**IJCCS (Indonesian Journal of Computing and Cybernetics Systems)**
Vol.19, No.1, January 2025, pp. 61~72
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: https://doi.org/10.22146/ijccs.103611

# Enhancing Neural Collaborative Filtering with Metadata for Book Recommender System

**Putri Ayu Sedyo Mukti[1], Z. K. A. Baizal*[2]**
[1] School of Computing, Telkom University, Bandung, Indonesia
[2] School of Computing, Telkom University, Bandung, Indonesia
e-mail: [1]princezz@student.telkomuniversity.ac.id, *[2]**baizal@telkomuniversity.ac.id**

***Abstrak***

*Sistem rekomendasi buku sering kali menghadapi tantangan overload informasi dan item cold start dikarenakan dinamika pasar buku yang terus berkembang. Makalah ini mengusulkan Feature Enhanced Neural Collaborative Filtering (FENCF), sebuah metode baru yang menggabungkan interaksi antara pengguna dan item dengan informasi metadata genres untuk mengatasi masalah item cold start dan meningkatkan akurasi prediksi rating. Keunikan FENCF terletak pada pengolahan preproses metadata genres yang berbeda dari penelitian rekomendasi buku pada umumnya. Percobaan dengan dataset buku Amazon menunjukkan kontribusi FENCF yang mengungguli NCF, dengan mengurangi RMSE sebesar 4,04% dan MAE sebesar 2,73%. Selain itu, FENCF juga lebih mampu mengatasi item cold start, dengan MAE yang lebih rendah di semua skenario pengujian data. Keunggulan FENCF dalam meningkatkan akurasi rating dan mengatasi item cold start pada data yang kompleks sangat relevan dengan kondisi aktual penjualan buku di e-commerce yang bersifat dinamis. Pada aplikasi nyata, FENCF dapat dengan akurat merekomendasikan buku lama maupun buku baru sesuai preferensi setiap pengguna. Hal ini tidak hanya mendorong pengguna untuk tetap menggunakan platform e-commerce dalam jangka panjang, tetapi juga berpotensi meningkatkan tingkat konversi penjualan.*

***Kata kunci***— *sistem rekomendasi, neural collaborative filtering, item cold start, Sentence BERT, K-means*

***Abstract***

*Book recommender systems often face the challenges of information overload and item cold start due to the dynamics of the evolving book market. This paper proposes Feature Enhanced Neural Collaborative Filtering (FENCF), which is a novel method that combines the interaction between users and items with genre metadata information to address the item cold start problem and improve the accuracy of rating predictions. The uniqueness of FENCF lies in the preprocessing of metadata genres, which is different from typical book recommendation research. Experiments with the Amazon book dataset show the contribution of FENCF, which outperforms NCF by reducing RMSE by 4.04% and MAE by 2.73%. In addition, FENCF is also better able to cope with item cold start, with lower MAE across all testing data scenarios. The advantages of FENCF in improving rating accuracy and overcoming item cold start on complex data are very relevant to the actual condition of book sales in e-commerce, which is dynamic. In real-world applications, FENCF can accurately recommend old and new books according to each user's preference. This not only encourages users to stay with the e-commerce platform in the long run but also has the potential to increase the conversion rate of sales.*

***Keywords***— *recommender system, neural collaborative filtering, item cold start, Sentence BERT, K-means*

*Received December 14th,2024; Revised January 20th, 2025; Accepted January 26th, 2025*

## 1. INTRODUCTION

The development of the Internet has accelerated in recent years, indicating that humanity has entered the era of big data [1]. In this era, there is an exponential increase in data, which not only provides vast amounts of information but also causes information overload problems. Information overload occurs on various platforms, including e-commerce, where users struggle to find products that match their preferences.

E-commerce refers to using electronic media and the Internet to trade goods and services [2], including various types of products, one of which is books. Book sales, especially on large e-commerce sites such as Amazon, are often updated rapidly due to the publication of new works and growing reader interest, making the book market dynamic with collections that cater to the needs of diverse consumers. This growth generates large amounts of complex data, including information about products, transactions, user preferences, and more. One of the biggest challenges of handling large and complex data is how to process and filter the information to make it relevant to the needs and preferences of each user. The solution to overcome this is by utilizing a recommendation system. recommendation system is a software engine that can be used to provide a product recommendation tailored to the preferences of each user [3]. The recommender system plays an important role in recommending items based on each user's preferences. However, when what is recommended by the system is too much or not relevant, the impact is that users will have difficulty in choosing the right book. This can lead to decreased satisfaction, potentially reducing the sales conversion rate.

In previous studies, the methods most often used for e-commerce recommendations, especially for Amazon books, include Collaborative Filtering and hybrid approaches. Margaris et al. [4] proposed the Confidence-Aware Collaborative Filtering (CACF) method with MAE results greater than 0.8 for the Amazon book dataset. In addition, Kharroubi et al. [5] proposed the Item Share Propagation for Link Ranking (ISpLR) method which was compared with several comparison methods (Item based, slope one predictors for online rating, incremental SVD, user-item clustering, and HSI) and found that the method proposed by the author is the best method with MAE results for the Amazon book dataset less than 0.80. Meanwhile, Gao et al. [6] proposed the Automated Collaborative Filtering (AutoCF) method, which they compared with single models (MF, FISM, GMF, MLP, DMF, JNCF-Dot, JNCF-Cat, CMF) and fused models (SVD++, NeuMF, DELF, SinBestFuse), and they found that the proposed method was the best, with RMSE below 0.90 and MAE below 0.80.

In addition, Addanki et al. [7] proposed a hybrid recommender system by combining pre-processing (debiasing) and post-processing (preference correction) applied to several recommendation algorithms (User KNN, Item KNN, ALS, and SVD), it was found that recommendations with the preference correction phase for Amazon books were best applied to the Item KNN algorithm, with RMSE results below 0.90 and MAE below 0.70. Moreover, Karabila et al. [8] proposed a hybrid recommender system by combining the CF method with sentiment analysis (Glove + Bi-LSTM), which they compared with other sentiment analysis methods (TF-IDF + SVM) they found that the proposed method was the best, with RMSE and MAE results less than 1.10.

In previous studies, researchers have successfully developed a recommender system that performs well. However, its accuracy still has the potential to be improved. One approach is to utilize deep learning methods with high accuracy, such as Neural Collaborative Filtering. Neural Collaborative Filtering uses a combined architecture of Generalized Matrix Factorization (GMF) that can handle linear data and Multi-Layer Perceptron (MLP) neural networks that can overcome the limitations of traditional methods in handling more complex data and capturing non-linear data. However, the Neural Collaborative Filtering method has disadvantages, one of which is less able to handle cold start items because it only uses user and item interactions. cold start items are conditions when the system has difficulty recommending products because they include new products that lack ratings [9].

In this paper, we propose the Feature Enhanced Neural Collaborative Filtering (FENCF) method, a modification of the Neural Collaborative Filtering method. FENCF is a method that utilizes genre metadata information as an item attribute, which not only relies on user and item interaction but also utilizes the content features of the item itself. By utilizing the content features of items, the recommender system can recommend new items with limited user interaction and no rating history from users. Thus, the recommender system can overcome item cold start and improve the accuracy of e-commerce recommender systems compared to conventional Neural Collaborative Filtering. The features of the Amazon book dataset used in this study include user ID, book ID, rating, and genre, providing important information about users and recommended items. This study evaluates its effectiveness using the RMSE (Root Mean Square Error) and MAE (Mean Absolute Error) metrics to prove its effectiveness, which demonstrates that FENCF outperforms the conventional NCF method. FENCF contribution as a new, more accurate approach to book recommendation is to produce a higher quality and more relevant recommender system to handle the dynamic nature of actual book sales on e-commerce platforms. FENCF is not only accurate in recommending old books but also in recommending new books with no ratings or only a few ratings. The advantage of FENCF, which uses genre metadata to improve accuracy and overcome the cold start problem, is that it increases user satisfaction with the system. The system's reliability in recommending old and new books based on each user's preferences will make users feel that the platform understands their needs, which, in turn, encourages users to use the e-commerce platform more frequently in the long run. In addition, a more accurate system can reduce the search time to find relevant books and speed up the purchase process. This improvement will result in a greater chance of increasing sales conversion.

## 2. METHODOLOGY

### *2.1 Dataset*

The dataset used in this research is from the book domain of Amazon, one of the largest e-commerce platforms. The book domain was chosen due to its dynamic nature, where books are rapidly evolving products, causing new books to appear frequently, leaving many books without user ratings. This characteristic demonstrates that the proposed FENCF method more effectively overcomes the item cold start problem than other methods. In addition, this study uses a dataset of books from Amazon because Amazon is the largest e-commerce platform with large and complex data. This demonstrates that the FENCF method is superior in rating prediction accuracy on large datasets compared to other methods. The Amazon book dataset used in this study came from Kaggle [10]. This dataset consists of two types of files. The first is a book rating file with 3,000,000 lines, and the second is a book data file with 212,404 lines. The book rating file includes features such as ID, title, user ID, profile name, review help, review score, review time, review summary, and review text. The explicit ratings in the book rating file range from 1.0 to 5.0, indicating the varying levels of user feedback . The book data file includes features such as title, description, author, image, preview link, publisher, publication date, info link, category (genre), and number of ratings. The dataset has several unique characteristics. The category (genre) feature contains 10,883 unique entries. The book ID feature in the book data file has 212,403 unique entries. The user ID feature in the rating file has 1,008,972 unique entries. The dataset includes a total of 3,000,000 interactions.

### *2.2 Preprocessing*

At this stage, data loading occurs. Next, The dataset will undergo manual inspection or identification to find irrelevant features or additional information that does not affect the results. Features in the book data, such as description, author, image, preview link, publisher, publication date, info link, and number of ratings will be removed. This research focuses only on genre (category) metadata so other book metadata, such as description, author, publisher, publication date, and number of ratings, are removed. This process also removes image features, preview links, and info links because they are additional information that does not affect the results. In the book rating data, the process removes the profile name feature because it is additional information that does not affect the results. The review/help, review/score, review/time, review/summary, and review/text features are removed because this research only requires explicit rating data, which aligns with the goal of improving rating prediction accuracy and addressing item cold start. Next, Next, the process will clean NaN values from the feature and rating data. NaN value cleaning is necessary because these values indicate missing or incomplete data, which can interfere with the recommender system development process. The Amazon book dataset used in this research contains many users who have never rated books or have only rated a few books. This issue indicates that the dataset has a user cold-start problem, where the model struggles to understand the preferences of new users due to a lack of interaction data with products. However, the dataset does not include user metadata. Therefore, users who have never rated a book or have rated fewer than 20 books will be removed. Without user metadata, no additional information is available to address the user cold-start issue. If the system does not address the user cold-start issue, it will negatively impact the model's performance. After preprocessing, this dataset contains 39 unique main genres entries, 74,363 unique Book ID entries, 7,080 unique User ID entries, and 393,941 interactions. The genres (categories) feature in the book data contains too many unique entries, so it needs grouping into main genres. The genre feature has two types of text: single words (e.g., "fiction") and sentences (e.g., "ABAP/4 (computer program language)"). Therefore, the clustering process starts with Sentence BERT and continues with K-means. Sentence BERT initially captures the meaning of the text in a multidimensional space, making it suitable for creating an embedding representation for both words and sentences. K-means is then applied because it is one of the most effective clustering methods, capable of grouping genres based on the semantic proximity between vectors. Genres grouped into several main genres will undergo one-hot encoding. User ID and item ID will be label-encoded, and the system will normalize ratings to a range of 0-1. The system will then divide the data into a training dataset (70%), testing dataset (20%), and validation dataset (10%).

### *2.3 Model Design*

In this section, we will describe the Feature Enhanced Neural Collaborative Filtering (FENCF) process. The FENCF model process will start with processing some input data, especially genre metadata, which will then be fed into the GMF, MLP, and NeuMF processes. The architecture of the FENCF model can be seen in Figure 1.

The input for the genre metadata feature contains only one genre in each field. The genre feature has two types of text: single words (e.g. fiction) and sentences (e.g. ABAP/4 (computer program language)). Therefore, in this study, the genre data is preprocessed using Sentence BERT and K-means. Sentence BERT is used to accurately represent text meaning because it captures the meaning of text in a multidimensional space, making it effective for creating embedding representations to process both single words and sentences. Afterward, K-means is applied to cluster genres, as it is one of the most effective clustering methods available, capable of grouping genres based on semantic proximity between vectors. The optimal number of genre clusters is determined using the Davies-Bouldin Index. This index evaluates clusters by measuring the average similarity of each cluster to the cluster it is most similar to, based on the ratio of intra-cluster distances to inter-cluster distances. Consequently, similar genres are more likely to be grouped within the same cluster. The main genres will be processed using one-hot encoding to produce a main genre vector, while user ID and item ID will be processed using label encoding to generate a user vector and an item vector. Input data for GMF consists of the user vector and item vector, while input data for MLP includes the user vector, item vector, and main genre vector.

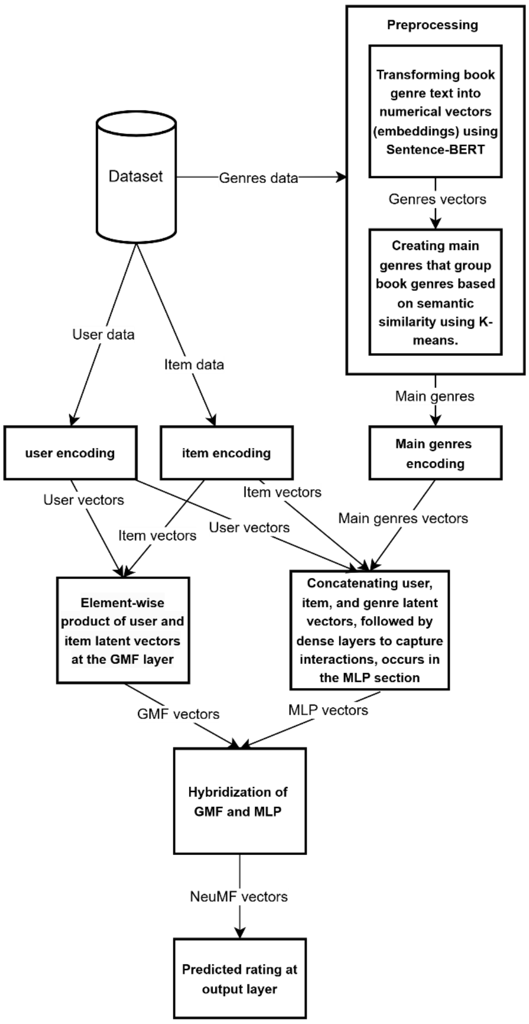
Preprocessing
Transforming book genre text into numerical vectors (embeddings) using Sentence-BERT
Genres vectors
Creating main genres that group book genres based on semantic similarity using K-means.
Dataset
Genres data
User data
Item data
Main genres
user encoding
item encoding
Main genres encoding
User vectors
Item vectors
Item vectors
User vectors
Main genres vectors
Element-wise product of user and item latent vectors at the GMF layer
Concatenating user, item, and genre latent vectors, followed by dense layers to capture interactions, occurs in the MLP section
GMF vectors
MLP vectors
Hybridization of GMF and MLP
NeuMF vectors
Predicted rating at output layer

Figure 1 System Design for FENCF Model

In the GMF process, an embedding dimension is used to represent the input data in a low-dimensional space, which serves to efficiently capture the complex relationships between users and items. The embedding dimension should be adjusted according to the data complexity and dataset size because too low can cause underfitting, while too high can cause overfitting. After going through embedding, the user vector will become the user latent vector and the item vector will become the item latent vector. Then, there will be an element-wise product multiplication process between the user latent vector and the item latent vector that will effectively combine the information of the two latent vectors, resulting in a new vector that describes the interaction between users and items in more detail. The new vector is called the GMF vector. This GMF process has the ability to integrate user and item characteristics linearly, which makes it very efficient in processing large data without losing important information. This approach provides a strong foundation for recommender systems that are not only fast and accurate, but also highly scalable. In addition, there is also the Multi-Layer Perceptron (MLP) process. The Multi-Layer Perceptron (MLP) process contributes greatly to improving prediction capabilities by utilizing genre feature extraction and non-linear interactions between users and items. MLP can capture complex aspects of data that cannot be fully understood by Generalized Matrix Factorization (GMF). MLP also uses embedding dimensions to represent user vectors and item vectors in a low-dimensional space, which serves to efficiently capture complex relationships between users and items and reduce data dimensions for computational savings. After passing through embedding, user latent vectors and item latent vectors will be generated which are combined with main genres vectors, resulting in vectors that are ready to be processed in neural networks. This vector is processed through 3 dense layers with ReLU activation function, which has the advantage of reducing the vanishing gradient problem and speeding up the computation time during training. In ReLU activation if the input $x$ is more than 0 then the output will return the value itself, whereas if the input is less than 0 or 0 then the output is 0. This is explained in Equation 1.

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases} \qquad (1)$$

MLP also employs several regularization techniques, namely dropout, batch normalization, and regularization. Dropout is employed to remove neurons randomly based on a predetermined probability. This technique prevents overfitting and enables the model to learn more diverse feature representations. Then, batch normalization stabilizes the input distribution in each layer, thus accelerating convergence and enhancing training stability. Meanwhile, regularization reduces the complexity of the model by limiting or penalizing overly large weights, thereby lowering the variance of the model without significantly increasing bias. The final output of the MLP after passing through all the dense layers is called the MLP vector.

The output of the GMF process, referred to as the GMF vector, and the output of the MLP process, referred to as the MLP vector, will be combined into a single tensor in the NeuMF process. This combination leverages the advantages of GMF, which focuses on linear interactions, and the advantages of MLP, which focuses on complex non-linear interactions. By integrating both, NeuMF provides more accurate predictions than using only GMF, which struggles to capture non-linear relationships, or only MLP, which is less effective in capturing linear relationships. The combined tensor is then processed through the output layer, producing the final prediction known as the NeuMF vector.

### *2.4 Optimal Hyperparameter Selection*

The selection of the hyperparameter combination to be employed in the FENCF method is an essential aspect. With the optimal hyperparameter combination, the research results are also optimal. The method to find the optimal hyperparameter combination may utilize grid search or random search. In this research, the hyperparameters employed in finding the optimal hyperparameter combination are very diverse and complex. Therefore, this study employs random search rather than grid search. This is because random search explores hyperparameter combinations randomly, which makes it more efficient in handling large and complex hyperparameter spaces. When using grid search for a vast and complex hyperparameter space, the process becomes inefficient as it demands significant time and resources to systematically explore each predefined combination in the grid. Some variations of the hyperparameters used in this study are shown in Table 1.

Table 1 Hyperparameter variations for random search

| Hyperparameter | Hyperparameter tuning options |
|---|---|
| Embedding Dimension | 100, 180, 200 |
| Dropout Rate | 0.2, 0.3, 0.5 |
| Learning Rate | 0.001, 0.003, 0.005 |
| Batch Size | 32, 64, 128, 256 |
| Epochs | 5, 10, 15 |
| MLP Neurons Layer 1 | 3-256 |
| MLP Neurons Layer 2 | 3-256 |
| MLP Neurons Layer 3 | 3-256 |
| Regularizer | 1e-6, 1e-5, 1e-4 |

In the random search, hyperparameter combinations are randomly selected for 15 iterations, and the best hyperparameter combination is selected based on the smallest validation loss. After random search, the optimal hyperparameter combination with the smallest validation loss for the FENCF method is found, namely embedding dimension of 100, dropout rate of 0.2, learning rate of 0.003, batch size of 256, number of epochs of 15, number of neurons in the first layer of Multi-Layer Perceptron (MLP) of 128, number of neurons in the second layer of MLP of 64, number of neurons in the third layer of MLP of 64, and regularizer of 1e-6.

### *2.4 Training*

The FENCF method will be trained with training data to capture interactions between users, items, and genre metadata. The Adam optimizer used in the training process is known to handle sparse or varying gradients in large-scale data. The Adam optimizer will use a learning rate parameter to control how much the optimizer updates the weights during the model training process. A learning rate that is too small results in slow training. A learning rate that is too large prevents convergence or leads to poor results. Therefore, an optimal learning rate is important. In addition to the learning rate, the training process also pays attention to optimizing batch size and epoch. That is because batch size greatly affects the speed of model convergence and memory usage. When the batch size is too small, the model will need more iterations to reach convergence, and when the batch size is too large, the memory may run out quickly and the model may struggle to perform efficient updates. The number of epochs is also crucial to consider as it affects the duration and quality of model training. Too few epochs lead to underfitting because the model has not learned enough from the data, while too many epochs put the model at risk of overfitting. We also use the Mean Squared Error (MSE) loss function, as it is suitable for calculating the mean square difference between predicted and actual values. MSE gives a larger penalty for significant errors, making it suitable for helping the model learn patterns to predict ratings more accurately.

Table 2 shows the results of the training process. At the beginning of training, the training loss value is 0.0156, and it gradually decreases with each epoch. This reduction in training loss reflects that the model successfully learned from the training data by reducing the prediction error. Meanwhile, the validation loss also shows a decrease, although there are some fluctuations. This behavior indicates that the model can generalize the patterns learned from the training data to new data that has not seen before, which indicates an improvement in model performance. The training loss is smaller than the validation loss, which indicates that the method learns well from the training data and does not experience overfitting. Convergence between training loss and validation loss indicates that the model successfully learns important patterns in the data, including the relationship between users and their preferences for certain items in the context so that the model can provide relevant and accurate recommendations.

Table 2 Model training performance

| Epoch | Training loss | Validation loss |
|---|---|---|
| 1 | 0.0156 | 0.0350 |
| 2 | 0.0150 | 0.0343 |
| 3 | 0.0037 | 0.0346 |
| 4 | 0. 0032 | 0.0345 |
| 5 | 0. 0028 | 0.0342 |
| 6 | 0. 0025 | 0.0338 |
| 7 | 0. 0023 | 0.0334 |
| 8 | 0. 0021 | 0.0336 |
| 9 | 0. 0020 | 0.0332 |
| 10 | 0. 0018 | 0.0340 |
| 11 | 0. 0017 | 0.0333 |
| 12 | 0. 0016 | 0.0332 |
| 13 | 0.0015 | 0.0335 |
| 14 | 0. 0015 | 0.0334 |
| 15 | 0. 0014 | 0.0335 |

### *2.6 Evaluation*

We measure the performance of the FENCF model and the comparison model using RMSE (Root Mean Square Error) and MAE (Mean Absolute Error). RMSE employs a quadratic function, penalizing larger residual values more heavily. MAE, on the other hand, calculates the absolute average of the errors by comparing predicted and actual values, thus identifying their differences on an absolute scale. RMSE is defined in Equation (2) and MAE in Equation (3).

$$RMSE = \sqrt{\frac{\Sigma_{ij}\left(R_{ij}-\widehat{R_{ij}}\right)^2}{T}} \qquad (2)$$

$$\text{MAE} = \sqrt{\frac{\Sigma_{ij}\left|R_{ij}-\widehat{R_{ij}}\right|}{T}} \qquad (3)$$

A low MAE indicates high accuracy in the model's prediction of item ratings by users. A low RMSE, on the other hand, indicates that the model is both accurate and consistent in its predictions.

### *2.7 Comparison of Model Performance*

Commonly used methods for comparing model performance with the author's proposed method will be comparatively analyzed. The same dataset and evaluation matrix will be used for both the comparison and proposed models. The description of the selected models is as follows:

- SVD: A commonly used method with matrix factorization in mapping users and items into a latent factor space with good dimensionality [11].
- NMF: A method used to identify latent characteristics between users and items to understand user preference patterns [12].
- Hybrid method (MF+ANN): Combining various methods to obtain a more comprehensive method by utilizing the advantages of each method [13].
- NCF: A neural-based deep learning method that combines GMF and MLP to improve the performance of recommender systems [14].

Comparative analysis evaluation is carried out similarly to the proposed method by dividing the dataset into 70% for training, 20% for testing, and 10% for validation. Furthermore, we also search for the best hyperparameter combination.

## 3. RESULTS AND DISCUSSION

### *3.1 Analysis of Rating Accuracy Results*

The FENCF method, along with a baseline method, will be trained using the hyperparameter tuning results of each method. This research will evaluate the methods using RMSE and MAE. The evaluation results are shown in Table 3 and Figure 2.

Table 3 Comparison results of FENCF and baseline method for rating accuracy

| **Model** | **RMSE** | **MAE** |
|---|---|---|
| NMF | 0.966 | 0.621 |
| SVD | 0.828 | 0.609 |
| MF+ANN | 0.827 | 0.519 |
| NCF | 0.768 | 0.512 |
| FENCF | 0.737 | 0.498 |

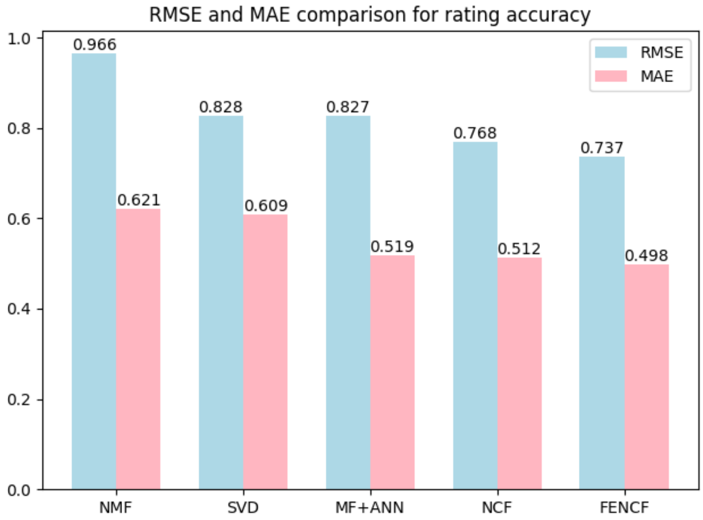

Figure 2 Visualization of rating accuracy from FENCF and baseline method

The rating accuracy evaluation results show that the SVD model performs better than NMF. This can be seen from the SVD evaluation results with RMSE of 0.828 and MAE of 0.609, smaller than the NMF evaluation results with RMSE of 0.966 and MAE of 0.621. The SVD model performs better than NMF because the NMF model is more limited in terms of data representation because the components must be non-negative, while SVD has more freedom to adapt the model to the data, allowing it to produce smaller errors. The prediction accuracy evaluation results also show that the MF+ANN hybrid method is superior to the single methods (SVD and NMF), achieving an RMSE of 0.827 and MAE of 0.519. This superior performance is because MF+ANN combines the advantages of each method to mitigate their respective shortcomings. The NCF method is the best baseline method among other baseline methods with an RMSE of 0.768 and MAE of 0.512. However, the proposed method, namely FENCF, obtains the best RMSE and MAE values with RMSE results of 0.737 and MAE of 0.498. The FENCF model improves rating accuracy by 4.04% in RMSE and 2.73% in MAE compared to the best competitor, NCF. The improved performance of the FENCF model over the baseline method is due to several factors. First, FENCF is superior in understanding complex relationships between users and items, especially non-linear relationships that are difficult to handle by a single method. In addition, FENCF uses hybrid methods, such as GMF, which is a development of MF methods, so it is better able to integrate user and item characteristics linearly, making it more efficient in processing large data without losing important information than MF methods in MF+ANN hybrid methods. Second, the addition of genre as an item attribute in FENCF improves the rating prediction accuracy compared to the NCF method. By including genre metadata, the model utilizes richer information about item characteristics in addition to explicit user-item interactions, enabling it to provide more relevant and accurate recommendations. Overall, FENCF is more effective in improving rating prediction accuracy than the baseline model, resulting in better performance in the recommender system.

### *3.1 Analysis of item cold start results*

We will test the FENCF and NCF methods using several data testing scenarios, namely with data consisting of books that have received ratings of 5, 10, 15, and 20, as well as all test data. We designed these scenarios to reflect different difficulty in dealing with the item cold start problem. In the "All Books" scenario, we include all books regardless of the number of interactions, making it the most challenging as many items have little to no interactions. Scenarios with books that have at least 5 ratings and scenarios with books that have at least 10 ratings still reflect the item cold start condition because the number of interactions is relatively small, so the model must be able to learn patterns with limited data. Conversely, scenarios with at least 15 and at least 20 ratings no longer fully reflect cold start conditions, as at this level the interaction data is more extensive, providing enough information for the model to make more accurate predictions. This approach helps evaluate how each method handles various item cold start or non-cold start conditions. Each scenario will be evaluated using the MAE matrix which can be found in Figure 3.

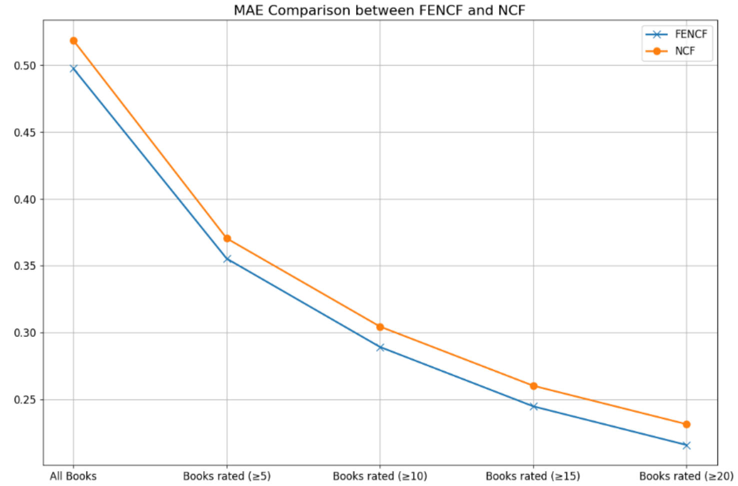

Figure 3 Comparison results of FENCF and NCF for item cold start

Evaluation using the FENCF method in all scenarios, both item cold start and non-cold start resulted in smaller MAE values than conventional NCF, indicating that the modification of the NCF method by adding item metadata is more effective in handling various conditions of item cold start and non-cold start data. This happens because when an item is in a cold-start condition, lacking information from metadata genres, the system has difficulty providing adequate recommendations for new items due to insufficient user interaction. The implication of this finding is highly relevant to real-world applications, particularly in e-commerce platforms that recommend books, which often face the challenge of item cold start. This refers to newly added products that have no ratings or only a few ratings. The FENCF approach can be applied to e-commerce platforms, such as book recommendation, by collecting the genre of each book in the catalog. These genres are processed using NLP Sentence BERT and grouped into main genres through K-means. Main genres are then combined with user-item interaction data to improve recommendation accuracy. New products with little or no rating can still be recommended. This is thanks to genre metadata, which helps the system connect products with user preferences based on genre similarities. With FENCF, e-commerce platforms can provide more relevant recommendations for new and old books, improving user experience and driving sales of new books that were previously hard to find due to lack of ratings.

## 4. CONCLUSIONS

The development of the internet has resulted in more people visiting e-commerce and more new products on a platform, such as Amazon. One of the concerns on Amazon is book recommendation, which often faces the challenge of information overload and item cold start due to the dynamics of the ever-evolving book market. This research proposes the FENCF model which is an extension of NCF by adding genre metadata that has been preprocessed using Sentence BERT and K-means, so thatit is more capable of recommending old and new books more quickly and accurately. This research uses the Amazon book dataset and is evaluated with RMSE and MAE. The evaluation results show that FENCF is superior to the comparison methods (NMF, SVD, MF+ANN, and NCF). FENCF successfully improves the accuracy of rating prediction by reducing RMSE by 4.04% and MAE by 2.73% compared to its best competitor, NCF. In addition, FENCF also proved to be superior for handling item cold start or non-cold start with lower MAE results in all data testing scenarios compared to the NCF method. All FENCF evaluation results show that the proposed model can recommend books more quickly and accurately and is promising because it is better able to handle the dynamic book market with large, complex data and many new products. The capabilities of the proposed model can have a positive impact on e-commerce by increasing user trust in the system, encouraging users to stay with the e-commerce platform in the long term and potentially increasing sales conversion rates.

Future research can improve model performance by experimenting with more diverse hyperparameters, optimizers, and activation functions. In addition, future research can enhance the system by using more book item features, such as year of publication, publisher, author, etc. User features such as age, location, gender, and other similar features can also enrich the available data. By integrating user features and using more book features, the system will be better able to read complex data patterns, identify hidden relationships between features, and generate more accurate and relevant recommendations according to user needs. Handling both item and user cold starts improves item performance. This makes the system more adaptive in handling various problems in usage scenarios.

## ACKNOWLEDGEMENTS

We thank Telkom University for their full support, including resources and facilities, which enabled this research.

## REFERENCES

[1] Yang, X., & Shi, Y. (2020). Self-attention-based group recommendation. 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). https://doi.org/10.1109/itnec48623.2020.9085011

[2] Nahta, R., Meena, Y. K., Gopalani, D., & Chauhan, G. S. (2021). Embedding metadata using deep collaborative filtering to address the cold start problem for the rating prediction task. *Multimedia Tools and Applications*, *80*(12). https://doi.org/10.1007/s11042-021-10529-4

[3] Jena, K. K., Bhoi, S. K., Mallick, C., Jena, S. R., Kumar, R., Long, H. V., & Son, N. T. K. (2022). Neural model based collaborative filtering for movie recommender system. *International Journal of Information Technology (Singapore)*, *14*(4). https://doi.org/10.1007/s41870-022-00858-4

[4] Margaris, D., Vassilakis, C., & Spiliotopoulos, D. (2020). What makes a review a reliable rating in recommender systems? *Information Processing and Management*, *57*(6). https://doi.org/10.1016/j.ipm.2020.102304

[5] Kharroubi, S., Dahmani, Y., & Nouali, O. (2022). Improving collaborative recommendation based on item weight link prediction. *Turkish Journal of Electrical Engineering and Computer Sciences*, *30*(1). https://doi.org/10.3906/elk-2008-26

[6] Gao, C., Yao, Q., Jin, D., & Li, Y. (2021). Efficient Data-specific Model Search for Collaborative Filtering. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. https://doi.org/10.1145/3447548.3467399

[7] Addanki, M., Saraswathi, S., SLAVAKKAM, D. B., Challagundla, R. B., & Pamula, R. (2023). Integrating Sentiment Analysis in Book Recommender System by using Rating Prediction and DBSCAN Algorithm with Hybrid Filtering Technique.

[8] Karabila, I., Darraz, N., El-Ansari, A., Alami, N., & El Mallahi, M. (2023). Enhancing Collaborative Filtering-Based Recommender System Using Sentiment Analysis. *Future Internet*, *15*(7). https://doi.org/10.3390/fi15070235

[9] Panda, D. K., & Ray, S. (2022). Approaches and algorithms to mitigate cold start problems in recommender systems: a systematic literature review. *Journal of Intelligent Information Systems*, *59*(2). https://doi.org/10.1007/s10844-022-00698-5

[10] "Amazon Books Reviews," *www.kaggle.com*. https://www.kaggle.com/datasets/mohamedbakhet/amazon-books-reviews

[11] Koren, Y., & Bell, R. (2021). Advances in Collaborative Filtering - Recommender Systems Handbook. *Recommender Systems Handbook*.

[12] Lee, H. C., Kim, Y. S., & Kim, S. W. (2024). Real-Time Movie Recommendation: Integrating Persona-Based User Modeling with NMF and Deep Neural Networks. *Applied Sciences (Switzerland)*, *14*(3). https://doi.org/10.3390/app14031014

[13] Yu, J., Yin, H., Xia, X., Chen, T., Li, J., & Huang, Z. (2024). Self-Supervised Learning for Recommender Systems: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, *36*(1). https://doi.org/10.1109/TKDE.2023.3282907

[14] Liang, W., Fan, Z., Liang, Y., & Jia, J. (2023). Cross-Attribute Matrix Factorization Model with Shared User Embedding. *arXiv preprint arXiv:2308.07284*.