

# Implementation of Content-Based Filtering in Book Recommender Systems Using K-Nearest Neighbor Model with Singular Value Decomposition and Word2Vec

Naufal Afif Sadewa<sup>1</sup>, Subhan<sup>2</sup>

<sup>1,2</sup>Computer Science Department, Faculty of Mathematics and Natural Sciences,  
Universitas Negeri Semarang, Indonesia

**Abstract.** Books are a medium for understanding various topics, such as science, history, and culture. With the development of digital technology, accessing books has become easier, but choosing the right book among thousands of choices is a challenge. Book recommender systems are an efficient solution to help users find relevant books. One approach that can be used in book recommender systems is Content-Based Filtering, which utilizes content information in books to provide recommendations.

**Purpose:** This research aims to develop a book recommender system by implementing Content-Based Filtering using K-Nearest Neighbor model with a combination of Singular Value Decomposition and Word2Vec in recommending relevant books according to each preference.

**Methods/Study design/approach:** The method used involves several stages. First, data preprocessing is carried out to remove noise so as to retain important information. After data preprocessing, feature extraction uses the Term Frequency-Inverse Document Frequency method to represent book features through vectors. The result of this vector is then reduced in dimension using Singular Value Decomposition to reduce complexity and capture the most significant data structures. At another stage, book features are extracted using Word2Vec, which produces a semantic representation of the word in vector form. Next, the vector results from Singular Value Decomposition and Word2Vec are combined to form more informative features. Finally, the K-Nearest Neighbor model using the cosine similarity distance metric is used to calculate the similarity between books based on the combined features, so as to generate relevant book recommendations. Tests were conducted on the GoodReads Best Books dataset taken from Kaggle.

**Result/Findings:** The evaluation results show that the proposed model is able to provide recommendations with good relevance values, measured using evaluation metrics such as Mean Average Precision and Normalized Discounted Cumulative Gain with the scores obtained respectively, namely 0.9637 and 0.9515 at parameter  $k$  is 5.

**Novelty/Originality/Value:** The novelty in this research is the combination of Singular Value Decomposition vectors and Word2Vec vectors to produce a more informative feature representation, by utilizing statistical relationships between words and capturing the semantic meaning of words.

**Keywords:** Recommender System, Content-Based Filtering, K-Nearest Neighbor, Singular Value Decomposition, Word2Vec

**Received** February 2025 / **Revised** March 2026 / **Accepted** March 2026

This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).



## INTRODUCTION

Books are a medium for understanding various topics, such as science, history, and culture. Reading books is important for enriching knowledge, broadening horizons, and practicing critical thinking skills [1]. In the digital era, accessing books has become easier, but choosing the right book is a challenge. Book recommender systems offer a solution to help users find relevant books efficiently. Recommender systems are designed to filter and sort book choices based on user preferences, enabling relevant recommendations faster than manual searches [2]. With algorithms that learn user habits, recommendations become more personalized and targeted [3].

According to Fu & Ma [4], recommender systems are divided into two main categories: Collaborative Filtering (CF) and Content-Based Filtering (CBF). CF analyzes the preferences of other users with similar interaction patterns, but faces the cold-start problem when a new user or book has no interaction data yet

---

<sup>1</sup>\*Corresponding author.

Email addresses: [naufalafif766@gmail.com](mailto:naufalafif766@gmail.com) (Sadewa)

DOI: 10.15294/rji.v4i1.21281

[5]. Meanwhile, CBF focuses on the characteristics of books, such as genre and topic, to match them with users' interests, making it more reliable for new users or those with specific preferences [6]. In the application of CBF to book recommender systems, the K-Nearest Neighbor (KNN) model is used to calculate the similarity between items based on features such as genre or theme, where the closest item becomes the main recommendation [7]. The advantage of KNN is its simplicity, but its performance may degrade with increasing data due to increased computational requirements [8].

The research of Delimayanti et al. [9] developed a CBF-based movie recommender web system, which uses a combination of movie characteristics with cosine similarity and KNN algorithm. With the Movie Lens dataset, the system achieved an accuracy of more than 87%, demonstrating the effectiveness of this method in providing recommendations according to users' genre preferences. Related research was also conducted by Dharmawan et al. [10] in designing a book recommender system using the KNN algorithm and Ball Tree data structure to accelerate the search for books similar to user preferences. With a dataset of 2,194 ratings from Gramedia, this system achieved an average precision@ $k$  of 80% in five test scenarios.

Research by Wicaksono & Setiawan [11] developed a user-based movie recommender system using ratings from tweets, by combining CF, Singular Value Decomposition (SVD), and Support Vector Machine (SVM). The test results show the best Root Mean Squared Error (RMSE) value of 0.8162 for user-based and 0.5911 for item-based. The combination of SVD and SVM produced precision and recall of 81% for user-based and 80% for item-based. In the mentioned research, SVD plays an important role in CF by reducing data dimensionality and extracting hidden patterns between items and user preferences [12].

In Pradana & Wibowo [13] research implemented the Hybrid Filtering (HF) method with Word2Vec for CBF using Google's pre-trained model and Restricted Boltzmann Machine (RBM) for CF using TensorFlow. With the Movie Lens dataset, the evaluation shows the best results in Precision@20 of 0.82 and Normalized Discounted Cumulative Gain (NDCG)@20 of 0.775. A related research was also conducted by Nguyen et al. [14] developed a similarity-based CBF approach between items for movie recommender with features such as genre, director, actors, and plot. Using Jaccard coefficient index and soft cosine similarity, as well as Word2Vec to represent the plot as a semantic vector, the system excelled with 82.636% accuracy, 83.433% precision, 77.301% recall, and 81.026% F1-score, especially in cold-start conditions. In that research Nguyen et al. [14], the Word2Vec word embedding model to convert words into numerical vectors, allowing words with similar contexts to be represented by adjacent vectors. This approach is more effective than bag of words or TF-IDF because it is able to capture the semantic meaning of words in a particular context.

Overall, this research focuses on developing a CBF-based book recommender system using a combination of KNN, SVD, and Word2Vec. KNN provides quick recommendations based on book similarity, while SVD and Word2Vec model the relationship between books in depth. This combination aims to improve efficiency and accuracy, especially for users with limited interaction history, so that the system can provide recommendations according to personal preferences.

## **METHODS**

Research approach and design are required to ensure the research runs more directed in accordance with the predetermined provisions. In this research, a recommender system is designed using the KNN model with a combination of SVD and Word2Vec to be able to provide relevant book recommendations. This research follows several stages, starting from collecting the GoodReads dataset. Then, the dataset goes through a data preprocessing stage to filter out important information. Next, feature extraction using TF-IDF and dimension reduction with SVD. In another stage, feature extraction utilizes word embedding from Word2Vec. Next, combine the result vectors from SVD and Word2Vec to be used as input for the KNN model. Finally, evaluate the model with Mean Average Precision (MAP) and NDCG metrics to determine model performance. The research flow is shown in Figure 1.

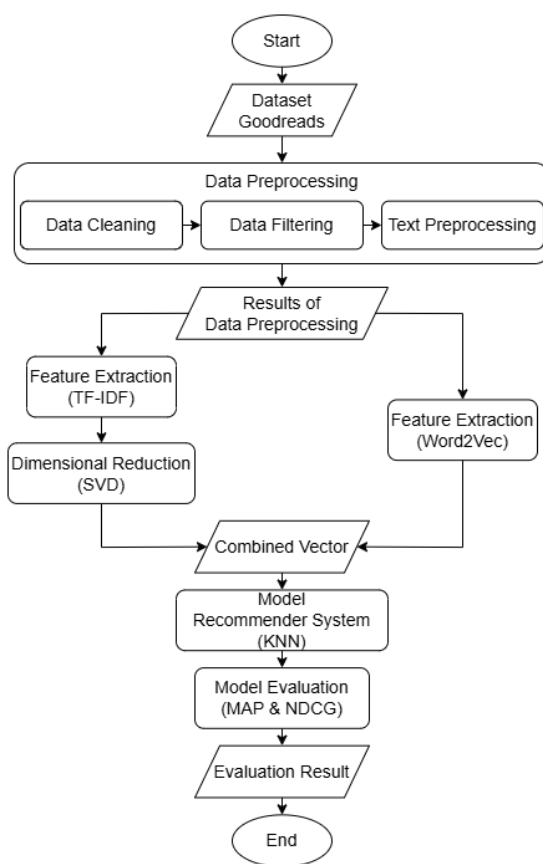


Figure 1. Flowchart of Research Methods

### Dataset

The data used in this research is in the form of book metadata that has been listed on the Goodreads site. The dataset is "GoodReads Best Books" which is collected from Kaggle. The dataset is unsupervised, where data training is done without using labels. This dataset can be utilized as research material for various approaches in recommender system models. Some features that will be used to build a recommender system model can be seen in Table 1.

Table 1. Dataset Features

Features	Description
title	The title of the book
author	The author of the book
genres	The genres of the book fall into
description	A brief description of the book
series	The series of the book belongs to
language	The language of the book is written in
bookFormat	The format of the book
publisher	The publisher of the book

### Data Preprocessing

Data preprocessing is done to adjust the raw data to what is needed by the model so that the analysis process can run more effectively and efficiently. In this research, data preprocessing includes data cleaning, data filtering, and text preprocessing [15]. First, data cleaning is a process to clean the dataset to avoid noise and redundancy. In this stage, missing values and duplicate data are treated. Second, data filtering will be applied to filter the data by providing thresholds on some features and resetting the index. Third, text preprocessing has several steps, namely case folding converts text into lowercase letters; remove special characters, numbers, punctuations, and extra spaces; tokenization breaks text into tokens; stopwords removal eliminates common words; stemming returns words to their basic form [16].

### Feature Extraction with TF-IDF

The Term Frequency-Inverse Document Frequency (TF-IDF) feature extraction process is done by measuring the importance of a word in a document in representing corpus data. In TF-IDF, the value will increase along with the high frequency of word occurrence in a document, but the value will decrease if the word often appears in other documents [9]. TF-IDF is calculated by multiplying the TF and IDF values for a word in a particular document. The following equations 1, 2, and 3 are the formula for calculating TF-IDF [17].

$$TF(t, d) = \frac{\text{Frequency of term } t \text{ in document } d}{\text{Total terms in document } d} \quad (1)$$

$$IDF(t) = \log\left(\frac{N}{1+df_t}\right) \quad (2)$$

$$TF - IDF(t, d) = TF(t, d) \times IDF(t) \quad (3)$$

where  $N$  is the total number of documents and  $df_t$  is the number of documents containing term  $t$ .

### Dimension Reduction with SVD

The vector or matrix generated by TF-IDF has a size of  $M \times N$ , where  $M$  as rows represent the number of documents and  $N$  as columns represent the words in the corpus. The matrix will be reduced in dimensions using the SVD technique to reduce complexity and capture important information [18]. The SVD decomposition is mathematically expressed by the equation 4.

$$A = U \Sigma V^T \quad (4)$$

where:

$A$  is the original matrix.

$U$  is an orthogonal matrix of left singular vectors.

$\Sigma$  is a diagonal matrix of singular values.

$V^T$  is an orthogonal matrix of right singular vectors.

### Feature Extraction with Word2Vec

In the Word2Vec model, the core of word representation learning is to map words into a vector space, so that words that have similar meanings have neighboring vector representations. To achieve this, the model must calculate the relationship between words based on their context. The Word2Vec model has two approaches in representing words as vectors, one of which is CBOW which will be used in this research. The CBOW approach predicts the target word (middle word) based on the surrounding words [19].

### Combination of SVD and Word2Vec Vectors

At this stage, the SVD and Word2Vec result vectors are combined using a horizontal stack to improve the quality of data representation. SVD reduces the dimensionality of the data while retaining important information, while Word2Vec generates word embedding that represents the context of the word. This combination leverages the advantages of SVD in identifying global structures and Word2Vec in capturing semantic relationships, resulting in a more in-depth and comprehensive representation to improve the accuracy of text analysis.

### KNN Model

In model building, a combination of vectors is used as input for the KNN model, which is used to find the nearest neighbor based on a certain distance [20]. This model is unsupervised with no learning process and uses cosine similarity as a distance metric to calculate the similarity between vectors. This approach is effective in text analysis and feature-based recommendation systems [9].

### Model Evaluation

The model evaluation in this study uses the evaluation metrics MAP and NDCG, which are ways to measure how well the recommendation system provides relevant results. These two metrics are often used to assess the accuracy of recommendations based on the order of the results provided. The following equations 5, 6, and 7 are used to calculate MAP metrics [21].

$$P@k = \frac{rel_k}{k} \quad (5)$$

$$AP = \frac{1}{R} \sum_{i=1}^n P@k \times rel(k) \quad (6)$$

$$MAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (7)$$

where:

$R$  is the number of relevant items.

$P@k$  is the precision at position  $k$ .

$rel(k)$  is 1 if the item at position  $k$  is relevant, 0 otherwise.

$N$  is the number of queries.

Equations 8 and 9 are used to calculate NDCG metrics [22].

$$DCG_p = \sum_{i=1}^p \frac{rel_i}{(i+1)} \quad (8)$$

$$NDCG_p = \frac{DCG_p}{IDCG_p} \quad (9)$$

where:

$rel_i$  is the relevance score of the item at position  $i$ .

$IDCG_p$  is the Ideal DCG, which is the DCG of the best possible ranking.

## RESULT AND DISCUSSION

This research aims to find out how the results of CBF implementation using the KNN model on a recommender system with SVD and Word2Vec. The recommendation process using the model is carried out through various processes as follows. The dataset used in this research is “GoodReads Best Books” which is public and available for download on Kaggle. The selected dataset is a dataset containing book metadata from the Goodreads site. The dataset has a total of 49,927 rows of data that will be used in this research.

After collecting the data, the next step is to process that data effectively so that the model training process can be done efficiently and accurately. At this stage, three data preprocessing steps are carried out, namely data cleaning, data filtering, and text preprocessing. The explanation of each step is as follows. The dataset potentially contains missing values which can cause the data to look large but actually empty. Therefore, a checking step is required and the following Table 2 shows the number of missing values in each feature.

Table 2. Number of Missing Values

Features	Missing Value
title	0
author	0
genres	0
description	1,338
series	29,008
language	3,806
bookFormat	1,473
publisher	3,696

It was found that the dataset had missing values in some features. Therefore, it is necessary to treat it by filling in the missing values. The “language” feature will be filled with the default value of ‘English’ if there are missing values. Meanwhile, the “series, description” feature will be filled with the value of the “title” feature and the “bookFormat, publisher” feature will be filled with an empty string if there are missing values. In addition, this step also handles duplicate data by removing redundant data in the “title” feature.

After data cleaning, the next step is data filtering. At this step, the data will be filtered based on the “language” feature by retaining data that has the values ‘English’ and ‘Indonesian’. As well as filtering against empty or inappropriate entries in the “genres” feature. After filtering is complete, it is necessary to

update the index data to ensure that the order matches the filtered data. Details of the comparison of the amount of data before and after passing the data cleaning and data filtering process are shown in Table 3.

Table 3. Data Amount after Filtering

Initial Data Amount	Wasted Data	Data Amount
49,927	9,768	40,159

The next step is text preprocessing, which aims to facilitate text processing by the model. This process is shown in Table 4 and continued to Table 5.

Table 4. Text Preprocessing Steps

Original Text	Case Folding	Remove Special Characters
Roman Tetralogi Buru mengambil latar belakang dan cikal bakal nation Indonesia di awal abad ke-20. Dengan membacanya waktu kita dibalikkan sedemikian rupa dan hidup di era membibitnya pergerakan nasional mula-mula, juga pertautan rasa, kegamangan jiwa, percintaan, dan pertarungan kekuatan anonim para srikandi yang mengawal penyemaian bangunan nasional yang kemudian kelak melahirkan Indonesia modern.	roman tetralogi buru mengambil latar belakang dan cikal bakal nation indonesia di awal abad ke-20. dengan membacanya waktu kita dibalikkan sedemikian rupa dan hidup di era membibitnya pergerakan nasional mula-mula, juga pertautan rasa, kegamangan jiwa, percintaan, dan pertarungan kekuatan anonim para srikandi yang mengawal penyemaian bangunan nasional yang kemudian kelak melahirkan indonesia modern.	roman tetralogi buru mengambil latar belakang dan cikal bakal nation indonesia di awal abad ke dengan membacanya waktu kita dibalikkan sedemikian rupa dan hidup di era membibitnya pergerakan nasional mula juga pertautan rasa kegamangan jiwa percintaan dan pertarungan kekuatan anonim para srikandi yang mengawal penyemaian bangunan nasional yang kemudian kelak melahirkan indonesia modern

Table 5. Continued Text Preprocessing Steps

Tokenization	Stopwords Removal	Stemming
['roman', 'tetralogi', 'buru', 'mengambil', 'latar', 'belakang', 'dan', 'cikal', 'bakal', 'nation', 'indonesia', 'di', 'awal', 'abad', 'ke', 'dengan', 'membacanya', 'waktu', 'kita', 'dibalikkan', 'sedemikian', 'rupa', 'dan', 'hidup', 'di', 'era', 'membibitnya', 'pergerakan', 'nasional', 'mula', 'mula', 'juga', 'pertautan', 'rasa', 'kegamangan', 'jiwa', 'percintaan', 'dan', 'pertarungan', 'kekuatan', 'anonim', 'para', 'srikandi', 'yang', 'mengawal', 'penyemaian', 'bangunan', 'nasional', 'yang', 'kemudian', 'kelak', 'melahirkan', 'indonesia', 'modern']	['roman', 'tetralogi', 'buru', 'mengambil', 'latar', 'cikal', 'nation', 'indonesia', 'abad', 'membacanya', 'dibalikkan', 'rupa', 'hidup', 'era', 'membibitnya', 'pergerakan', 'nasional', 'pertautan', 'kegamangan', 'jiwa', 'percintaan', 'pertarungan', 'kekuatan', 'anonim', 'srikandi', 'mengawal', 'penyemaian', 'bangunan', 'nasional', 'kelak', 'melahirkan', 'indonesia', 'modern']	['roman', 'tetralog', 'buru', 'ambil', 'latar', 'cikal', 'nation', 'indonesia', 'abad', 'baca', 'balik', 'rupa', 'hidup', 'era', 'bibit', 'gera', 'nasion', 'taut', 'gamang', 'jiwa', 'cinta', 'tarung', 'kuat', 'anonim', 'srikandi', 'awal', 'semai', 'bangun', 'nasion', 'kelak', 'lahir', 'indonesia', 'modern']

After completing the data preprocessing steps, feature extraction will be executed using the TF-IDF and Word2Vec methods.

Feature extraction using TF-IDF method is performed to represent text in numerical form based on the frequency of occurrence of words and their importance in a document. Afterwards, dimensionality reduction is performed using the SVD method to reduce data complexity without removing important information. This process is used to extract the main features from the TF-IDF matrix so that it is more efficient to be used in modeling the recommender system. The following is an example of the results of the SVD dimension reduction shown in Table 6.

Table 6. Example of SVD Result Vector

Matrix	Length
[0.19441745, 0.03125465, 0.04396864, 0.0064614, -0.01817497, ..., -0.04419565, 0.00189647, 0.056442, 0.05191763, -0.04798343]	100

At another step, feature extraction utilizing word embedding from the Word2Vec model is used to generate a vector representation for each document by taking the average of the word vectors in the document. The following Table 7 shows an example of the result vector with Word2Vec embedding.

Table 7. Example of Word2Vec Result Vector

Matrix	Length
[-0.05312536, -0.01554606, 0.06998263, -0.02206726, 0.46495605, ..., 0.8218902, -0.11246391, -0.4313535, -0.4397001, -0.9279991]	100

After obtaining the results of the numeric vectors from SVD and Word2Vec. The next step is to merge these two vectors horizontally to produce a more informative feature representation. Next, it is necessary to normalize the values to have a uniform scale. An example of the result from combining the two vectors is shown in Table 8.

Table 8. Example of Vector Combination Result

Matrix	Length
[0.04304687, 0.00692024, 0.0097353, 0.00143065, -0.0040242, -0.01024789, 0.01172369, -0.00523987, 0.01469177, -0.00624008, ..., 0.07528822, 0.04719136, 0.16158692, 0.16310332, -0.08689814, 0.18197852, -0.02490116, -0.09550798, -0.09735604, -0.20547258]	200

After all those processes are done, the next step is to define several model scenarios based on the data used to build the model. The recommendation system model built in this research uses the KNN model with cosine similarity as the distance metric formula. The following Table 9 shows each scenario used in this research.

Table 9. Model Scenario

Scenario	TF-IDF & SVD	Word2Vec	KNN	Detail
1	✓			TF-IDF + SVD + KNN
2		✓		Word2Vec + KNN
3	✓	✓	✓	TF-IDF + SVD + Word2Vec + KNN (Hstack Combination)
4	✓	✓		TF-IDF + SVD + Word2Vec + KNN (Average Combination)

All scenarios use the same parameters, such as a cosine distance metric and a brute force algorithm.

The evaluation of the recommender system model in this research is carried out by utilizing certain features, namely “genres” to measure the relevance of the recommendations generated. This model uses an unsupervised approach, which means it does not rely on explicit labels, but rather explores similarity patterns in the data. Therefore, a threshold is used to determine the relevance value of the recommendation result data based on the number of genres that match the target data genre.

After determining a threshold on the evaluation of recommendation results, model evaluation is carried out using several evaluation metrics including Precision, MAP, and NDCG. Table 10 shows the evaluation metrics score results from each scenario using a sample size of 8,000 data.

Table 10. Model Result Comparison

Scenario	k	Precision	MAP	NDCG
1	5	0,9266	0,9454	0,9343
	10	0,9176	0,9383	0,9274
	15	0,9106	0,9334	0,9238
	20	0,9051	0,9294	0,9216
2	5	0,9249	0,9559	0,9462
	10	0,9105	0,9433	0,9368
	15	0,8992	0,9342	0,9319
	20	0,8909	0,9269	0,9289
3	5	0,9354	0,9632	0,9509
	10	0,9227	0,9517	0,9414
	15	0,9119	0,9435	0,9366
	20	0,9043	0,9368	0,9335
4	5	0,9347	0,9627	0,9504
	10	0,9212	0,9509	0,9408
	15	0,9111	0,9425	0,9359
	20	0,9032	0,9358	0,9329

Referring to Table 10, the best score is obtained in the third scenario and it is known that as the value of  $k$  increases (from 5 to 20), the scores of evaluation metrics Precision, MAP, and NDCG in each scenario show a decrease. This indicates that as more items (books) are considered (larger  $k$ ), accuracy at the closest relevance level becomes more difficult to maintain. The following Figure 2 shows a graph to determine the best performance of the model in the third scenario with  $k$  values between 1 and 10.

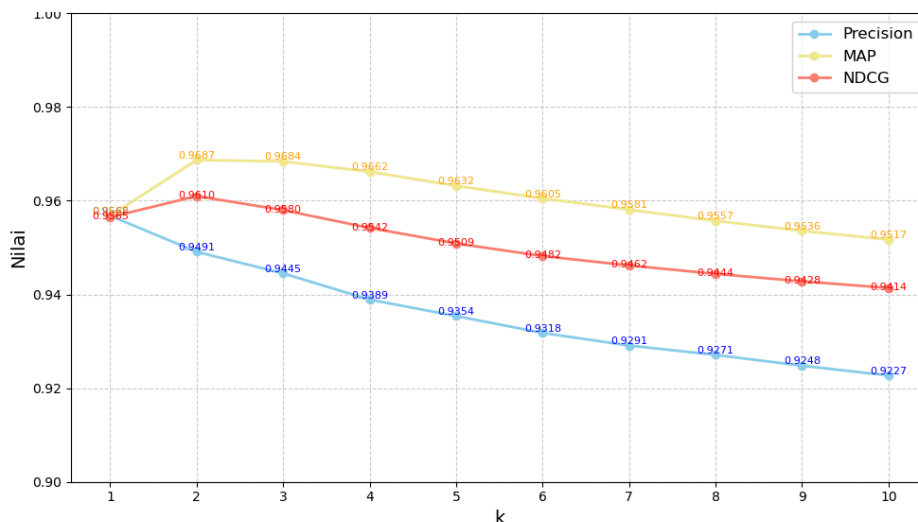


Figure 2. Comparison Graph of Scores for each  $k$

Referring to Figure 2, it is known that the highest scores on MAP and NDCG are obtained when the  $k$  value is 2. While the highest Precision value is obtained at a value of  $k$  is 1.

From the results obtained, it is known that the Precision, MAP, and NDCG evaluation metrics that produce the highest scores are achieved in scenarios that use a combination of SVD and Word2Vec vectors. The combination utilizes statistical relationships between words and captures the semantic meaning of words. Based on the evaluation results, it can be known that the KNN model with the combination of SVD and Word2Vec is superior in providing relevant book recommendations and ranking them appropriately. The model offers a more comprehensive approach to book feature representation.

## CONCLUSION

The implementation of CBF in a book recommender system using KNN model by combining SVD and Word2Vec shows that the combination of TF-IDF-based feature representation reduced using SVD and word vectors from Word2Vec is able to improve the recommendation quality. This process allows the system to capture semantic information from the book description text (Word2Vec) while retaining important features at document scale (SVD). The best scores obtained from the model evaluation are 0.9632 MAP and 0.9509 NDCG. This approach allows the model to more accurately find similarities between books, thus providing relevant recommendations and ranking them appropriately.

## REFERENCES

- [1] M. Hasan, H. Maulidyanti, M. I. T. Tahir, and N. Arisah, "Analisis Keterampilan Berpikir Kritis Peserta Didik melalui Kegiatan Literasi," 2022.
- [2] D. Sarma, T. Mitra, and M. S. Hossain, "Personalized Book Recommendation System using Machine Learning Algorithm," 2021. [Online]. Available: [www.ijacsa.thesai.org](http://www.ijacsa.thesai.org)
- [3] A. Lommatzsch, B. Kille, and S. Albayrak, "Incorporating Context and Trends in News Recommender Systems," in *Proceedings - 2017 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2017*, Association for Computing Machinery, Inc, Aug. 2017, pp. 1062–1068. doi: 10.1145/3106426.3109433.
- [4] L. Fu and X. Ma, "An Improved Recommendation Method Based on Content Filtering and Collaborative Filtering," *Complexity*, vol. 2021, 2021, doi: 10.1155/2021/5589285.
- [5] J. Wei, J. He, K. Chen, Y. Zhou, and Z. Tang, "Collaborative Filtering and Deep Learning Based Recommendation System for Cold Start Items," *Expert Syst Appl*, vol. 69, pp. 1339–1351, Mar. 2017, doi: 10.1016/j.eswa.2016.09.040.
- [6] J. Son and S. B. Kim, "Content-based Filtering for Recommendation Systems Using Multiattribute Networks," *Expert Syst Appl*, vol. 89, pp. 404–412, Dec. 2017, doi: 10.1016/j.eswa.2017.08.008.
- [7] S. Katkamr, A. Atikam, P. Mahesh, M. Chatre, S. S. Kumar, and G. R. Sakthidharan, "Content-based Movie Recommendation System and Sentimental analysis using ML," in *Proceedings of the 7th International Conference on*

- Intelligent Computing and Control Systems, ICICCS 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 198–201. doi: 10.1109/ICICCS56967.2023.10142424.
- [8] V. Subramaniaswamy and R. Logesh, “Adaptive KNN based Recommender System through Mining of User Preferences,” *Wirel Pers Commun*, vol. 97, no. 2, pp. 2229–2247, Nov. 2017, doi: 10.1007/s11277-017-4605-5.
- [9] M. K. Delimayanti *et al.*, “Web-Based Movie Recommendation System using Content-Based Filtering and KNN Algorithm,” in *Proceedings - 2022 9th International Conference on Information Technology, Computer and Electrical Engineering, ICITACEE 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 314–318. doi: 10.1109/ICITACEE55701.2022.9923974.
- [10] H. Dharmawan, S. Shofiah Hilabi, I. Kaniawulan, S. Informasi Fakultas Ilmu Komputer Universitas Buana Perjuangan Karawang Jl Ronggo Waluyo Sirnabaya, T. Timur, and J. Barat, “Sistem Rekomendasi Buku Dengan Metode K-Nearest Neighbor Pada Gramedia,” 2023.
- [11] R. A. Wicaksono and E. B. Setiawan, “Recommender System Based on Tweets with Singular Value Decomposition and Support Vector Machine Classification,” *Journal of Computer System and Informatics (JoSYC)*, vol. 3, no. 4, pp. 294–302, Sep. 2022, doi: 10.47065/josyc.v3i4.2072.
- [12] F. Colace, D. Conte, M. De Santo, M. Lombardi, D. Santaniello, and C. Valentino, “A Content-based Recommendation Approach based on Singular Value Decomposition,” *Conn Sci*, vol. 34, no. 1, pp. 2158–2176, 2022, doi: 10.1080/09540091.2022.2106943.
- [13] M. A. Pradana and A. T. Wibowo, “Movie Recommendation System Using Hybrid Filtering With Word2Vec And Restricted Boltzmann Machines,” *JIPi (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 9, no. 1, pp. 231–241, Feb. 2024, doi: 10.29100/jipi.v9i1.4306.
- [14] L. V. Nguyen, T.-H. Nguyen, J. J. Jung, and P. Kim, *Content-Based Collaborative Filtering using Word Embedding: A Case Study on Movie Recommendation*. The Association for Computing Machinery, 2020.
- [15] Moh. A. Rohman, Suhartono, and T. Chamidy, “Bidirectional GRU dengan Attention Mechanism pada Analisis Sentimen PLN Mobile Bidirectional GRU with Attention Mechanism on Sentiment Analysis of PLN Mobile,” 2023.
- [16] S. R. Aisy and B. Prasetyo, “Sentiment Analysis of the TPKS Law on Twitter Using InSet Lexicon with Multinomial Naïve Bayes and Support Vector Machine Based on Soft Voting,” *Recursive Journal of Informatics*, vol. 1, no. 2, pp. 93–101, Sep. 2023, doi: 10.15294/rji.v1i2.68324.
- [17] İ. KUŞ, S. BOZKURT KESER, and S. OKYAY, “A Novel Article Recommendation System Empowered by the Hybrid Combinations of Content-Based State-of-the-Art Methods,” *International Journal of Applied Mathematics Electronics and Computers*, vol. 11, no. 1, pp. 1–12, Mar. 2023, doi: 10.18100/ijamec.1199886.
- [18] J. Zhu, B. G. Patra, and A. Yaseen, “Recommender system of scholarly papers using public datasets,” 2021. [Online]. Available: <http://genestudy.org>.
- [19] H. D. Abubakar, M. Umar, and M. A. Bakale, “Sentiment Classification: Review of Text Vectorization Methods: Bag of Words, Tf-Idf, Word2vec and Doc2vec,” *SLU Journal of Science and Technology*, vol. 4, no. 1 & 2, pp. 27–33, Aug. 2022, doi: 10.56471/slujst.v4i.266.
- [20] A. Sawant *et al.*, “Content-Based Filtering Using K-Nearest Neighbor,” in *5th Biennial International Conference on Nascent Technologies in Engineering, ICNTE 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICNTE56631.2023.10146638.
- [21] B. Niu, J. Ma, and Z. Yang, “A Comparative Study of CF and NCF in Children’s Book Recommender System,” in *Proceedings - 2021 3rd International Workshop on Artificial Intelligence and Education, WAIE 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 43–47. doi: 10.1109/WAIE54146.2021.00017.
- [22] M. H. Maqbool, U. Farooq, A. Mosharraf, and A. B. Siddique, “MobileRec: A Large-Scale Dataset for Mobile Apps Recommendation,” in *SIGIR 2023 - Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Association for Computing Machinery, Inc, Jul. 2023, pp. 3007–3016. doi: 10.1145/3539618.3591906.