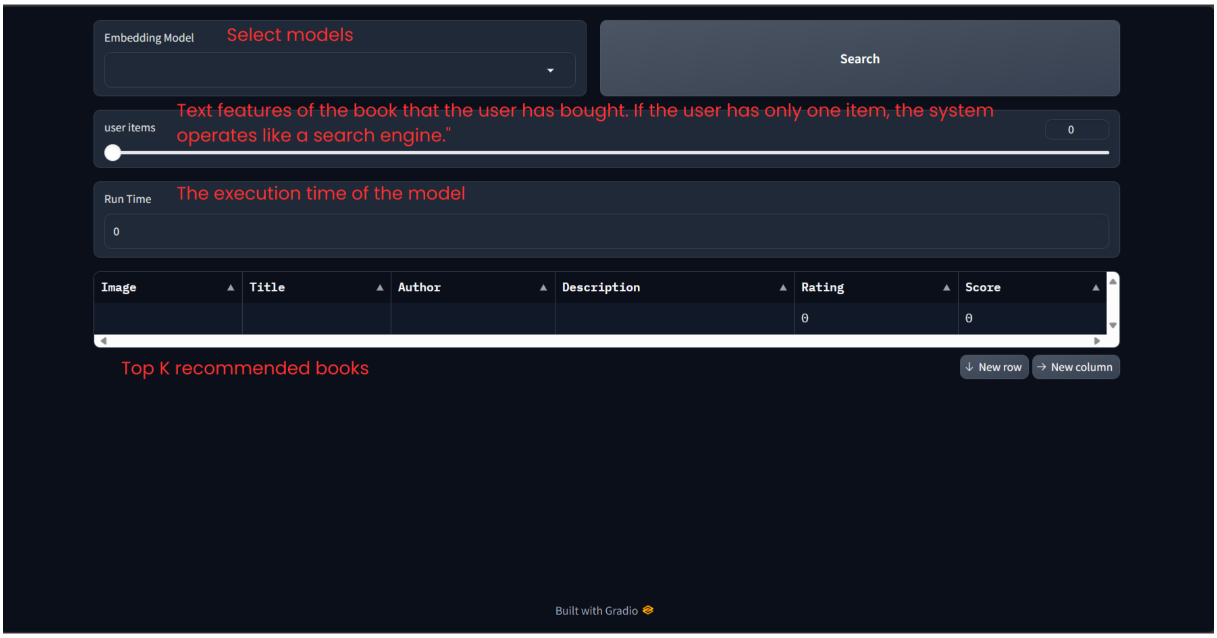
Nuofan Xu and Chenhui Hu. 2023. Enhancing e-commerce recommendation using pre-trained language model and fine-tuning.

Chong Yang, Xiaohui Yu, Yang Liu, Yanping Nie, and Yuanhong Wang. 2016. Collaborative filtering with weighted opinion aspects. *Neurocomputing*, 210.

Zihan Zhang, Meng Fang, Ling Chen, and Mohammad Reza Namazi Rad. 2022. Is neural topic modelling better than clustering? an empirical study on clustering with contextual embeddings for topics. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3886–3893, Seattle, United States. Association for Computational Linguistics.

A Demo of our content-based recommender system

In this appendix, we present a demonstration of our content-based recommender system built using Gradio. This showcases the interactive features and functionality of our recommendation model within the user-friendly Gradio interface.



Embedding Model

Select models

Text features of the book that the user has bought. If the user has only one item, the system operates like a search engine."

user items: 0

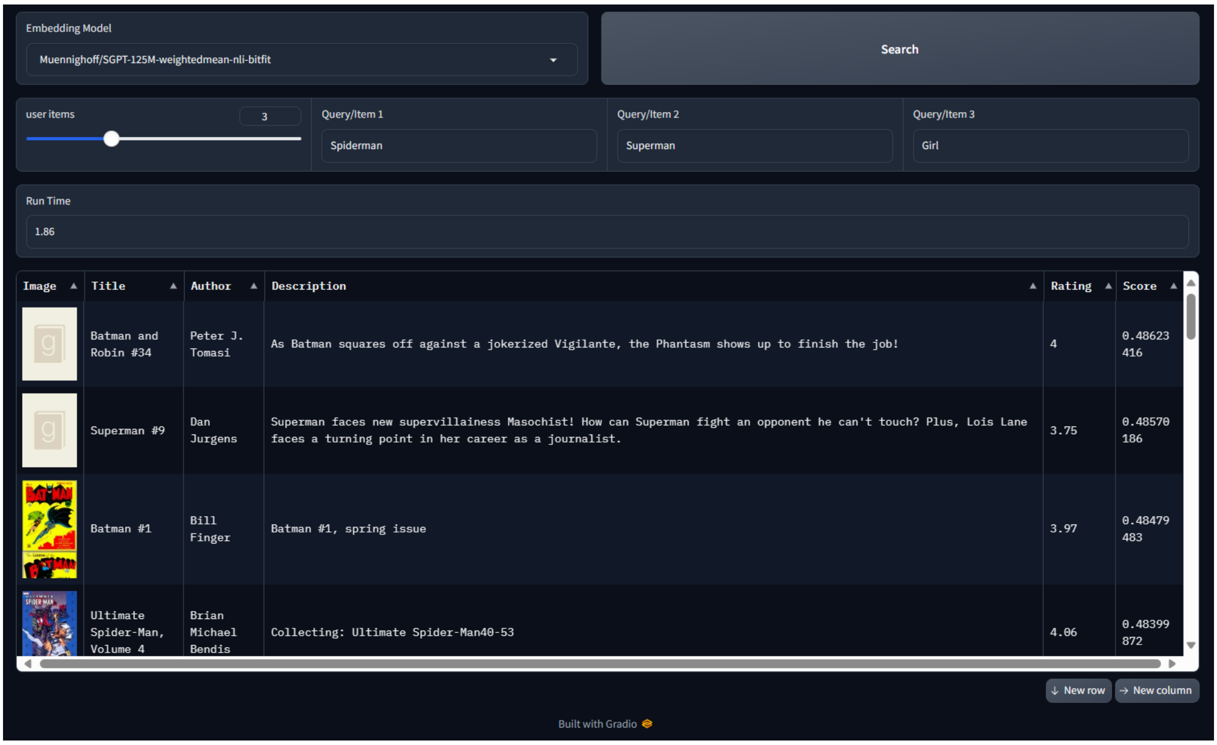
Run Time: 0

The execution time of the model

Image	Title	Author	Description	Rating	Score
				0	0

Top K recommended books

Figure 7: System Interface



Embedding Model: Muennighoff/SGPT-125M-weightedmean-nli-bitfit

user items: 3

Query/Item 1: Spiderman

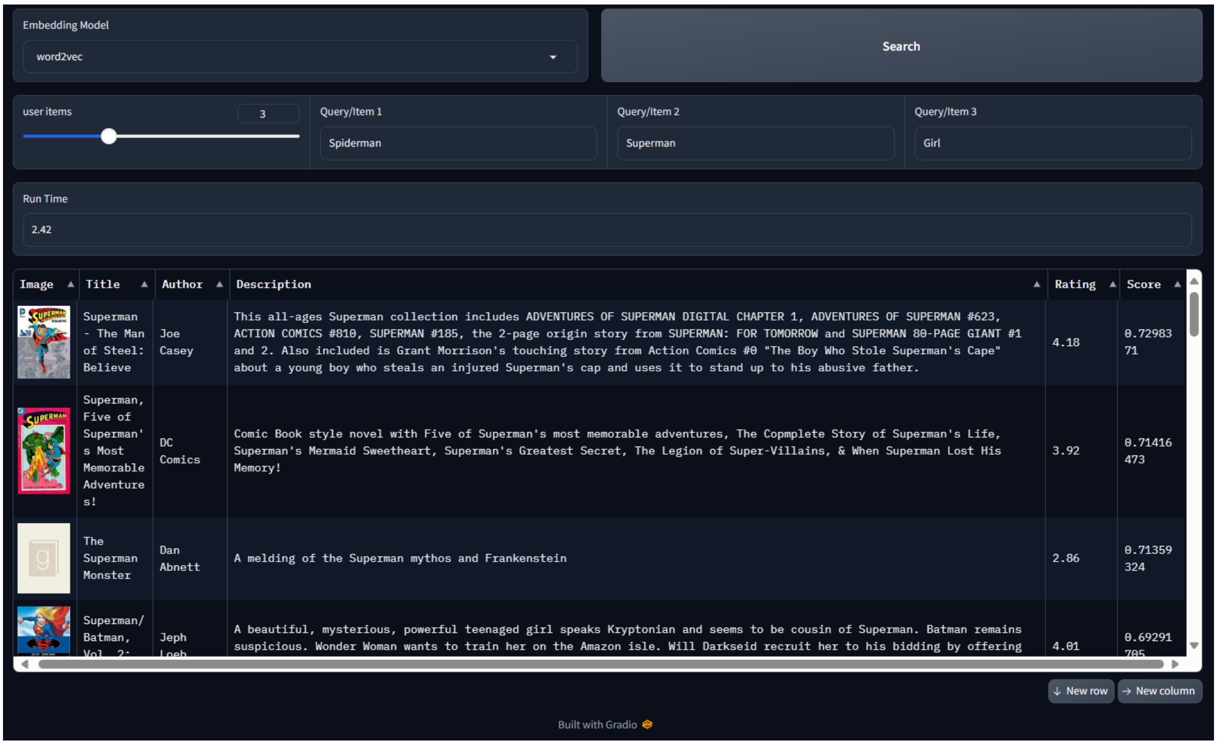
Query/Item 2: Superman

Query/Item 3: Girl

Run Time: 1.86

Image	Title	Author	Description	Rating	Score
	Batman and Robin #34	Peter J. Tomasi	As Batman squares off against a jokerized Vigilante, the Phantasm shows up to finish the job!	4	0.48623416
	Superman #9	Dan Jurgens	Superman faces new supervillainess Masochist! How can Superman fight an opponent he can't touch? Plus, Lois Lane faces a turning point in her career as a journalist.	3.75	0.48570186
	Batman #1	Bill Finger	Batman #1, spring issue	3.97	0.48479483
	Ultimate Spider-Man, Volume 4	Brian Michael Bendis	Collecting: Ultimate Spider-Man40-53	4.06	0.48399872

Figure 8: System use SGPT



Embedding Model

word2vec

user items 3

Query/Item 1: Spiderman

Query/Item 2: Superman

Query/Item 3: Girl

Run Time

2.42

Image	Title	Author	Description	Rating	Score
	Superman - The Man of Steel: Believe	Joe Casey	This all-ages Superman collection includes ADVENTURES OF SUPERMAN DIGITAL CHAPTER 1, ADVENTURES OF SUPERMAN #623, ACTION COMICS #810, SUPERMAN #185, the 2-page origin story from SUPERMAN: FOR TOMORROW and SUPERMAN 80-PAGE GIANT #1 and 2. Also included is Grant Morrison's touching story from Action Comics #0 "The Boy Who Stole Superman's Cape" about a young boy who steals an injured Superman's cap and uses it to stand up to his abusive father.	4.18	0.7298371
	Superman, Five of Superman's Most Memorable Adventures!	DC Comics	Comic Book style novel with Five of Superman's most memorable adventures, The Copmplete Story of Superman's Life, Superman's Mermaid Sweetheart, Superman's Greatest Secret, The Legion of Super-Villains, & When Superman Lost His Memory!	3.92	0.71416473
	The Superman Monster	Dan Abnett	A melding of the Superman mythos and Frankenstein	2.86	0.71359324
	Superman/ Batman, Vol. 2:	Jeph Loeb	A beautiful, mysterious, powerful teenaged girl speaks Kryptonian and seems to be cousin of Superman. Batman remains suspicious. Wonder Woman wants to train her on the Amazon isle. Will Darkseid recruit her to his bidding by offering	4.01	0.69291705

Figure 9: System use Word2Vec