

# Book Recommendation System Development Using User-Based Collaborative Filtering

Missi Hikmatyar<sup>1\*</sup>, and Ruuhwan<sup>2</sup>

<sup>1</sup> Universitas Perjuangan Tasikmalaya, Tasikmalaya, Indonesia

<sup>2</sup> Universitas Perjuangan Tasikmalaya, Tasikmalaya, Indonesia

\*missi@unper.ac.id

**Abstract.** In general, a library has a Book Search System, both search books by title, author, publisher, and book subject. But with so many book search results displayed by the system, sometimes it makes users (library members) find it difficult to choose books that match their profile. For that, we need a system that can provide recommendations in the book search process to provide recommendations systematically based on previous user searches. This study aims to produce a book search recommendation system in a desktop-based library using the Python programming language and the MySQL database. recommendation system aims to reduce errors in getting the reference books needed. The information that comes out of the search process is right on target with user needs so that it can streamline time in searching for books. The book search recommendation system uses a user-based collaborative filtering method based on the similarity of one member to another member based on the lending pattern which is grouped based on the subject of the book being sought. This system will display the results of book search recommendations by ranking from highest to lowest and provide book title solutions to users according to their profile. The development of the book search recommendation system uses the System Development life cycle (SDLC) method to design a system with a case study research in the TasikmalayaPejuang University library. The first stage of system development is the Identification of Problems that occur in the University of Struggle library following the phenomena that occur in the field. The second stage is the analysis of software requirements, hardware, and information needed in developing a system. The third stage is the design of workflow system design recommendations for book search using Unified Modeling Language (UML), database design and designing the interface. The fourth stage is the implementation of the design results in real form using Python as a programming language and MySql as a database system. The last step is testing the book search recommendation system using the Blackbox testing method by Trial and Error.

## 1. Introduction

The library is fundamental in an educational institution such as a university because of the library as a source of media for storing literature in conducting research and teaching activities. Each book has a relationship with other books in terms of content and references. A large number of books will contain a lot of data information as well, so a search system is needed so that it is easy to search. Book search system based on the title, author, publisher, and book subject still uses the syntax method which is based on the basic search. The use of the syntax method will contain a lot of information and is not effective in searching books that fit the user's needs.



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](https://creativecommons.org/licenses/by/3.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

Educational institutions that have a book search system in a library are still using a system with the syntax method that can cause a buildup of information so that users will long decide on the choice of books to borrow. The problem of information accumulation will have an impact on the effectiveness of the time and quality of references because it can cause human errors in selecting books. A good system is a system that can simplify work and can work more effectively to support work acceleration without causing problems, it is necessary to design a book search recommendation system that can be a solution to problems that occur.

A book search recommendation system is needed to reduce large data output so that book search will be more effective compared to book search system with syntax method. The book search recommendation system uses a user-based collaborative filtering method. the method uses the opinions of a user community to help an individual from the community to find certain information. The purpose of making a book search recommendation system is to make it easier for users to find the right reference according to user needs.

Book search recommendation system has been widely researched including research by Hasibuan [1] developed a book search recommendation system that utilizes the user's search history. The formulation of the problem in this research is the development of library information systems using the User-Based Collaborative Filtering method as a recommendation for searching for books in the library. The Specific Purpose of this Research is to develop a book search recommendation system using User-Based Collaborative Filtering.

## 2. Basic theory

Library Recommendation System is an application model of the results of observations on the circumstances and desires of library members [1], [2]. Library Recommendation System utilizes a person's opinion of a book in a particular category, to help someone in choosing a book [3], [4]. So that the search recommendation system requires an appropriate recommendation model so that what is recommended is following the wishes of the members, and makes it easier for users to make the right decision in determining the book to be chosen [5]. The recommendation system is widely used in various studies such as the game selection recommendation system following the game trends that are often chosen by users [6]. Another recommendation system is in the field of marketing by using a rating system so that it makes it easier for users to select certain products [7]. Book search recommendation system that will be developed using collaborative filtering methods that refer to the needs of users in selecting books [2].

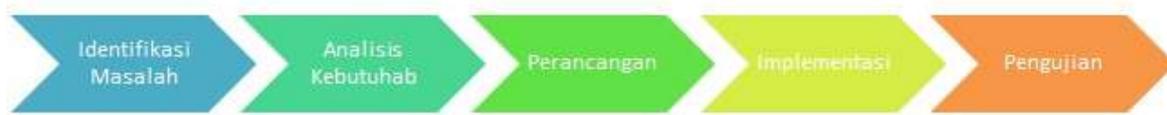
The type of collaborative filtering is a prediction resulting from user choice [8], [9] based on the value of product use (Item-Based) and the level of similarity of user needs (User-based). Collaborative filtering methods are often used in recommendation systems, such as shoe purchase recommendation systems based on product selection ratings [10]. The Collaborative Filtering method is also in the tourism sector which is based on the recommendation of the location of tourist attractions [11]. In the field of marketing public housing loans that help customers in choosing a house [12] In designing a library recommendation system, the modeling process uses the Unified Modeling Language (UML).

Unified Modeling Language is a "language" that has become the industry standard for visualizing, designing and documenting software systems. UML also uses classes and operations in its basic concept, so it is more suitable for writing software in object-oriented languages. library information systems are better described using the UML model because they display the workflow of information systems [13] - [15]. After making the model with UML then do the implementation or making coding.

Implementation of a book search recommendation system to be made based on the desktop with the programming language used is the Python programming language which is more effective in making the program Because the code that is loaded is simpler. Python has been widely used in making systems including the manufacture of laboratory systems [16]. The database is designed for a recommendation system using MySQL (My Structured Query Language) because the MySQL database server can hold up to hundreds of gigabytes of records. MySQL is a database that is widely used in making information systems such as book search systems using data mining [5].

### 3. Methodology

The research workflow is following the method used, namely the System development Life cycle (SDLC).



**Figure 1.** Method of System Development Life Cycle (SDLC)

- **Identification of Problems**  
Stages of problem identification are done in 2 ways: first, conducting interviews related to phenomena that occur in the library environment both the flow of existing systems and the obstacles faced by users when searching books. The object of the interview is the user, employees who work in the library and employees who are in control of the system. Second, make observations on problems that occur to get a hypothesis as information data to be analyzed.
- **Need analysis**  
Stages of needs analysis include 2, namely the analysis of software requirements (software) for implementation needs, and analysis of hardware requirements (hardware) for the needs of researchers. This stage is a tool that supports the development of a system that will be designed following the needs of researchers in solving problems that occur.
- **Design**  
The design phase is the stage of designing a workflow system or Unified modeling language (UML), database and system interface design that will be created. UML design includes the design of use cases and Activity Diagrams created to facilitate implementation. Then design the database using MySQL and design a prototype display interface for each system function.
- **Implementation**  
Stages of implementation are the application of the previous stages to create a whole system. At this stage, Python programming language is used to implement systems that have been designed.
- **Testing**  
The last stage of the research is testing the system that has been implemented. Testing is done using BlackBox testing. Blackbox testing is testing that focuses on functional requirements or the correctness of the inputs and outputs that are generated from software built [17]. Black box testing will be done by giving input from the user to the system that is already running and observing the output of the system. The test will be carried out in each use case to determine the suitability of the function of the software.

## 4. Result

### 4.1. Analysis Of Software Needs

Requirement analysis in research aims to obtain data that is useful to produce a design so that the results of making the program following the points to be achieved and can be developed again as needed.

#### 4.1.1. Analysis of Problems

- **Identify the Cause of the Problem**  
The Struggle University Library currently does not have a system that can be used to make a book recommendation to students so that in making book recommendations it will have a lot of difficulties in managing data and also in providing information about books recommended to each student.
- **Decision Point Identification**

Book recommendation system in the library is needed so that students can easily find and get book recommendations that fit the expected criteria. With the book recommendation system in the library, the library will be helped because it will facilitate the management of book data that will be processed into a book recommendation data for students. Surely it will make time-efficient for the library and students. So from the identification of the causes of the problem a book recommendation system will be developed in the library using a collaborative filtering approach.

- Current System Procedures  
The current system procedure will be illustrated with an Activity Diagram

#### 4.1.2. Analysis Results

From the results of the analysis obtained from the ongoing system procedures to help make a book recommendation that can be used by students, a book recommendation system will be developed in the University of Struggle Library using a collaborative filtering approach.

#### 4.1.3. Proposed System Procedures

The proposed book recommendation system procedure will be illustrated with an Activity Diagram

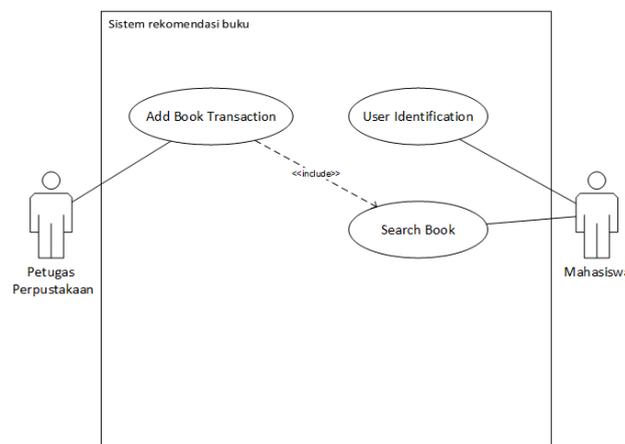
#### 4.1.4. Analysis of System Requirements

- Software Specifications  
The minimum software specifications needed for a book recommendation system in a library to work
- Hardware Specifications  
For the minimum hardware specifications needed so that the book recommendation system in the library can run

#### 4.2. Design

The results of the analysis obtained will be made system design using UML (Unified Modeling Language) which consists of Use Case Diagrams, Activity Diagrams, and Class Diagrams.

- Use Case Diagrams  
The draft use case diagram for developing a book recommendation system in the University of Struggle library is as follows:



**Figure 2.** Use Case Diagram of a Book Recommendation System

- Class Diagram  
The following is a class diagram of the University of Perjuangan library book recommendation system.

### 4.3. Implementation

The library system will be developed by adding features that can provide a book recommendation for students. In developing this system will use the Python 3.7 programming language with a MySQL database and will utilize Kivy and KivyMD as GUI (Graphical User Interface). Making the application code is done on the Windows 10 Pro operating system by using the Visual Studio Code as a text editor and Xampp to run the MySQL service. The method used to obtain book recommendations is the collaborative filtering method and will use centered cosine similarity, cosine similarity, and KNN. In making an application on Windows 10 Pro with Kivy must set the SDL (Simple DirectMedia Layer) version of the program code to SDL2 because Kivy does not support SDL1 on Windows 10 Pro.

#### 4.3.1. Recommendation System.

The method used to create a recommendation system is collaborative filtering and will use centered cosine similarity, cosine similarity, and k nearest neighbors. In giving a book recommendation to students in implementing the program code there are a number of steps that make all the data from borrowing the book into a matrix in the form of an array that will be calculated using an algorithm, in the second stage will be done the process of normalizing the ranking because in making book recommendations using borrowing data the number of books will continue to grow if borrowed by the same student and this makes the ranking scale erratic so that it needs normalization of ranks using centered cosine similarity, after the stage of normalization ranking is complete it will start using the cosine similarity algorithm to get the results of book rating, for the last stage sort the results of the previous calculation from the closest to k nearest neighbors (KNN).

#### 4.3.2. Centered Cosine Similarity

Before doing the normalization process for the first to get data from MySQL with the library MySQL and pandas. At this stage, there is still a NaN value which is caused because in the database there is a column with a NULL value. To overcome this will change the NaN value to 0. The results obtained as follows.

```

      mem_id  id_book  offi_id  receiver
0    2170900001     236      3      3.0
1    2170900001     234      3      3.0
2    1171000003     222      3      4.0
3    1171000003     289      3      4.0
4    1171000003     288      3      4.0
..         ...      ...      ...      ...
899  3190800001    1680      4      0.0
900  3190600012     638      4      0.0
901  3181100001    3073      4      4.0
902  3171100003    2743      4      0.0
903  3171100003    1021      4      0.0

[904 rows x 4 columns]
```

**Figure 3.** Book Transaction Data

Previously obtained data is then normalized with the help of the Pandas library. Here are the results of normalization with the `id_book` column and `mem_id` row.

id_book	2	24	45	62	...	4605	4606	4628	4741
mem_id					...				
1170900004	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1170900005	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1171000001	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1171000002	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1171000003	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1171200001	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1180200001	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1180200002	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1180400001	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5
1180900001	0.0	-0.220779	-0.6	-0.333333	...	0.098039	0.058824	0.166667	0.5

Figure 4. Normalized Results Table

- Cosine Similarity

In this step, the results of previous calculations that have been obtained with Centered Cosine Similarity will then be replaced by the NaN value with the average value of the item. After there is no NaN value, it will proceed with calculations using the Cosine Similarity algorithm. The library needed to calculate cosine similarity is sklearn. To get similar items to use the Cosine Similarity algorithm with the formula:

$$sim(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\|_2 \|\vec{j}\|_2}$$

The results obtained from the implementation of using the Cosine Similarity algorithm using the library sklearn.metrics.pairwise.cosine\_similarity and displayed with library pandas in tabular form are as follows.

mem_id	1170900004	1170900005	...	3190800001	4171200001
mem_id			...		
1170900004	0.000000	0.998022	...	0.999147	0.999443
1170900005	0.998022	0.000000	...	0.998183	0.998479
1171000001	0.996316	0.995357	...	0.996479	0.996773
1171000002	0.998711	0.997747	...	0.998870	0.999167
1171000003	0.999493	0.998529	...	0.999653	0.999950

[5 rows x 147 columns]

Figure 5. Cosine Similarity Results

- K Nearest Neighbors (KNN)

The next stage is to find other members who are closest to the members who want to find book recommendations and rank them. In this study, looking for 10 members who are most similar to members who want to find book recommendations. The results obtained and displayed in tabular form using the Pandas library are as follows.

	ke-1	ke-2	...	ke-9	ke-10
mem_id			...		
1170900004	1171000003	3181000005	...	3180600002	3170900043
1170900005	3170900002	3190300008	...	3180400001	3170900008
1171000001	3170900037	3180900010	...	3170900008	2170900001
1171000002	3170900002	1171000003	...	2180200002	2180300001
1171000003	3171200006	3170900016	...	3190300006	3190300008

[5 rows x 10 columns]

**Figure 6.** Most Similar Member Ranking

For results of book recommendations by (mem\_id 1170900004) can be seen in the image below.

Hasil rekomendasi untuk mem\_id : 1170900004

('Php5 Pemograman Berorientasi Objek Konsep & Implementasi', 'Stendy B. Sakur')  
 ('Chlorella Makanan Kesehatan Global Alami', 'Sukiman Wirosaputro')  
 ('Artificial Intelligence', 'Sri Kusumadewi')  
 ('Kriptografi Keamanan Data dan Komunikasi', 'Dony Ariyus')  
 ('Kecerdasan Buatan', 'Sutajo T')  
 ('Autobioragi Resti Dwi Wahyuni Its My Life', 'Resti Dwi Wahyuni')  
 ('English Sentence Structure', 'Robert Krohn')  
 ('Kimia Organik ; Edisi Ketiga Jilid 2', 'Ralp J. Fessenden')  
 ('Manajemen Sumber Daya Manusia : Human Resource Management Edisi 14', 'Gery Dessler')  
 ('Manajemen Pemasaran ; Edisi 13 Jilid 2', 'Philip Kotler')

**Figure 7.** Book Recommendation Results

## References

- [1] P. R. Hasibuan, A. S. Honggowibowo, and H. Wintolo, "Sistem Pakar Rekomendasi Peminjaman Buku Di Perpustakaan Sekolah Tinggi Teknologi Adisutjipto Yogyakarta Dengan Metode Penyaringan Kolaboratif," *Compiler*, vol. 1, no. 1, 2012.
- [2] M. Irfan, A. D. C, and F. H. R, "Sistem Rekomendasi: Buku Online Dengan Metode Collaborative Filtering," *J. Teknol. Technoscintia*, vol. 7, no. 1, pp. 76–84, 2014.
- [3] M. I. Fathurrahman, D. Nurjanah, and R. Rismala, "Sistem Rekomendasi pada Buku dengan Menggunakan Metode Trust-Aware Recommendation Recommendation System for book by using Trust-Aware Recommendation Method," vol. 4, no. 3, pp. 4966–4977, 2017.
- [4] A. Riyandwyana, E. S. Mukhlason, J. S. Informasi, and F. T. Informasi, "Pengembangan Sistem Rekomendasi Peminjaman Buku Berbasis Web Menggunakan Metode Self Organizing Map Clustering Pada Badan Perpustakaan Dan Kearsipan (BAPERSIP) Provinsi Jawa Timur," *J. Tek. ITS*, vol. 1, no. 1, 2013.
- [5] L. K. Dwi Maryati Suryana, Sri Setyaningsih, "Penerapan association rule mining untuk rekomendasi penelusuran buku dengan algoritma ct-pro," 2015.
- [6] H. Maharani, "Rancangan Sistem Rekomendasi Game Dengan Model-Based Collaboration," 2015.
- [7] C. Saha and D. Prasetya, "Sistem Rekomendasi Pada E-Commerce Menggunakan K-Nearest Neighbor," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 3, pp. 2355–7699, 2017.
- [8] Y. Koren and R. Bell, "Advances in collaborative filtering," *Recomm. Syst. Handbook, Second Ed.*, pp. 77–118, 2015.
- [9] E. A. Laksana, "Collaborative Filtering dan Aplikasinya," *J. Ilm. Teknol. Inf. Terap.*, vol. 1, no. 1, pp. 36–40, 2014.
- [10] A. Kurniawan, "Sistem Rekomendasi Produk Sepatu Dengan Menggunakan Metode

- Collaborative Filtering,” *Semin. Nas. Teknol. Inf. dan Komun.*, vol. 2016, no. Sentika, pp. 2089–9815, 2016.
- [11] A. Arief, Widyawan, and B. Sunafri Hantono, “Rancang Bangun Sistem Rekomendasi Pariwisata Mobile dengan Menggunakan Metode Collaborative Filtering dan Location Based Filtering,” *Jnteti*, vol. 1, no. 3, 2012.
- [12] A. Pamuji, “Sistem Rekomendasi Kredit Perumahan Rakyat Dengan Menggunakan Metode Collaborative Filtering,” vol. 10, no. 1, pp. 1–9, 2017.
- [13] S. Nurajizah, “Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Prototype: Studi Kasus Sekolah Islam Gema Nurani Bekasi,” *Pros. SNIT 2015*, pp. 214–219, 2015.
- [14] D. Luhukay, Y. Kurniawan, and Titan, “Analisis Perancangan Sistem Informasi Perpustakaan Sma Negeri Xyz,” *J. Sist. Inf. Perpust.*, vol. 5, no. 9, pp. 387–396, 2014.
- [15] M. A. J. Plaza, “Sistem Informasi Perpustakaan Di Smp Negeri 12 Kotabumi Berbasis Borland Delphi 7,” *J. Inform.*, vol. 15, no. 2, pp. 102–115, 2015.
- [16] Solekhan, “Sistem informasi laboratorium berbasis jaringan dengan menggunakan python dan mysql,” *J. Sains*, pp. 1–8, 2011.