

# **Content and Collaborative based Sinhala Book Recommendation System**

**P.N.C. Perera**

**2024**





# **Content and Collaborative based Sinhala Book Recommendation System**

**A Thesis Submitted for the Degree of Master of  
Computer Science**

**P. N. C. Perera**

**University of Colombo School of Computing**

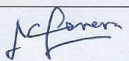
**2023**

# Statement of Declaration

## Declaration

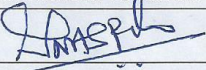
<b>Name of the student:</b> P. N. C. Perera
<b>Registration number:</b> 2019/MCS/067
<b>Name of the Degree Programme:</b> Master of Computer Science
<b>Project/Thesis title:</b> Content and Collaborate based Sinhala Book Recommendation system

1. The project/thesis is my original work and has not been submitted previously for a degree at this or any other University/Institute. To the best of my knowledge, it does not contain any material published or written by another person, except as acknowledged in the text.
2. I understand what plagiarism is, the various types of plagiarism, how to avoid it, what my resources are, who can help me if I am unsure about a research or plagiarism issue, as well as what the consequences are at University of Colombo School of Computing (UCSC) for plagiarism.
3. I understand that ignorance is not an excuse for plagiarism and that I am responsible for clarifying, asking questions and utilizing all available resources in order to educate myself and prevent myself from plagiarizing.
4. I am also aware of the dangers of using online plagiarism checkers and sites that offer essays for sale. I understand that if I use these resources, I am solely responsible for the consequences of my actions.
5. I assure that any work I submit with my name on it will reflect my own ideas and effort. I will properly cite all material that is not my own.
6. I understand that there is no acceptable excuse for committing plagiarism and that doing so is a violation of the Student Code of Conduct.

<b>Signature of the Student</b>	<b>Date (DD/MM/YYYY)</b>
	01/03/2024

## Certified by Supervisor(s)

This is to certify that this project/thesis is based on the work of the above-mentioned student under my/our supervision. The thesis has been prepared according to the format stipulated and is of an acceptable standard.

	<b>Supervisor 1</b>	<b>Supervisor 2</b>	<b>Supervisor 3</b>
<b>Name</b>	Prof MGNAS Fernando		
<b>Signature</b>			
<b>Date</b>	01/03/2024		

I would like to dedicate this thesis to my beloved wife, parents and all readers and authors in  
Sri Lanka

## **ACKNOWLEDGEMENTS**

Completing the Master degree program while working in a stressful environment and fulfilling the duties of a father was a challenging task. It was a great life experience to manage all the tasks without delay. I would like to take this opportunity to all the people behind me and helped me to complete the master's degree program.

First of all, I would like to express my heartfelt gratitude to my supervisor Prof. M.G.N.A.S. Fernando providing guidance, comments, feedback, and vital support throughout the entire research in order to make it a success.

Next, I would like to express my sincere gratitude to famous Authors Dileepa Jayakody, Shamel Jayakody and Nethindu Warapitiya for the help and support provided by evaluating the system and sharing their expertise knowledge with feedback to enhance the application.

Furthermore, I would like to thank all the book readers in Facebook groups who helped me to collect the dataset by filling a google form. Without the dataset I would not be able to complete the research as the model was implemented based on the dataset which is not available in online dataset providers like Kaggle.

Finally, I would like to thank my beloved wife, fellow colleagues, and all the lecturers at the University of Colombo, School of Computing for their help and support throughout the research.

## ABSTRACT

The saying "Reading makes a man perfect" tells us how important it is to gain knowledge by reading different things like books, articles on the internet, newspapers, magazines, or even simple pieces of paper. This article talks about how reading a lot can have a big impact on people. It says that people who read regularly usually have a better understanding of life, make smarter choices, and handle difficult situations better.

The traditional way of picking books, like choosing randomly or going by recommendations, can be hard, especially as libraries have more and more books. The COVID-19 pandemic and distance can make it even harder for people to find the books they want.

This situation shows that there is a need for a better way for people to borrow books from the library. While many online platforms use systems that suggest products, these usually focus on making more sales and not necessarily on what the user really likes.

This article talks about the lack of a system that suggests Sinhala books and suggests a new idea. The proposed solution combines both content and collaboration methods to create a hybrid model for recommending Sinhala books. This system aims to help online users find interesting and relevant books without wasting too much time and money. It not only benefits readers but also gives useful information to authors, helping them understand what readers like and adjust their writing to match popular interests. This recommendation system has the potential to change the way Sinhala books are discovered and chosen, making it a useful tool for both readers and authors in the digital age.

**Keywords:** Sinhala Books, Collaborative filtering, Content based filtering, Hybrid model, Recommendation system

# TABLE OF CONTENTS

ACKNOWLEDGMENT.....	iii
ABSTRACT.....	iv
TABLE OF CONTENT.....	v
LIST OF FIGURES.....	vi
LIST OF TABLES.....	vii
LIST OF ABBREVIATIONS.....	viii
CHAPTER 1.....	1
INTRODUCTION.....	1
1.1 Chapter Overview.....	1
1.2 Background.....	1
1.3 Motivation.....	2
1.4 Statement of the problem.....	3
1.5 Research Aim and Objective.....	3
1.5.1 Aim.....	3
1.5.2 Objective.....	4
1.6 Scope.....	4
1.7 Resource Requirement.....	5
1.7.1 Hardware Requirement.....	5
1.7.2 Software Requirement.....	6
1.8 Structure of the Thesis.....	6
1.8.1 Chapter 02: Literature Survey.....	6
1.8.2 Chapter 03: Methodology.....	6
1.8.3 Chapter 04: Implementation.....	6
1.8.4 Chapter 05: Evaluation and Results.....	7
1.8.5 Chapter 06: Conclusion and Future work.....	7
1.9 Chapter Summary.....	7
CHAPTER 2.....	8
LITERATURE REVIEW.....	8
2.1 Chapter Overview.....	8
2.2 Problem Domain.....	8
2.2.1 Natural Language Processing.....	8
2.2.1.1 Sentiment Analysis.....	9
2.2.1.2 Machine Translation.....	9
2.2.1.3 Named Entry Recognition.....	9
2.2.1.4 Spam Detection.....	9
2.2.1.5 Grammatical Error Correction.....	9
2.2.2 Machine Learning Models.....	10
2.2.2.1 Supervised.....	10
2.2.2.1.1 Regression.....	11
2.2.2.1.2 Classification.....	11
2.2.2.2 Unsupervised.....	11
2.3 Literature Review.....	11
2.4 Chapter Summary.....	16
CHAPTER 3.....	17
METHODOLOGY.....	17
3.1 Chapter Overview.....	17
3.2 Software Design Approach.....	17
3.3 Data Set.....	17
3.3.1 Validate Data Set.....	19
3.3.2 Summary of the Data Set.....	19
3.3.2.1 Gender.....	19
3.3.2.2 Age Range.....	20
3.3.3 Format Data Set.....	20
3.4 Architectural Diagram.....	21
3.5 Machine Learning Models.....	23
3.5.1 Collaborative Filter.....	23
3.5.2 Content-based Filter.....	23



3.5.3	Hybrid Approach.....	23
3.6	Web Application.....	24
3.7	Technology Selection .....	24
3.8	Preprocessing.....	25
3.8.1	Convert Sinhala review to English .....	25
3.8.2	Sentiment Analysis .....	26
3.8.2.1	Using Libraries .....	27
3.8.2.1.1	VADER - Valence Aware Dictionary and sEntiment Reasoner .....	27
3.8.2.1.2	TextBlob .....	27
3.8.2.1.3	Compare VADER and TextBlob .....	28
3.8.2.2	Using Own Mechanism .....	28
3.8.2.2.1	Convert Uppercase to Lowercase .....	29
3.8.2.2.2	Remove Links .....	29
3.8.2.2.3	Remove Punctuations .....	29
3.8.2.2.4	Remove Numbers .....	29
3.8.2.2.5	Remove Stop words .....	30
3.8.2.2.6	Apply Stemming .....	30
3.8.2.2.7	Build Vocabulary .....	31
3.8.2.2.8	Vectorization.....	31
3.8.2.2.9	Model Training and Evaluation .....	34
3.8.2.2.10	Logistic Regression.....	34
3.8.2.3	Get Sentiment Analysis Rate .....	34
3.9	Collaboration based filter .....	35
3.10	Content based filter.....	38
3.11	Web Application.....	40
3.11.1	User Interface .....	40
3.11.2	Authenticate.....	46
3.12	Database .....	46
3.13	Chapter Summary .....	47
CHAPTER 4.....		48
EVALUATION AND RESULTS .....		48
4.1	Chapter Overview.....	48
4.2	Evaluation Metrics.....	48
4.2.1	Accuracy Metrics.....	48
4.2.1.1	Precision .....	48
4.2.1.2	Recall.....	48
4.2.1.3	F1 Score.....	49
4.2.2	User Engagement Metrics .....	49
4.2.2.1	Click Through Rate (CTR) .....	49
4.2.2.2	Conversion Rate .....	50
4.2.2.3	Bounce Rate .....	50
4.2.3	Diversity Metrics .....	50
4.2.4	User Feedback .....	50
4.2.5	Online Evaluation.....	50
4.2.6	Benchmarking .....	50
4.2.7	Mean Absolute Error (MAE).....	50
4.3	Test Results .....	52
4.3.1	Self-Evaluation .....	52
4.3.1.1	Verification of Functional Requirements .....	53
4.3.2	Qualitative Evaluation .....	53
4.3.2.1	Feedback from Domain experts - Famous authors .....	54
4.3.2.2	Feedback from Technical and industry experts .....	58
4.3.2.3	Feedback from Book readers.....	59
4.3.3	Quantitative Evaluation .....	59
4.3.3.1	System Calculation .....	59
4.3.3.1.1	Collaborative Filter Evaluation.....	59
4.3.3.1.2	Content based Filter Evaluation.....	60
4.3.3.1.3	Artificial Neural Network (ANN) .....	61
4.3.4	Online Survey.....	61
4.4	Chapter Summary .....	62
CHAPTER 5.....		63
CONCLUSION AND FUTURE WORK .....		63

5.1	Chapter Overview .....	63
5.2	Conclusion .....	63
5.3	Challenges and Solutions.....	64
5.4	Limitation .....	64
6.1	Future Enhancements .....	65
6.2	Chapter Summary .....	65
Appendix – A .....		I
Appendix - B .....		III
Appendix - C .....		IX

## LIST OF FIGURES

<i>Figure 1: Questions Users ask from groups</i>	2
<i>Figure 2 : Machine Learning Models</i>	10
<i>Figure 3: precision of different algorithm</i>	13
<i>Figure 4 : Gender wise book readers</i>	19
<i>Figure 5: Age wise Book readers</i>	20
<i>Figure 6: Collected data sample</i>	20
<i>Figure 7: Expected format to apply algorithms</i>	20
<i>Figure 8: Architecture Diagram</i>	21
<i>Figure 9 : Review after converting to English</i>	25
<i>Figure 10: VADER result</i>	27
<i>Figure 11: TextBlob result</i>	27
<i>Figure 12: Kindle Review Data sample</i>	28
<i>Figure 13 : The way how the stemming is applied</i>	31
<i>Figure 14: Build the vocabulary</i>	31
<i>Figure 15 : Vectorization 02</i>	31
<i>Figure 16 : Vectorization 01</i>	31
<i>Figure 17 : Balance Dataset</i>	33
<i>Figure 18 : All models related to recommend system</i>	35
<i>Figure 19 : User based Collaborative filter</i>	36
<i>Figure 20 : Item based collaborative filter</i>	36
<i>Figure 21 : User vs Rate matrix</i>	37
<i>Figure 22 : User vs book rate matrix</i>	37
<i>Figure 23 : Collaborative filter result</i>	38
<i>Figure 24 : Top Rated Books</i>	43
<i>Figure 25 : Login Page</i>	43
<i>Figure 26 : Popular Book List</i>	44
<i>Figure 27 : List of books user can select</i>	44
<i>Figure 28 : Recommended Books</i>	45
<i>Figure 29 : Selected book details with reviews and rate</i>	45
<i>Figure 30 : User stored data</i>	46
<i>Figure 31 : Saved book details</i>	46
<i>Figure 32 : Email Confirmation - Dileepa Jayakody</i>	55
<i>Figure 33 : Email Confirmation - Shamel Jayakody</i>	56
<i>Figure 34 : Email Confirmation - Nethindu Warapitiya</i>	57
<i>Figure 35 : Facebook respond after sharing the YouTube link of the implements system</i>	57
<i>Figure 36 : MAE Formula</i>	59
<i>Figure 37 : Invoke the MAE method for all data</i>	60

## LIST OF TABLES

<i>Table 1: Google form with fields details .....</i>	<i>18</i>
<i>Table 2: Technology Stack .....</i>	<i>24</i>
<i>Table 3 : Functional Requirements .....</i>	<i>53</i>
<i>Table 4 : Famous Authors list contacted .....</i>	<i>54</i>
<i>Table 5 : Feedback from Technical experts .....</i>	<i>58</i>
<i>Table 6 : Feedback from Normal Book Readers .....</i>	<i>59</i>
<i>Table 7 : Accuracy calculated for selected book .....</i>	<i>61</i>
<i>Table 8 : Feedback of online survey .....</i>	<i>62</i>

## LIST OF ABBREVIATIONS

Term	Definition
AI	Artificial Intelligence
ANN	Artificial Neural Network
CBF	Content Based Filter
CF	Collaborative Filter
CTR	Click Through Rate
GUI	Graphical User Interface
IDE	Integrated Development Environment
KNN	K Nearest Neighbors
MAE	Mean Absolute Error
MS	Microsoft
NLP	Natural Language Processing
OS	Operating System
RMSE	Root Mean Square Error
RS	Recommender System
SDLC	Software Development Life Cycle
SVD	Singular Value Decomposition
UI/UX	User Interface and User Experience

# CHAPTER 1

## INTRODUCTION

### 1.1 Chapter Overview

This chapter provides a foreword for the project in terms of background study, problem domain, the main aim, objectives, scope and activities that will be carried out towards the completion of the research. Finally, the chapter concludes with an overview on how the other chapters of the document fit into the project context.

### 1.2 Background

From our childhood, everyone has heard that “Reading makes a man perfect”. People acquire the knowledge by reading a variety of materials. These materials could be a book, an internet article, a newspaper, a magazine, or even a piece of paper, and the gain knowledge by reading these materials is intense. People who read a lot tend to know more about life and are smarter when making decisions and handling difficult situations. (Marappan, 2022) It may not be possible for the reader to “know it all,” but a lot of reading brings man close to perfection. Most of them like to read books as a hobby because it imagines readers' own movie in their mind rather than watching a movie directed by someone.

In today’s world, time has more value and the researchers have no much time to spend on searching for the right articles according to their research domain. (Murali et al., 2019)

Book readers usually select books by reading some random pages or asking someone to recommended any book. When reading that book, if he finds that the book is not interesting, he will not read any book after that. therefore, it is better to suggest books that he is interested in. With the increase in library collections, it is difficult for readers to quickly find the books they want. It is also difficult for readers to find Sinhala books of interest in a short period of time in the face of various bibliographies. Therefore, the user experience of the traditional library borrowing method is poor.(Dhanda and Verma, 2016) Due to the Covid-19 pandemic situation and the geographical barriers also it becomes a tremendous challenge for readers (Sarma et al., 2021) to find a relevant book

as they do not like to go out and spend time searching books of their preference. Even the pandemic period is over it is better to be prepared to face such situation in future.

### 1.3 Motivation

When we navigate through social media like Facebook there are so many groups available for almost everything. If you are living in an area, there is a group for that area, if you have an aqua car, there is a group created for aqua car owners. The benefits of such group are you can learn many things and if you have any question you can ask from the group and get it clarified. For Sinhala book readers also, there are so many groups available in the Facebook. You can share what your thought on a specific book or you can see what are latest books released through the groups if you follow those groups. One thing I have noticed is many people asking I have read this particular book; can anyone suggest similar type of books. And some asking what are the books related to Sri Lankan history or related to world war. Some other have asked whether the book is good to read by uploading an image of the book. When considering these three scenarios I thought like it is better to have a system that displays what other users' thoughts about a book and how much of rate could be given to the book and what are other similar books. It gives the motivation to initiate kind of such system for Sinhala book readers.

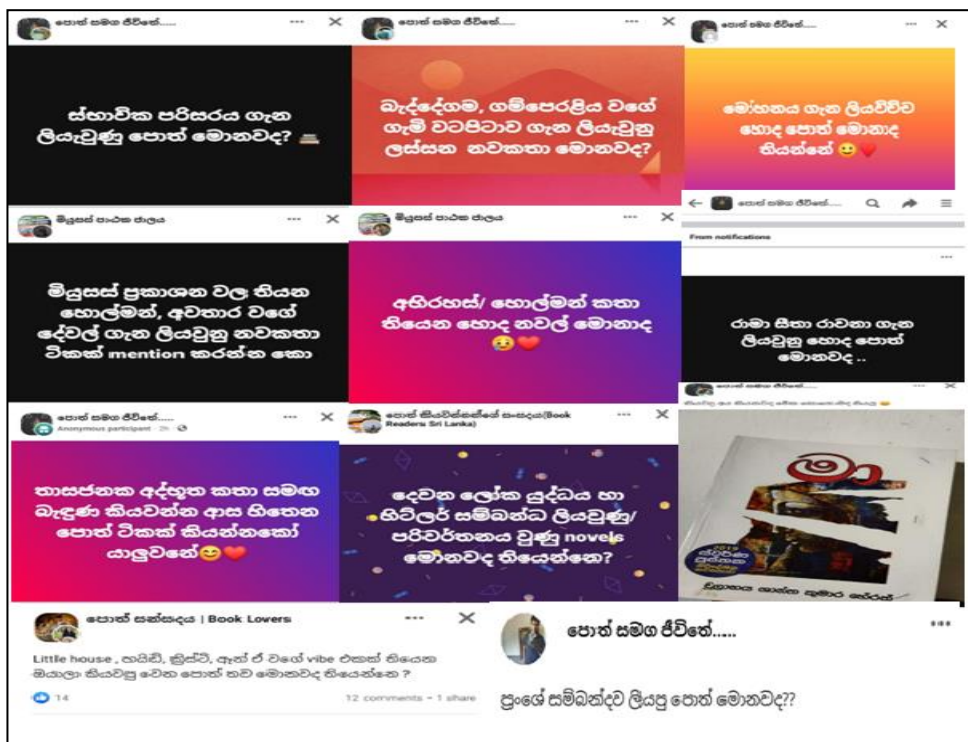


Figure 1: Questions Users ask from groups

## **1.4 Statement of the problem**

Most organizations like Amazon, Ebay have implemented their recommendation system when users buy products online. But almost all the websites are not developed for the buyer's interest; the organizations force add-on sales to buyers by recommending unnecessary and irrelevant products (Sarma et al., 2021) If book recommendation point of view for an instance, if a user has read a book named 'Madol Duwa', he would like to read similar books and there is no Sinhala book recommendation system to address this problem. Additionally, some members of readers groups on Facebook have problems like, is this book good or I have read this book and are there any similar kind of books like the mentioned book?

Many personal book recommendation systems have emerged to conduct effective search based on user rating and interest.

This paper proposed an effective Sinhala book recommendation system for online users that rated a book list using the content and collaboration (hybrid) method. The solution could be used by all Sinhala book readers to find interesting or suitable books without wasting time or money. Authors also could use the system to have an idea of what kind of books readers rate and are interested more and write books accordingly.

## **1.5 Research Aim and Objective**

### **1.5.1 Aim**

The main aim of the research is to analyze, design, implement and evaluate an accurate recommendation system related to Sinhala books using content and collaboration algorithms with attractive user-friendly interface which display the searched book details along with already given reviews and suggesting a list of recommended books. The main aim can be further divided to three sub aims as;

- Input Data will be a dataset collected from readers.
- Preprocess collected data set by removing null values and unwanted data then apply the content and collaborative algorithm respectively
- Combine Content based list and collaborate based list and finally output data will be displayed in the system.



### **1.5.2 Objective**

The final outcome of this project is helping Sinhala book readers to find correct and recommend books based on their preferences using content and collaboration algorithm. There is no exact formula for determining how much data would be enough for a recommendation system. Even though capturing many data sets, ends up with manageable data sets after preprocessing and removing null values from the collection list as most online dataset contains parsed data.

The main goal of this research will be achieved by targeting the following objectives.

1. To Collect selected Sinhala book details like title, author, publisher, description, image url and keywords. Online book stores will be used for collecting the details of books.
2. To produce a data set containing user details, selected Sinhala books and rates given by users for those books.
3. To implement web application along with login and registration features.
4. To recommend ten books based on specific field of interest using Content and Collaborative (Hybrid) methodology.
5. To determine the categories preferred by readers so that it motivates authors to write books as per the user's preference.
6. To Increase the number of book readers by recommending books according to their preferences.

### **1.6 Scope**

The scope of the project can be defined as bellow.

1. Sharing a google form containing selected books along with authors and collecting user rates and reviews for those books based on previous readers experience.
2. Applying sentimental analysis for the reviews collected from above and assigning a new rate.

3. Provide facilities for readers to login or register to the system with their email id which is unique.
4. Provide facilities for readers to search a specific book in the repository and it will display the details such as author, publisher and image along with the reviews and rates given by users.
5. Additionally, the system will display a list of recommended Sinhala books based on users' rates and reviews of specific interested field. Content based and Collaborative based (Hybrid) approach will be taken place in order to recommend books.
6. Only Sinhala books are recommended and it is based on users' rates and reviews as well as keywords provided for selected books

We can use library to collect book details, but we will not be able to collect user reviews and rates for selected books. I am using online book store to collect book details but I preferred to get user reviews and rates from users themselves so that they are aware that they have provided the information for the application rather than just coping from the online without their awareness. 4200 records have been collected so far and they will be used for the research as the data set is considerable amount of data for a machine learning application.

## **1.7 Resource Requirement**

In order to implement and execute the application, following hardware and software requirements should be satisfied.

### **1.7.1 Hardware Requirement**

- A Laptop or desktop with core i3 or above processor
- At least 4GB Ram
- At least 30GB

## **1.7.2 Software Requirement**

- Python latest version – 3.12.0
- VS code as IDE for implementation and execute the application
- MS Excel and Notepad ++ for viewing and manipulating data
- Stable internet connection for downloading relevant libraries.
- GitHub for storing images and implemented code

## **1.8 Structure of the Thesis**

The outline of all the chapters are as follows.

### **1.8.1 Chapter 02: Literature Survey**

This chapter will discuss about the review, conducted on the proposed project. It will extensively describe on the stakeholders, the problem, existing solutions, methodologies, and approaches along with their benefits and limitations.

### **1.8.2 Chapter 03: Methodology**

This chapter will discuss about the methodology to be used to implement the solution. The stakeholders, main technology, libraries, prioritized items, how the collected data is analyzed and the how the architecture of the system will be organized will be in detailed discussed. Furthermore, why the selected technology is more suitable than other existing technologies will be clarified.

### **1.8.3 Chapter 04: Implementation**

This chapter covers the implementation stage of the project. Algorithms used and challenges faced and how they are resolved will be discussed in this phrase. Screen shots and code segments for some selected functionalities are also provided to facilitate easier understanding and manipulating over the project implementation.

### **1.8.4 Chapter 05: Evaluation and Results**

The evaluation chapter provides how the results are evaluated based on the feedback collected from Domain experts in this projects Authors. The project will be shown to them and get the feedback for the evaluation. Other than that validation methods will be used to further evaluate the accuracy of the system.

### **1.8.5 Chapter 06: Conclusion and Future work**

The objectives that were able to be successfully achieved will be discussed in conclusion chapter. The challengers and the limitation of implemented system will be highlighted in order for someone to enhance the system.

## **1.9 Chapter Summary**

The chapter began with explanation on background and problem domain of the system. Although many applications have been developed for book recommending systems, most of them are related to English books. Proper applications that satisfy all the requirements with user satisfaction were limited. The main approach is to make an application that help all Sinhala book readers to recommend Sinhala books based on their preference. A goal followed by objectives was defined to make the effort to be success.

# **CHAPTER 2**

## **LITERATURE REVIEW**

### **2.1 Chapter Overview**

This main aim of the chapter is, study the existing systems implement for Book recommendation system and find out the limitations. As per the study there are two main models that can be used for the system named as Collaboration filtering and Content based filtering. The chapter concludes by explaining the hybrid model which is the combinations of Content and collaborate filter.

### **2.2 Problem Domain**

In today's world recommendation systems plays significant role for user to find items which they prefer. When you buy any product, it suggests similar items or items which customers buy along with the item you bought. When it comes to book recommendation it help readers to find similar books or books read by other users who has similar preference as you. There are multiple recommendation systems have been implemented for English book. Implement a recommendation system for Sinhala books is kind of a challenge as there are no dataset can be found in many datasets provides. Following are the key research areas to be focused in order to complete the application successfully.

#### **2.2.1 Natural Language Processing**

Natural Language Processing refers to a branch of Artificial Intelligence (AI) and it gives computers the ability to understand text as well as spoken words which basically human language and act upon commands. NLP has existed for more than 50 years and has roots in the field of linguistics ("What is Natural Language Processing?," n.d.). As you all know 'Siri' in Apple utilize NLP to respond

There are NLP based applications available as follows which understand human text and voice and help computer to make sense of what it to be performed.

### **2.2.1.1 Sentiment Analysis**

This is the process of determining the sentiment or emotional behind a text. As an example, if there are items with reviews, the algorithm can be used to determine how many of reviews given is positive, negative or neutral. It helps to increase the productivity and quality of the item.

### **2.2.1.2 Machine Translation**

The process of translating to different language automatically without human intervention. The input is from a different language and the translated to expected language as output. Google Translate is the main widely available technology of NLP which helps users to communicate without language barrier.

### **2.2.1.3 Named Entry Recognition**

The main aim of Named Entry Recognition is to extract phrases in a piece of text into predefined categories such as locations, personal names, organizations and quantities. The input of the model takes as text and the output will be the various named entities with their start and end positions.

### **2.2.1.4 Spam Detection**

It is not believable that Spam detection could be implemented via NLP technology. But it is identified the best spam detection technologies use text classification capabilities of NLP to scan emails that often indicates spam or phishing. Spam detectors takes email text as input along with other parameters like title, company name, senders name and find they are spam and placed to a specific spam folder

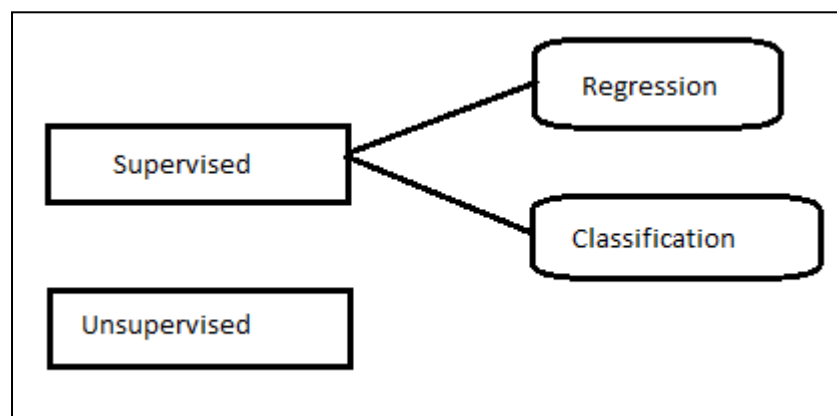
### **2.2.1.5 Grammatical Error Correction**

Grammatical error correction model encodes grammatical rules to correct the grammar within text. Most popular word processing systems like MS Word and Online grammar

checkers like Grammarly use these kind of systems to provide a better writing experience to their users.

## 2.2.2 Machine Learning Models

Machine learning model is a program which find a pattern from a dataset and make decisions. It helps to train the machine to a model and get the output for a given input and behave like a human but in fastest way. Many models are available in the world which helps human to perform many activities such as NLP, image recognition, NLP recognize the sentence and categorize the while image recognition identify objects like car, dog, computer. The machine learning model perform above NLP and image recognition with train the machine with large amount of dataset. While training the algorithm used to find a pattern or results from the dataset being provided. The output or the pattern usually called as machine learning model. All machine learning models are break down into two main categories as supervised and unsupervised. Supervise model further categorized as regression and classification.



*Figure 2 : Machine Learning Models*

### 2.2.2.1 Supervised

In this model machine is trained with labelled data which means some input data tagged with proper output. After train the model the machine will predict the out for any input data provide. The training data input to the machine work as a supervisor who teaches the machine to predict the output. It can be used to real world applications such as image classification, risk assessment and spam filtering.

### **2.2.2.1.1 Regression**

If there is a relationship between input field and output field, regression algorithm can be used. The algorithm is well supported to predict continuous fields like market trends, whether forecasting. Linear regression, non-linear regression, regression trees, polynomial regression are some of regression algorithms.

### **2.2.2.1.2 Classification**

When the output variable can be categorized which means there are two main classes like, yes-no, male-female, true-false, the classification algorithm can be used. Random forest, Decision tree, Logistic regression and support vector machine are some of classification algorithms.

### **2.2.2.2 Unsupervised**

On the other hand, unsupervised learning is a machine learning techniques models are not able to used supervised simply labelled data. In this model, it needs to find hidden patterns by itself from the data provided. The model needs to be trained with unlabeled data and act without any supervision. The unsupervised model cannot be applied to regression or classification problem as we just have input data without output data. The main goal is to find any structure of the dataset, group them based on similarities and apply an algorithm to find similar items. K-means clustering, K-nearest neighbors, Neural Network, Apriori are some of algorithm of unsupervised machine learning.

## **2.3 Literature Review**

According to the research (Sarma et al., 2021) they proposed an effective system for recommending books to online users that used the clustering approach to rate books and then found book's similarity to suggest new books. The data set were collected from Good readers book repository of Kaggle for the research. Based on the classifier they removed books that could be boring books for readers. To measure distance and determine similarity between book groups, the suggested system uses the K-means Cosine Distance function and the Cosine Similarity function. This study presented a



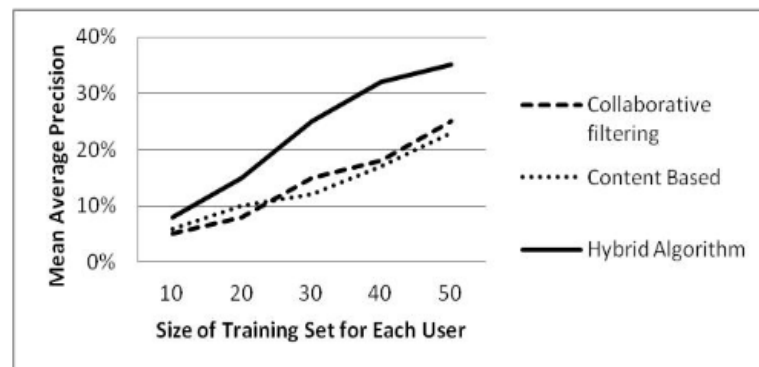
clustering-based book suggestion framework that utilizes various methodologies including collaborative filtering, hybrid, content-based, knowledge based, and utility-based filtering in order to achieve the highest accuracy. Since the accuracy is a crucial aspect of evaluation, they calculate precision, sensitivity, specificity and F1 score and according to the value they have evaluate the system. To display the Graphical view of the accuracy, receiver operating characteristic (ROC) curve was plotted. Further they will propose a system for recommending online courses using the technology Convolutional Neural Network (CNN)

The research (Wadikar et al., 2020) proposes a platform that employs a Convolutional Neural Network (CNN) to recommend books based on two approaches. First approach is, using text processing and the second one is using image classification. In text processing approach, it takes the input from the user as a text and process it. The required data set were taken by performing web scrapped from websites like Amazon and Flipkart and processed separately then converted to csv files. In image classification, a book cover image needs to be uploaded and the results are displayed accordingly. The book cover images data set were taken also using web scrapped. They use cosine similarity measure to find the similar books related to the subject or image from the sites. The researches have tried to improvised and modified the traditional recommending system and filtering techniques like content based or collaborative based have not been used in the system. They conducted the experiment to list the similar books from Amazon and Flipkart. The highlighted advantages of the application are feature engineering is not needed, it gives best results for unstructured data, No need of labelling data, efficient at delivering high quality results, fast access of books that are highly rated and purchased and finally based on recent ratings, it allows smart search. But the evaluation and validation process have not been presented in the research.

The study (Ijaz, n.d.) propose how to use machine learning algorithms K-Nearest Neighbor and matrix factorization for the recommendation system. It first gathers the rankings or a preference of books provided by multiple users and then suggests books to different individuals based on various previous tastes and preferences. K-Means Multipathing together with K-Nearest Neighbor is applied on the BX dataset which are collected from the Kaggle official website to achieve the greatest-optimized outcome. To calculate the accuracy of the system predictions it used an ordinary statistical metric named root mean square error (RMSQ). RMSE is a measurement of the variation

between the user's real books ratings and the predicted rating for the same books. If the RMSE has lower value, the more acceptable the model. An RMSE of zero means the model is absolutely guess the user ratings.

The research (Tian et al., 2019) is one of the best found while reading the literature for recommendation system and it designs a personalized recommendation system for college library based on hybrid recommendation algorithm which combines both collaborative filtering and content based filtering. According to the algorithm it first classified the readers, then establish user-item scoring matrix, then construct vector space model and finally calculate the similarity among users. The experimental data were collected from Library of Inner Mongolia University of Technology. Since the sparsity is a common problem in Collaborative filtering, the research use clustering to alleviate it. In order to verify the effectiveness of the system it performs the calculation of precision for single algorithm and hybrid algorithm respectively and compared. Below is the precision score calculated as per the dataset size for each approach.



*Figure 3: precision of different algorithm*

The research (Shah, 2019) implemented an application for e-commerce and it explain the algorithm collaborative filtering with memory based and model based. A user can either enter rate or sentence which ultimately calculate rate by determining the nature of the sentence using natural language processing. The paper discussed the main problems such as Scalability, Sparsity, Security, Cold start and veracity of profile of recommendation system in details. Even the paper discussed various methods that can be used to build a recommendation system they have used clustering, classification, item-based collaboration approach as user-based approach having some issues like the cost for calculating the similarity between each and every user is high and users' behavior changes very often and because of that it needs to reevaluate the model based on users'

new behavior. Further it performed correlation matrix to represent the relationship between each value in the corresponding column and corresponding row. It used “goodbooks10k” dataset in Kaggle for training, python was used to experiment and Mean Absolute Error (MEA) is used to verify the accuracy and determine the quality of the system.

According to (Mercy Milcah Y et al., 2020) they demonstrated a recommendation model that involves Metrix Factorization as a collaborative filtering solution and with further application of artificial intelligence over the previously obtained results from collaborative filtering. The paper presents six types of recommending systems that can be used by user friendly resources or websites or personalized recommending systems. They are collaborative, content based, demographic based, Utility based, Knowledge based and hybrid recommender systems. The case they consider was a book recommendation system that assist users to select appropriate books to read. The technology in this paper let computer to learn from previous experience, thus it trained to recognize patterns via deep learning and Natural Language Processing (NLP). It then adopted to any new use inputs and provide a result that was solved via Artificial Intelligent (AI) which based on learning, reasoning and problem solving. To increase the accuracy of the application, hybrid model was used combining both collaborate filtering and content-based filtering. It first provides a personalized recommend book list using a model based Collaborative filtering method called matrix factorization. Next the book list with context similarity calculated via Lexile score is listed. And this step does not require the ratings and reviews. As a final result it combines both results and displayed to the users. They also addressed the collaborative problems such as sparsity and cold start by combining the system with content based and make it as hybrid.

As per the research (Sallam et al., 2020) they have implemented a book recommendation system using model based and memory based approach which of the approaches belong to collaborative filtering method. When considering memory based, there are two approach user-based and item-based. They have decided to select item-based approach as user-based approach is not easily scalable and sometimes inaccurate. Then they implemented K-nearest neighbors (KNN) calculate the similarities between items. Despite the success implementation of item-based technique, it found some issues like sparsity and scalability. To overcome these problems, they have integrated model-based

approach via Matrix Factorization techniques. There are various matrix factorization models such as Singular Value Decomposition (SVD), Principal Component Analysis (PCA), Probabilistic Matrix Factorization (PMF) and Nonnegative Matrix Factorization (NMF) We use SVD as it is one of the most common and successful matrix factorization techniques used in collaborative filtering. (Sallam et al., 2020).

(Wang et al., 2018) implemented Content based recommend system which gets the information about the scientific article and suggest most appropriate conferences or journals. After deciding the mode of feature acquisition, the content-based filtering approach was used to predict through softmax regression which is more generic approach of logistic regression. It provides two kind of recommendation results. The first method is 'One class' and it recommends only one journal or conference. The other method is 'Three class' and it recommends three candidate journals or conferences. For the evaluation Chi-square, MI and IG are implemented to make comparisons for feature selection.

Collaborative algorithm is the most desired and widely implemented as well as one of most matured algorithms that are available in the industry. It is mainly based on the assumption that users who liked items in the past will like in the future. And also, users would like similar kind of items as they wanted in the past. The approach builds the model based on rating given by other users for a particular book and users past behavior towards the system. One of the drawbacks of this algorithm is that it needs a tremendous amount of real time user data. Other than that sparsity, cold start and scalability are some of limitation of the approach. But user-item scoring matrix and clustering can be used to alleviate the sparsity problem as it allows re grouping all the books based on the rating and user preference datasets.

Content based algorithm is based on description of the item and the profile of the user's preference. It compares various candidate items with the books previously borrowed or rated by the user and the best matching books will be recommended. The method can be used when a new user login to the system and search for a particular book. The according to the category of the book, a recommended list can be displayed. Some of the draw backs are, it filters the entire set of books from the data set based on the content thus it

hinders the performance and it does not help to find out the content quality of the book and it has low accuracy.

Combining any of two types of recommending systems is known as Hybrid recommender system. This is the most demanded method used by many industries as it combines the strength of more than two types of recommending systems while eliminate weaknesses that were there when only one recommended system is used. Since Collaborative based and content-based filtering algorithm having limitations when they used respectively, Hybrid algorithm will be used in proposed system in order to produce efficient and effective book recommendation

Even though several research papers have been published related to book Recommendation system, all of them related to English books and no research paper was found related to Sinhala Book Recommendation.

## **2.4 Chapter Summary**

The literature review chapter contains what are the existing system available along with tools and technologies used in these systems. As explained, most of the system have implemented mainly either collaboration based or content based specially for English books. While implementing these systems, it is highlighted the limitations of those systems so that the limitations can be addressed in proposed system.

# CHAPTER 3

## METHODOLOGY

### 3.1 Chapter Overview

Earlier Literature review chapter helped to identified what are similar systems available in the world and what are the limitations in those applications. The main focus in the chapter is how the problem is analyzed and identify the methodologies that can be used to implement the system. The architectural diagram will be explained along with why hybrid-based application is focused rather than one particular model will be discussed in details. Since the prototype of the proposed system should address the main objectives that were identified in the first chapter it will further discuss the implementation details individually for identified modules. At the end, the decisions taken on the low-level implementation would be discussed.


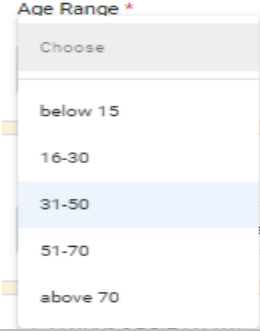

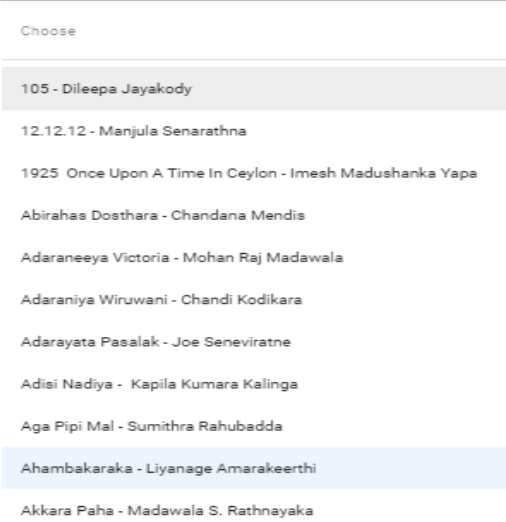
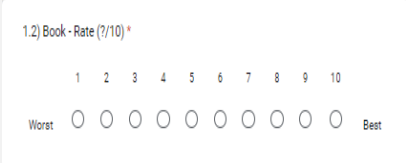
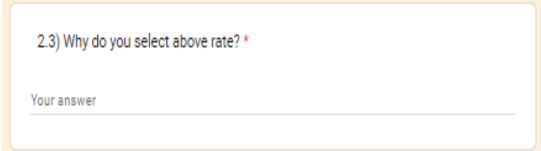
### 3.2 Software Design Approach

Since the requirement is clearly understand and it is not changing every time, waterfall method can be used as a design methodology. The research is conducted based on the method and the progress of each phrase will be explained

### 3.3 Data Set

For any machine learning application, a dataset plays a significant role. While going through some initial process I was trying to find a data set from popular dataset providers like Kaggle. All those data providers have lot of datasets related to book recommendation system. But the limit is all of them related to English books. There were no data set for Sinhala books. Therefore, a google form is created to collect the dataset. The google form is shared in all book readers groups in Facebook and able to collect fair amount of data which can be used to build a model.

Following fields are listed in the form to be provided by the book readers. The full Questionnaire will be shown in Appendix A.

Input Field	Usage	Google form
Email	Unique id to differentiate the user	
Age Range	Drop down list with age range	
Gender	Radio button with two fields of male and female	
Book	This will be a dropdown list which contains 304 books	
Rate	This is a range field from 1 to 10	
Reason	The input will be considered as a review and taken for sentiment analysis	

**Table 1: Google form with fields details**

Initial stage of the implementation, an input text field is given to provide books. But readers entered different data for same value for example, ‘Madol duwa’, ‘Madol Duuwa’. Therefore, a list is created for readers to be selected. In addition, if any

preferred books are not listed, those books can be included at the end of the form so that the books can be added in to the list by admins in future.

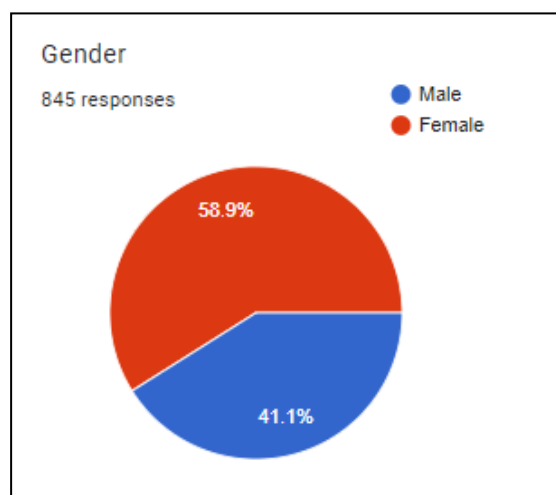
### 3.3.1 Validate Data Set

So far more than 800 users have entered data for the google sheet provided and shared in Facebook groups. In the form there are fields like rate and review. If the user entered higher rate and give positive review, it means the data properly entered. If the user entered higher value and provide a negative review, these rates of the data should be considered by comparing a review rate which could be calculated for the reviews via sentiment analysis method. Then the rate given by the user and the rate calculated via sentiment analysis can be compared and made sure the dataset is suitable for the system.

### 3.3.2 Summary of the Data Set

#### 3.3.2.1 Gender

As per the below diagrams, it shows that most data are entered by female. We can't conclude that female reads more books than male but female responds to the google form than male.



*Figure 4 : Gender wise book readers*



### 3.3.2.2 Age Range

Below diagrams shows the age range of the users read books and contributed for the survey. Age between 16 to 30 and 31 to 50 reads books than other range. Its good to note that young generation moving to read books rather than just surfing in social media. It helps to produce people who not only having the knowledge of the technology but also learning and protect the environment as well as others.

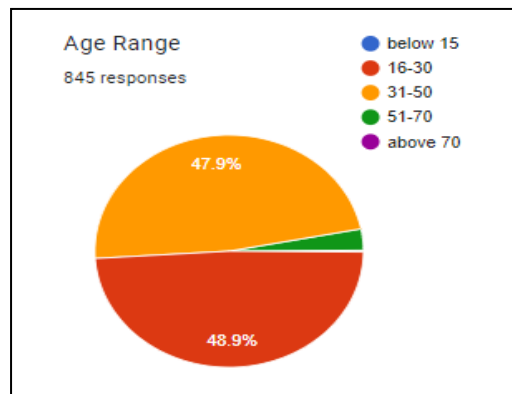


Figure 5: Age wise Book readers

### 3.3.3 Format Data Set

When we analyze the dataset, it contains one user with multiple columns which means the selected books are repeated as bellow.

Ti	Username	Gender	Age Range	1.1) Book	1.2) Book	1.3) Why	2.1) Book	2.2) Book	2.3) Why	3.1) Book	3.2) Book	3.3) Why	4.1) Book	4.2) Book	4.3) Why	5.1) Book	5.2) Book	5.3) Why	6) What are books not listed in the list and
20	nc21perer	Male	31-50	Rana Rala	9	It explains	Lohitha Pa	10	This is on	Adaraneer	9	A great st	Senkottan	10	a story ab	Madol	10	My very first book read and still love to read it.	
20	wrsachith	Female	31-50	Amba Yah	10	Story focu	Haidy - Ch	10	Good boo	Akkara Pa	10	Amazing t	Gamperal	10	I really lik	Sudu V	10	Story is really nice.	

Figure 6: Collected data sample

In Order to apply and algorithm data should be formatted as below. The data set should be preprocessed and python libraries could be used for the below format.

A	B	C	D	E	F	G
Timestamp	Username	Gender	Age Range	Book	Book Rate	Why do you select above rate?
2022/04/3	nc21perer	Male	31-50	Rana Rala	9	It explains how to face problems and win as a t
2022/04/3	nc21perer	Male	31-50	Lohitha Pa	10	This is one of best detective series
2022/04/3	nc21perer	Male	31-50	Adaraneer	9	A great story in a village and our history
2022/04/3	nc21perer	Male	31-50	Senkottan	10	a story about a village and the way it written is
2022/04/3	nc21perer	Male	31-50	Madol Du	10	My very first book read and still love to read it.

Figure 7: Expected format to apply algorithms

### 3.4 Architectural Diagram

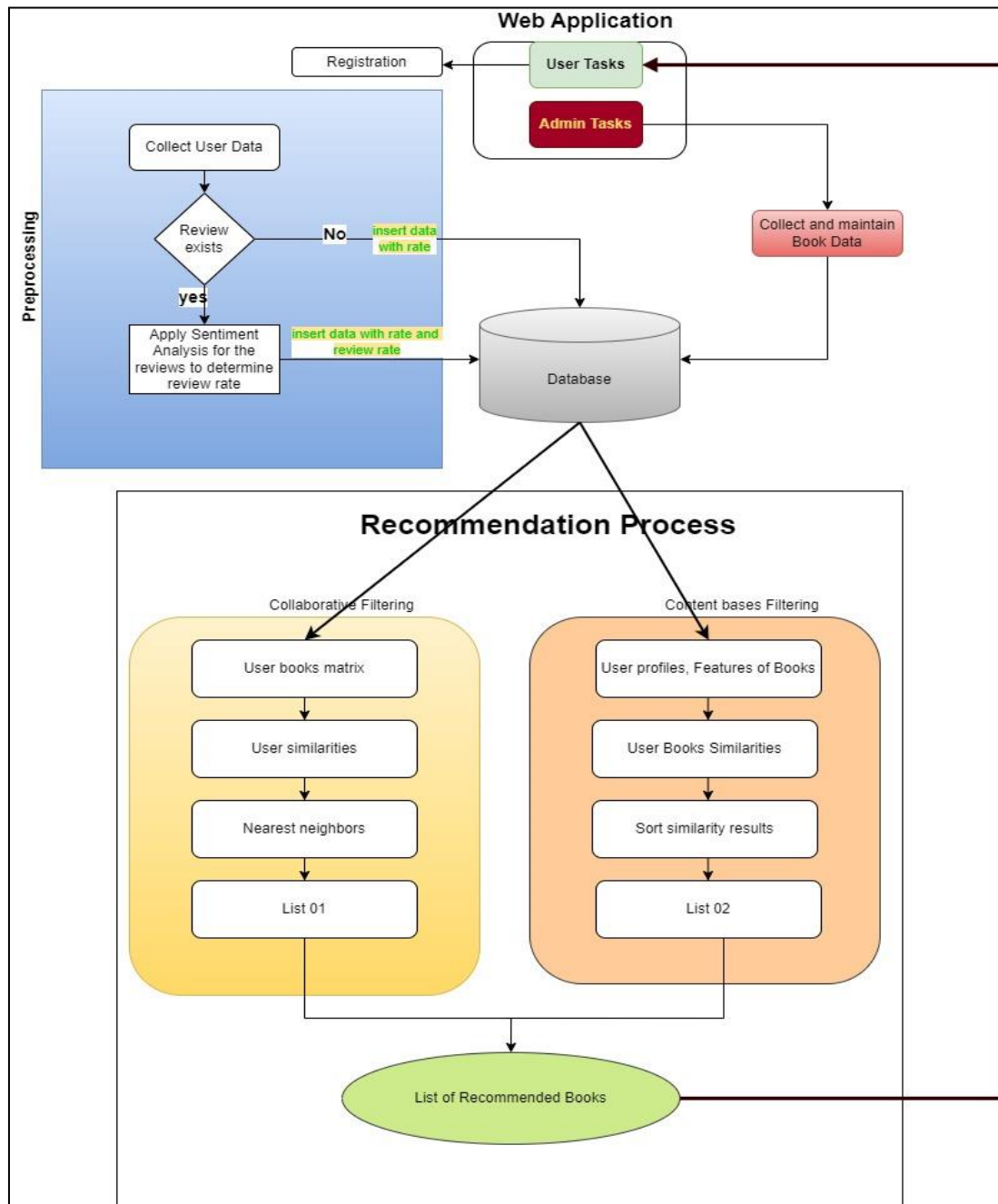


Figure 8: Architecture Diagram

According to the below architectural diagram, initially data is collected and then it needs to be categorized and store in separate 3 tables as Book\_Details, User\_Details and Rate\_Details. All user related information will be stored in User\_Details table. Additional book data like ISBN, publisher, year should be added in Book\_Details table to display the details of the book when a user search for a particular book. Ratings given for

a book by users are stored in Rate\_Details table. If a review exists for a book, a separate review rate will be assigned for the review after completing of the sentimental analysis and include it in Rate\_Details table as a separate column. For the final calculation, mean value of review rate and normal rate will be taken. If there is no review given, normal rate will be taken for the calculation.

Before starting the recommendation process, collected data need to be preprocessed in order to remove unwanted data like null rate values. And data like giving maximum rate for just one and only book should be eliminated as those kinds of books should not be recommended.

Then the process of applying the algorithms will be taken place. As per the paper (Murali et al., 2019) More than 250 research paper recommender systems were published and the quantity of research papers published every day is increasing rapidly. Thus, it needs an efficient searching and filtering mechanism to choose the quality research papers, so that the effort and time of researchers can be saved. (Murali et al., 2019)

One of the main two methods used to implement the system is Collaborate filtering. The method is used to recommend a user with best books in their domain according to the queries and preferences based on the similarities found from other users.(Murali et al., 2019) It will find the adjacent neighbors of a customer based on the ratings given by the other users. In user based collaborative algorithm, first we need to build a matrix upon users and books with the rating given. Then cosine similarity, which is one of the techniques of K Nearest Neighbor (KNN) will be computed for each user in the matrix. The KNN is a machine learning algorithm to find clusters of similar users based on common book ratings. The cosine similarity first collects books in which is evaluated by all the users in the nearest neighbors, and then the candidate list which the target user has rated or reviewed is removed. Finally, a list of recommended books will be generated based on the similarity.

In content-based approach, it is based on the description of the book and a user's preference. The algorithm tries to recommend books which are similar to those that a user rated in the past. Initially it abstracts the features such as title, author, genre of books in the system. Then information such as books user read and the rates given will be considered to create user preference vector. Finally various candidate items are compared

with the books previously rated by the user and the best matching books are recommended.

According to the researches most of them use hybrid method which combines both collaborative filtering and content-based filtering. Even though there are multiple strategies to apply the hybrid method, applying content-based and collaborative filtering separately and then combining them together will be adopted as it is more effective in book recommendation system. Therefore, the common book list which are generated from collaborative filtering and content-based filtering will be displayed as the final recommended book list.

### **3.5 Machine Learning Models**

Even though many applications have been implemented using either collaborate based or content based, few applications were implemented using both. How the methodologies can be used in the system will be elaborate below.

#### **3.5.1 Collaborative Filter**

The data set contains user, book and rate. As the first step, the data set should be converted to used based mastics. Then cosine similarity can be used to find the recommend books.

#### **3.5.2 Content-based Filter**

The data set contains author, review and description. Removing stop words, links and numbers, then combining all together we can create a tag list. Similar books can be selected by applying cosine similarity.

#### **3.5.3 Hybrid Approach**

There are multiple ways we can apply hybrid methodology for the dataset. Finding books separately for collaborate and contend based and check common books is one approach and the other approach is, apply collaborate first and then apply content-based for filtered books. Since the first approach will be more feasible and manageable, it will be used as hybrid model.




### 3.6 Web Application

The web application will be implemented using python flask which is one of library for develop web applications. The basic html with css without any library will not be used as it will be an extra effect to connect with python backend. Mainly register and login page will be there and two mail roles as registered users and admin users will be maintained. Registered users will be able to view the top-rated books, popular books and recommend books where admin uses can maintain book data addition to above features where normal users perform. Top rated books list can be taken from the dataset it self by calculating the highest average rates where popular book list can by taken by calculating the number of rates given for books.

For the recommendation list collaborative and contend based filtering will be applied as hybrid approach and the list will be displayed at the bottom of the page after book meta data and reviews with rate provided by users.

### 3.7 Technology Selection

The main technology along with other related technologies and libraries which will be used for the system is as follows.

Front End	Logic	Data Persistence
		

*Table 2: Technology Stack*

### 3.8 Preprocessing

Every data set provided by many providers like Kaggle, to be preprocessed and captures the data we required for the algorithm. Since the data combines with stop words, numbers, links which do not have proper meaning, these to be removed. But as the first step, all reviews which were written in Sinhala to be converted to English.

#### 3.8.1 Convert Sinhala review to English

Once the google form is shared user is able to enter data in both English and Sinhala. Most data were entered via English but few were entered via Sinhala. Since the data review entered via Sinhala is considerable amount, we are not going to ignore but applying google translator by python on Sinhala reviews can be converted to English so that we can utilize those reviews as well for the machine learning algorithm.

In order to convert the language, Google Translator library could be used as below. Then all the reviews written in Sinhala will be converted to English reviews which ultimately could be applied sentiment analysis on top of the reviews.

```
from googletrans import Translator  
translator = Translator()
```

31-50	Sihina Miyadunaden - Dilhani Wickramaratne	10	ආත්තවම ආදරණීය කතාවක්.. ඒ වගේම පොඩ් දරුවෙක් ගෙ ...	A really lovely story.. and a story of a small...
31-50	Mahathma Gandhi - David Karunaratne	8	ජීවිතයටම ආදර්ශමත් කතාවක්	An exemplary story for life
31-50	Ginigath Sanda - Rohana Weththasnghe	7	දුක තිතෙන කතාවක්..ආදර්ශමත් කතාවක්	A sad story..an exemplary story
31-50	Adaraniya Wiruwani - Chandi Kodikara	10	ආදරණීය කතාවක්.. ජීවිතයේ යතාර්ථයක්	A lovely story.. a reality of life

Figure 9 : Review after converting to English

### 3.8.2 Sentiment Analysis

Most online stores like Amazon, AliExpress, Ebay provide a website for users to express their opinions about different items they bought. Since then, it has been established that buying online, 90% of consumers are testing different websites channels to determine the quality of their purchase. To evaluate the text data and then extract the sentiment element from that the field of sentiment analysis is frequently used. (Wassan et al., 2021) From user ratings, suggestions, recommendations and messages, online business websites produce a massive volume of textual data every day.(Wassan et al., 2021)

Sentiment analysis is the process of analyzing a given text and determine if the text means to positive, negative or neutral. It basically helps to understand the human feelings via text. It is one of the Natural Language Processing (NLP) technique used to analyze the text.

As per the research (Tripathy et al., 2015) it says Sentiment analysis is the most prominent branch of natural language processing and it refers to feelings, attitudes, emotions. Most people used to express their sentiments to others through social media, ratings and reviews. Based on the review and the rate other can determine the quality and usability of a product that is sell over internet. The paper presents the comparison of results that is calculated by applying two algorithms Naïve Bayes and Support Vector Machine (SVM). The calculation was based on the dataset taken from a movie dataset.

According to the study of the research (Chandrasekaran et al., 2022), it build a sentiment analysis model based on images from social media. For the work they used different transfer learning models, including the VGG-19, ResNet50V2, and DenseNet-121 models, to perform sentiment analysis based on images. As a dataset, Twitter-based images available in the Crowdfower dataset were used which contains URLs of images with their sentiment polarities.

There are several ways to apply sentiment analysis for a text. Following are some of them.

### 3.8.2.1 Using Libraries

#### 3.8.2.1.1 VADER - Valence Aware Dictionary and sEntiment Reasoner

It is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. (Caren, 2019) It is available in the NLTK package and can be directly applied to a text and gives both polarity(positive/negative) and intensity or strength. The feature depends on a dictionary which maps lexical features with sentiment score.

```
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
sent_analyzer = SentimentIntensityAnalyzer()
text = "the greatest story"
sentiment = sent_analyzer.polarity_scores(text);
print("Analyser ----", sentiment)
```

```
Analyser ---- {'neg': 0.0, 'neu': 0.323, 'pos': 0.677, 'compound': 0.6369}
```

*Figure 10: VADER result*

#### 3.8.2.1.2 TextBlob

It is another lexicon-based python library which can be used to process a text and gives two main values polarity and subjectivity. Other than sentiment analysis, the library contains lot of features like noun phrase extraction, tokenization, lemmatization, spelling correction. As per the below example the text contains the word ‘greatest’ which textblob consider as the sentiment analyser and return positive value 1.0.

Polarity has the value between -1 to 1 where -1 represents the most negative words like ‘worst’, ‘aweful’, ‘disgusting’ while 1 represents most positive words like ‘the best’, ‘excellent’. Subjectivity lies between 0 to 1 where 0 represent factual information while 1 represent more personal opinion. (Barai, 2021)

```
from textblob import TextBlob
text = "the greatest story"
testimonial = TextBlob(text)
print("textblob -- ", testimonial.sentiment)
```

```
textblob -- Sentiment(polarity=1.0, subjectivity=1.0)
```

*Figure 11: TextBlob result*



### 3.8.2.1.3 Compare VADER and TextBlob

When we check some value in Textblob, it is noted that some text which have more negative values like not and slow, it multiplies -0.5 and -0.3 and gives the polarity of the sentence as a positive value. Another issue of Textblob is, if it finds any negative word in between in a sentence, it gives some polarity other than 0. Due to these issues Textblob could not be considered as one of the best sentiment analyzers.

When the same above sentences check with VADER, it gives better result than Textblob. As per the (Barai, 2021) it compares both analyzers and came to a conclusion that Textblob struggled with negative sentences. The discussion further explained that It is not that VADER is better than Textblob in sentiment analysis. But it works better for negative sentences.

As conclusion, there are drawbacks for both of the analyzers. Therefore, it would be more convenient to implement own mechanism for sentiment analyze and predict the value for a given sentence.

### 3.8.2.2 Using Own Mechanism

There are some limitations when use any library in our system like not able to customized and not able to understand the logic behind the functionality. There for own mechanism of implementing sentimental analysis would be used. Following are the list of main steps for building the model then implement the pipeline for the model built. ("Machine Learning Project | Classification | Sentiment Analysis | Sinhala - YouTube," n.d.) . In order to train the model for sentiment analysis, 'Kindle reviews' dataset was taken from Kaggle. The dataset looks like as below.

```
data = pd.read_csv('kindle_reviews.csv')
data.head()
```

	Unnamed: 0	rating	reviewText	summary
0	0	5	This book was the very first bookmobile book I...	50 + years ago...
1	1	1	When I read the description for this book, I c...	Boring! Boring! Boring!
2	2	5	I just had to edit this review. This book is a...	Wigglesliscious/new toy ready/!!
3	3	5	I don't normally buy 'mystery' novels because ...	Very good read.
4	4	5	This isn't the kind of book I normally read, a...	Great Story!

Figure 12: Kindle Review Data sample

### 3.8.2.2.1 Convert Uppercase to Lowercase

The review text contains uppercase as well as lowercase. As a first step all the characters to be converted to lowercase so the case sensitiveness can be ignored when comparing values.

```
data['reviewText'].apply(lambda x : " ".join(x.lower() for x in x.split()))
```

### 3.8.2.2.2 Remove Links

Since the links do not have any meaning for sentiment analyzer, those links to be removed with below code.

```
data['reviewText'].apply(lambda x: " ".join(re.sub(r'http?:\V.*[\r\n]*', "", x, flags=re.MULTILINE) for x in x.split()))
```

### 3.8.2.2.3 Remove Punctuations

Since the punctuations also do not have any meaning for sentiment analyzer, all the punctuations to be removed with below code. Punctuation list can be found in string library. A function is defined to remove the punctuations and it is invoked in all the review text.

```
def remove_punctuations(text):
    for punctuation in string.punctuation:
        text = text.replace(punctuation, "")
    return text

data['reviewText'] = data['reviewText'].apply(remove_punctuations)
```

### 3.8.2.2.4 Remove Numbers

There were numbers also added in the review test and those were also to be removed as they do not have any meaning for sentiment analyzer process. Removing numbers in a text can be achieved by below code.

```
data['reviewText'].str.replace('\d+', "", regex=True)
```

### 3.8.2.2.5 Remove Stop words

There were number of stop words exists in the review text and those were also to be removed as they do not have any meaning for sentiment analyzer process. The list of stop words can be download from nltk library to a folder specified.

```
nltk.download('stopwords', download_dir='static/model')
```

A variable is defined to store the list of stop words as below.

```
with open('static/model/corpora/stopwords/english', 'r') as file:  
    sw = file.read().splitlines()
```

Finally, the stop words are removed from the review text with following code

```
data['reviewText'].apply(lambda x : " ".join(x for x in x.split() if x not in sw))
```

### 3.8.2.2.6 Apply Stemming

After removing all unnecessary values in the text, the next phrase is converting different verb formats to a common pattern. As an example, write, wrote, written, writing are converted to base form write. This process is called as stemming and the following code snippet will do the conversion.

```
from nltk.stem import PorterStemmer  
ps = PorterStemmer()
```

```
data['reviewText'].apply(lambda x : " ".join(ps.stem(x) for x in x.split()))
```

After completing the preprocessing part for the reviews, the text came up without uppercases, links, punctuations, numbers and stop words. Finally stemming has been applied to convert all the text to their base form. Following depict shows how the conversion has been done up to now.

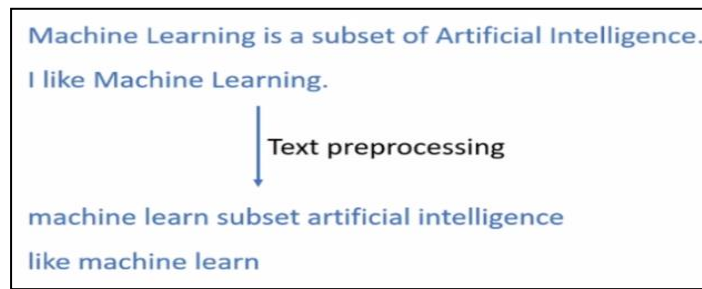


Figure 13 : The way how the stemming is applied

### 3.8.2.2.7 Build Vocabulary

In order to build a model, the machine is not able to read and understand the text and they need to be converted to numerical values. The building the vocabulary is the process of converting the text to appropriate numerical values. As the first step a unique vocabulary set to be created from the converted text. In the above example, following is the list.

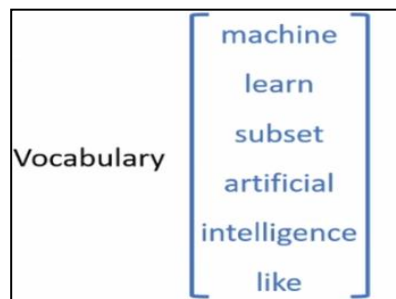


Figure 14: Build the vocabulary

### 3.8.2.2.8 Vectorization

The next step of converting the text to numerical values, is vectorization process. As per the above example, all the sentence could be converted to numerical value which has the length of six (06). The value is same as the length of the vocabulary list. The list contains values which are called as features. In this example there are six features in the vocabulary.

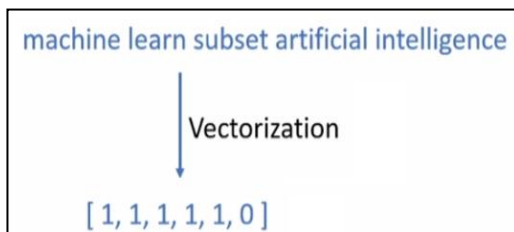


Figure 16 : Vectorization 01

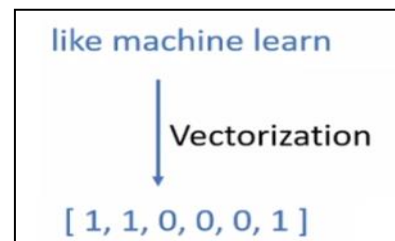


Figure 15 : Vectorization 02

After the process, all the reviews will be converted to a numeric value which have the same length. Then the output can be fed to machine learning model.

Following code will check the size of the vocabulary list simply a number of features. The list contains a unique text and the number of times the text is used in the review.

```
from collections import Counter
vocab = Counter()

for sentence in data['reviewText']:
    vocab.update(sentence.split())

len(vocab)

33599
```

As per the above result, there are 33,599 features found in the reviews. In the scene, all the reviews will be represented as numeric value which has the length of 33,599. But the Kindle review data set contains around 12,000 records. If this much of features are used, the model will be over fit. The number of features should be less than the number of records in order for model to be a good one.

To overcome the issue, the feature selection will be used to reduce the feature count as bellow. Then the feature count is reduced to 3645.

```
tokens = [key for key in vocab if vocab[key] > 30]

len(tokens)

3645
```

The final output of vocabulary list will be saved as bellow

```
def save_vocabulary(lines, filename):
    data = '\n'.join(lines)
    file = open(filename, 'w', encoding='utf-8')
    file.write(data)
    file.close()

save_vocabulary(tokens, './static/model/vocabulary.txt')
```

When an own method of creating a model for sentiment analysis, accuracy plays a significant role. It gives how the model is accurate as percentage.

Before vectorization, the review dataset to be divided to two main parts as training data and test data. The training data will be used to train the model and the test data will be used to test and get the accuracy of the model.

```
x = data['reviewText']
y = data['rating']
```

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size=0.2)
x_train.shape
(9600,)
x_test.shape
(2400,)
```

The vectorization can be done via bellow function.

```
def vectorizer(ds, vocabulary):
    vectorized_list = []

    for sentence in ds:
        sentence_list = np.zeros(len(vocabulary))

        for i in range(len(vocabulary)):
            if vocabulary[i] in sentence.split():
                sentence_list[i]= 1

        vectorized_list.append(sentence_list)

    vectorized_list_new = np.asarray(vectorized_list, dtype=np.float32)

    return vectorized_list_new
```

Then the function will be invoked for both train and test data as follows. Then all train data reviews and test data reviews will be converted to numeric data set.

```
vectorized_x_train = vectorizer(x_train, tokens)
```

```
vectorized_x_test = vectorizer(x_test, tokens)
```

Note how the values of data set are fairly shared (balanced dataset) for each rate as below.

```
: y_train.value_counts()
:
rating
4    2420
5    2367
2    1609
1    1608
3    1596
Name: count, dtype: int64

: y_test.value_counts()
:
rating
5     633
4     580
3     404
1     392
2     391
Name: count, dtype: int64
```

*Figure 17 : Balance Dataset*

### 3.8.2.2.9 Model Training and Evaluation

The next stage of the sentiment analysis process is building a model and evaluation.

```
from sklearn.linear_model import LogisticRegression

from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score

def training_scores(y_act, y_pred):
    acc = round(accuracy_score(y_act, y_pred),3)
    pr = round(precision_score(y_act, y_pred),3)
    rec = round(recall_score(y_act, y_pred),3)
    f1 = round(f1_score(y_act, y_pred),3)
    print(f"Training Scores:\n\tAccuracy = {acc}\n\tPrecision = {pr}\n\tRecall = {rec}\n\tF-Score = {f1}")

def validation_scores(y_act, y_pred):
    acc = round(accuracy_score(y_act, y_pred),3)
    pr = round(precision_score(y_act, y_pred),3)
    rec = round(recall_score(y_act, y_pred),3)
    f1 = round(f1_score(y_act, y_pred),3)
    print(f"Testing Scores:\n\tAccuracy={acc}\n\tPrecision={pr}\n\tRecall={rec}\n\tF-Score = {f1}")
```

### 3.8.2.2.10 Logistic Regression

In order to train the model, we use logistic regression as it has the highest accuracy rate among other classification algorithms like decision Tree, Random Forest, Naïve bayes.

```
lr = LogisticRegression(random_state=0, max_iter=1000)
lr.fit(vectorized_x_train, y_train)
y_train_pred=lr.predict(vectorized_x_train)
y_test_pred=lr.predict(vectorized_x_test)
```

After the model is trained properly, it is saved to a location as below

```
import pickle
with open('./static/model/model.pickle', 'wb') as file:
    pickle.dump(lr, file)
```

### 3.8.2.3 Get Sentiment Analysis Rate

After the model is build, the sentiment analysis value to be calculate for a given text. There for the given test to be preprocessed, vectorized before get the prediction. Following code will invoke the appropriate function and return the predicted value.

```

txt = "I think that story has been fictionalized very interestingly in a fantasy world"
preprocessed_txt = preprocessing(txt)
vectorized_txt = vectorizer(preprocessed_txt,tokens)
prediction = get_prediction(vectorized_txt)
prediction[0]

```

4

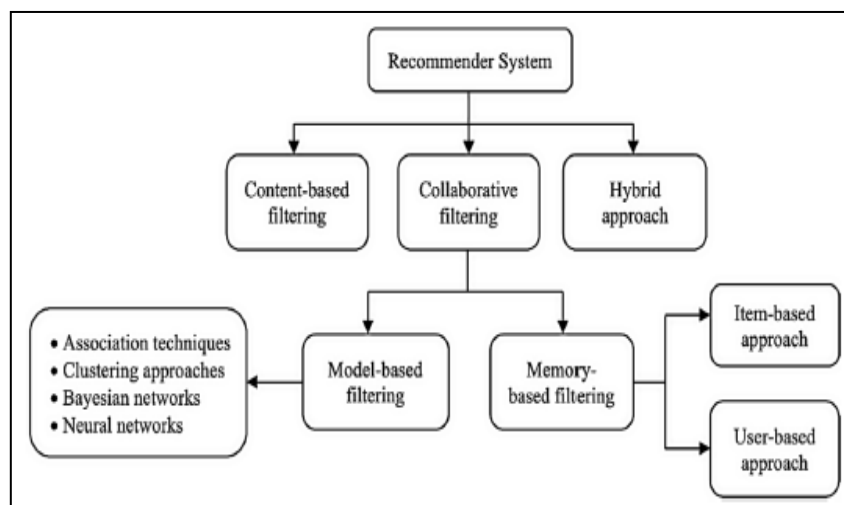
Above result gives positive value 4 out of 5 for the given text. Since the rate calculated for the application is out of 10, it needs to be multiplied by 2 as the review rate.

### 3.9 Collaboration based filter

Over the past decade, collaborative filtering algorithms have evolved from research algorithms intuitively capturing users' preferences to algorithms that meet the performance demands of large commercial applications. (Schafer et al., n.d.)

Collaborative approaches make use of the measure of similarity between users. (Roy and Dutta, 2022) The model starts with finding a group or collection of user Y whose preferences, likes, and dislikes are similar to that of user X. Y is called the neighborhood of X. The new items which are liked by most of the users in Y are then recommended to user X. The accuracy of the approach is depending on how efficiency and accuracy the model can find the similarities of the target user. The main drawback of this algorithm is cold start and privacy concern as the user data has to be shared.

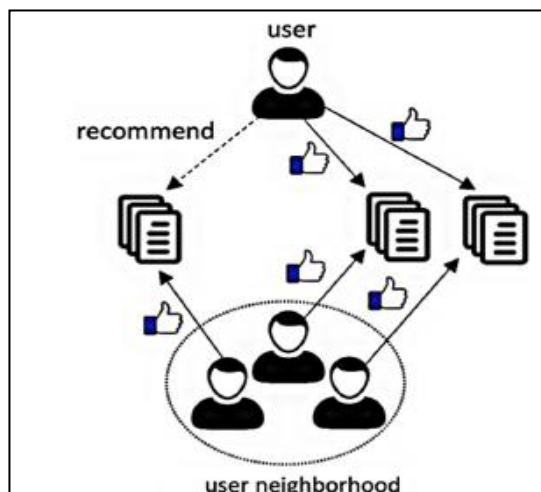
Collaborative approach is divided into two main categories named memory based and model based. Memory based again divided to Item based and User based. Following figure depicts all the approached in recommendation system.



*Figure 18 : All models related to recommend system*

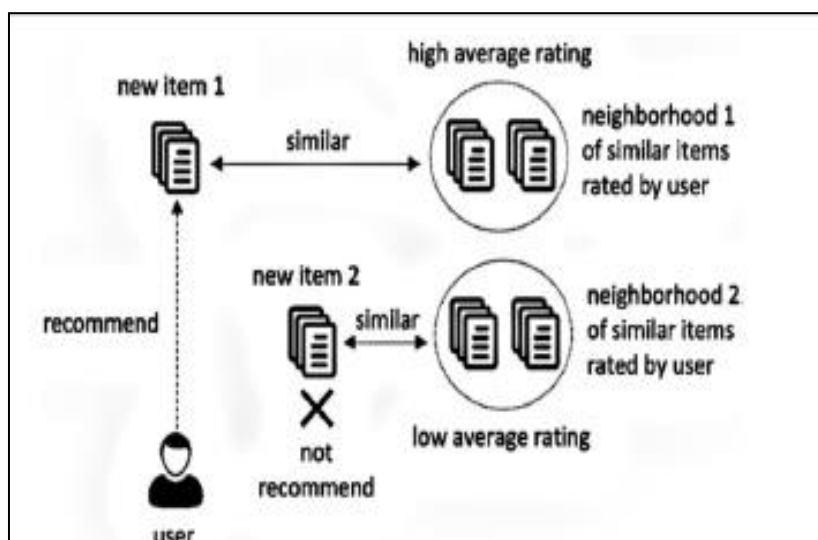


Memory based approach recommend item based on preference of its neighborhood. In this approach to make recommendations for a new user, the user profile must be added to the utility matrix. If the user profile cannot find, then this approach faces cold start issue. In user based approach, the user rating of a new item is calculated by finding other users from the user neighborhood who has previously rated that same item. If a new item receives positive ratings from the user neighborhood, the new item is recommended to the user. Below figure depicts the user-based filtering approach.(Roy and Dutta, 2022)



*Figure 19 : User based Collaborative filter*

In the item-based approach, an item-neighborhood is built consisting of all similar items which the user has rated previously. Then that user's rating for a different new item is predicted by calculating the weighted average of all ratings present in a similar item-neighborhood as shown in below figure. (Roy and Dutta, 2022)



*Figure 20 : Item based collaborative filter*

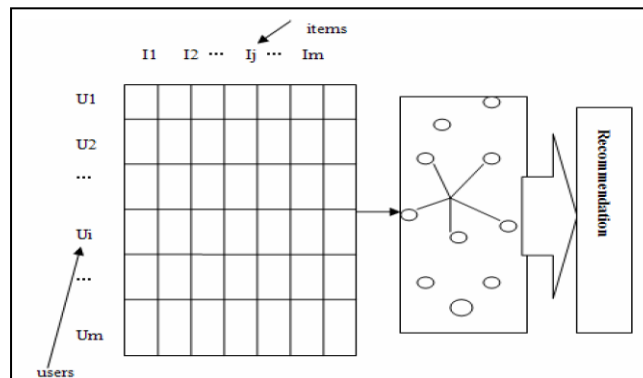
Since Content based filtering cannot discover the quality of an item, collaborative filtering system is used to overcome this problem.

As per the first step, we need to find books that are selected by at least more than 5 users. If no one is selected a book, they should be ignored and should not recommend those books for the users. Below code will remove such books.

```
y = books.groupby('Book').count()['Book Rate'] >=5
famous_books = y[y].index
famous_books.drop_duplicates()
```

```
final_ratings = books[books['Book'].isin(famous_books)]
final_ratings = final_ratings.drop_duplicates()
```

The process of applying the Metrix and how the collaboration filter works is depicted below.



**Figure 21 : User vs Rate matrix**

The code snippet to implement the matrix as below.

```
pt = final_ratings.pivot_table(index='Book', columns='Username', values='Final_Rate')
pt.fillna(0, inplace=True)
pt
```

	Username 00rvnd@gmail.com	12pramodmadusanka@gmail.com	2020ba23889@stu.cmb.ac.lk	88.rudmi@gmail.com	91iahadarshi@gmail.com	99.madhawa@gmail
Book						
105 - Dileepa Jayakody	0.0	0.0	0.0	0.0	0.0	
12.12.12 - Manjula Senarathna	0.0	5.0	9.5	0.0	8.0	
1925 Once Upon A Time In Ceylon - Imesh Madushanka Yapa	0.0	0.0	0.0	0.0	0.0	
Abirahas Dosthara - Chandana Mendis	0.0	0.0	0.0	0.0	0.0	
Adaraneeya Victoria - Mohan Raj Madawala	0.0	0.0	0.0	0.0	10.0	
...	...	...	...	...	...	...
Wassana						

**Figure 22 : User vs book rate matrix**

Finally, applying cosine similarity for the matrix and finding the similar books as below.

```
from sklearn.metrics.pairwise import cosine_similarity
similarity_scores = cosine_similarity(pt)
similarity_scores.shape
```

With below function, we can recommend the books.

```
import numpy as np

def recommend(book_name):
    index = np.where(pt.index==book_name)[0][0]
    similar_items = sorted(list(enumerate(similarity_scores[index])), key=lambda x:x[1], reverse=True)[1:11])

    for i in similar_items:
        print(pt.index[i[0]])
```

Calling the function and getting the book list

```
recommend("Apoiyawa - Mahinda Prasad Masimbula")

Adaraneeya Victoria - Mohan Raj Madawala
Charitha Thunak - K Jayathilake
Ape Gama - Martin wickramasinghe
Senkottan - Mahinda Prasad Masimbula
Amba Yahaluwo - T. B. Illangaratne
Guru Geethaya - Dedigama V. Rodrigo
Amma - Dedigama V. Rodrigo
Amma - Upul Shantha Sannasgala
Manikkawatha - Mahinda Prasad Masimbula
Nil Katrol - Mohan Raj Madawala
```

*Figure 23 : Collaborative filter result*

### 3.10 Content based filter

The Collaboration filter is totally based on the previous data collected by other users. Content-based filtering uses the assumption that items with similar objective features will be rated similarly. (Schafer et al., n.d.) For example, if you liked a web page with the words “tomato sauce,” you will like another web page with the words “tomato sauce.”(Schafer et al., n.d.)

If the user does not have previous data, similar books are not be able to recommended. Even a new book added to the system and it has not been rated, it cannot be recommended. Content based filtering introduced to overcome these problems, even previous data is not found, books can be recommended based on the contents.

For an instance a book contains keywords like ‘sherlock Holmes’, ‘detective’ books which have similar keywords will be recommended. The first task of the process is replacing ‘and’ with a comma (,). It could be done with bellow code snippet.

```
def convert_tags(str):  
    return [x.lower().strip() for x in str.replace('and','').split(',')]
```

```
book_dataFrame['tags'] = book_dataFrame['tags'].apply(convert_tags)  
book_dataFrame.head(2)
```

With below code, all authors can be extracted and save in a different field.

```
def convert_author(str):  
    return str.split('-')[1].lower().strip()
```

Below code will convert Sinhala description to English as the description will be added to the tags.

```
from googletrans import Translator  
translator = Translator()  
  
def translate_description(str):  
    return translator.translate(str, dest="en").text  
  
book_dataFrame['eng_description'] = book_dataFrame['description'].apply(translate_description)
```

Combine all together as tags as below

```
book_dataFrame['all_tags'] = book_dataFrame['tags'] + book_dataFrame['auth_eng']
```

Apply cosine similarity as below

```
from sklearn.metrics.pairwise import cosine_similarity  
content_similarity = cosine_similarity(vectors)
```

Define a function to recommend content similarity.

```
def recommend(book):
    movie_index = new_df[new_df['book_with_author'] == book].index[0]
    distances = content_similarity[movie_index]
    book_list = sorted(list(enumerate(distances)), reverse=True, key=lambda x:x[1])[1:6]

    for i in book_list:
        print(new_df.iloc[i[0]].book_with_author)
    #return
```

Calling the function and getting the book list

```
recommend('Madol Duwa - Martin wickramasinghe')
Ape Gama - Martin wickramasinghe
Kaliyugaya - Martin wickramasinghe
Viragaya - Martin Wickramasinghe
Karuwala Gedara - Martin Wickramasinghe
Amba Yahaluwo - T. B. Illangaratne
```

## 3.11 Web Application

The web application is developed using Flask in python and it contains user interface and the authentication.

### 3.11.1 User Interface

The graphical user interface (GUI) of the web application is implemented using python flask, HTML, CSS and Bootstrap. The main file which defines the routes of the application is as follows.

```
from flask import Flask, render_template, request, redirect, session
import pickle
import numpy as np
import pandas as pd
import mysql.connector
import os

app = Flask(__name__)
app.secret_key=os.urandom(24)

@app.route("/")
def login():
    if 'user_id' in session:
        return redirect('/home')
    else:
        return render_template('login.html')
```

```

@app.route('/register')
def register_ui():
    return render_template('register.html')

@app.route('/logout')
def logout():
    session.pop('user_id')
    return redirect('/')

@app.route('/home')
def home():

    if 'user_id' not in session:
        return redirect('/')
    else :
        try:
            connection = mysql.connector.connect(host="localhost", database="sinhala_book_recommendation",
user="root", password="admin");
            cursor = connection.cursor();
            book_names = []
            book_authors = []
            book_images = []
            book_publishers = []
            listt = topRatedBooks['Book'].values;
            for bookAuthor in listt:
                query = "SELECT * from books where book_with_author = " + bookAuthor + ";"
                cursor.execute(query);
                result = cursor.fetchall();
                if result:

                    bookName = result[0][2];
                    bookAuthor = result[0][3];
                    publisher = result[0][4];
                    url = result[0][5];

                    book_names.append(bookName)
                    book_authors.append(bookAuthor)
                    book_images.append(url)
                    book_publishers.append(publisher)
            connection.commit();
            avg_list = list(topRatedBooks['avg_rating'].values)
            rating_list = [ '%.3f' % elem for elem in avg_list ]

        except Exception as e:
            print("Something went wrong", e);
        finally:
            if connection.is_connected:
                connection.close();

    return render_template('home.html',
        book_name = book_names,
        author = book_authors,
        publisher = book_publishers,
        image = book_images,
        votes=list(topRatedBooks['num_ratings'].values),
        ratings=rating_list
    )

```

```

@app.route('/recommend')
def recommend_ui():
    if 'user_id' not in session:
        return redirect("/")
    else :
        try:
            booklist=[];
            connection = mysql.connector.connect(host="localhost", database="sinhala_book_recommendation",
user="root", password="admin");
            cursor = connection.cursor();
            query = "SELECT id,book_with_author from books;"
            cursor.execute(query);
            results = cursor.fetchall();

            if results:
                for result in results:
                    list=[];
                    id = result[0];
                    bookNameAuthor = result[1];
                    list.append(id);
                    list.append(bookNameAuthor);
                    booklist.append(list);

            # print(booklist)

        except Exception as e:
            print("Something went wrong", e);
        finally:
            if connection.is_connected:
                connection.close();
            return render_template('recommend.html', booklist = booklist)

@app.route('/login')
def login_ui():
    return render_template('login.html')

@app.route('/register_user', methods=['post'])
def register_user():
    loginName = request.form.get('login_name')
    password = request.form.get('password')
    query = "INSERT INTO users VALUES ('" + loginName + "', '" + password + "');"
    try:
        connection = mysql.connector.connect(host="localhost", database="sinhala_book_recommendation",
            user="root", password="admin");
        cursor = connection.cursor();
        cursor.execute(query);
        connection.commit();

    except Exception as e:
        print("Something went wrong", e);
    finally:
        if connection.is_connected:
            connection.close();

    return render_template('login.html')

```

Sinhala Book Recommendation

User Login

Login name

abc

Password

...

Login

Not a member? [Create Account](#)

Figure 25 : Login Page

Top Rated Books

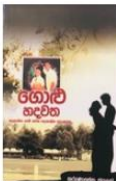







 <p>ගොළු හඳුවන කරුණාසේන ජයලත් කතරා ප්රකාශන</p> <p>Votes - 26</p> <p>Rating - 9.096</p>	 <p>සැබෑ මිනිසෙකුගේ කතාවක් දැදිගම වි රුද්ල කෙසෙලි ප්‍රකාශකයෝ</p> <p>Votes - 20</p> <p>Rating - 8.825</p>	 <p>කුරුලු හඳුවන ලියනගේ අමරකිකරී විදුරන්නා ප්‍රකාශකයෝ</p> <p>Votes - 22</p> <p>Rating - 8.795</p>	 <p>සිව් රහස් සලකුණ චන්දන මෙන්ඩිස් චන්දන මෙන්ඩිස් ප්‍රකාශන</p> <p>Votes - 27</p> <p>Rating - 8.778</p>
 <p>චරිත කුහාක් කේ ජයතිලක කතරා ප්රකාශන</p> <p>Votes - 31</p> <p>Rating - 8.565</p>	 <p>බිහිසුණු නිමිනය චන්දන මෙන්ඩිස් චන්දන මෙන්ඩිස් ප්‍රකාශන</p> <p>Votes - 30</p> <p>Rating - 8.550</p>	 <p>මළගිය ඇත්තෝ එදිරිමිර සරච්චන්ද්‍ර ගොඩගේ ප්‍රකාශකයෝ</p> <p>Votes - 40</p> <p>Rating - 8.512</p>	 <p>චස්සාන සිහිනය උපුල් ශාන්ත සන්නස්ගල සංගීඳ්‍ය ප්‍රකාශකයෝ</p> <p>Votes - 23</p> <p>Rating - 8.500</p>

Figure 24 : Top Rated Books



## Popular Books



**12.12.12**

මංජුල සේනාරත්න

මිනිසුන් සොන් ප්‍රකාශන

Votes - 219

Rating - 7.970



**අම් යහලුවෝ**

එම්. ඉලංගරත්න

සරසවි ප්‍රකාශන

Votes - 141

Rating - 8.110



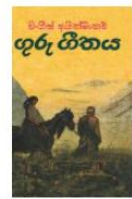
**සෙංකොට්ටන්**

මහින්ද ප්‍රසාද මස්ඉඩුලු

සන්ථව ප්‍රකාශනය

Votes - 127

Rating - 8.280



**ගුරු ගීතය**

දැඩිගම වී රුද්‍රිගු

ප්‍රගති ප්‍රකාශකයෝ

Votes - 98

Rating - 8.270



**ඇමසෝනියා**

මංජුල දිසානායක

මිනිසුන් සොන් ප්‍රකාශන

Votes - 86

Rating - 8.233



**ආදරණීය වික්ටෝරියා**

මොහාන් රාජ් මඩවල

විසේෂ ප්‍රකාශන

Votes - 86

Rating - 7.558



**අසනග වැසි**

දඟරන ගමිණී විජේතිලක

සුසර ප්‍රකාශකයෝ

Votes - 79

Rating - 8.456



**1925**

ඉසෙක් මධුසංක යාපා

මිනිසුන් සොන් ප්‍රකාශන

Votes - 71

Rating - 7.528

Figure 26 : Popular Book List

## Recommend Books

-- Select --

-- Select --

Antharaya Adaviyaka - Chandana Mendis  
 Handa Nihanda - Kumara Siriwardhana  
 Bihisunu Nimnaya - Chandana Mendis  
 Gamperaliya - Martin Wickramasinghe  
 Madol Duwa - Martin wickramasinghe  
 Malagiya Aththo - Ediriweera Sarachchandra  
 Deveni Gahaniya - Manjula Senarathna  
 Charitha Thunak - K Jayathilake  
 Guru Geethaya - Dedigama V. Rodrigo  
 Poliyana - Kathyana Amarasinghe  
 Manikkawatha - Mahinda Prasad Masimbula  
 Anne 01 (Arabe Gedara Anne) - Premasiri Mahingoda  
 Senkottan - Mahinda Prasad Masimbula  
 Amma - Dedigama V. Rodrigo  
 Da Vinci Kethaya - Kumara Siriwardhana  
 Asanaga Wasi - Darshana Shammai Wijethilake  
 Baskerville Shapaya - Chandana Mendis

Submit

Figure 27 : List of books user can select



Figure 29 : Selected book details with reviews and rate

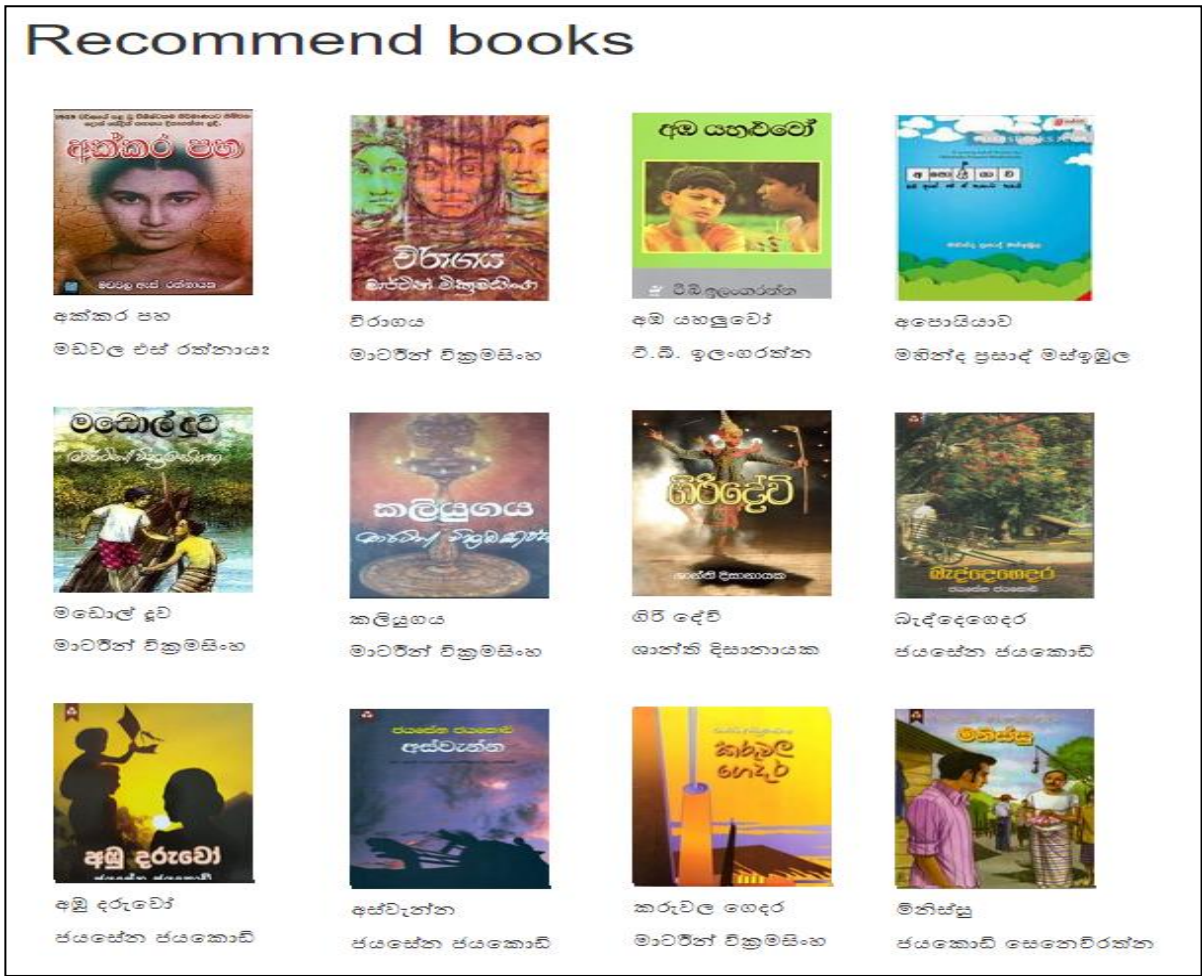


Figure 28 : Recommended Books

### 3.11.2Authenticate

The authentication also integrated with the application so that none of a user can view the data without login to the system and it was done to improve the security of the application. Admin user will be added as the main user who can view all the top rated, most popular and recommended books.

### 3.12Database

In order to store persistence data like book details, user login details my sql database is used. MySql workbench is used to manage data in mysql database. Even there are multiple database like oracle, postgres available and can be used for the same purpose, Mysql was used as it is open source and can easily be managed. Mysql connector in python was used to connect the application with the database.

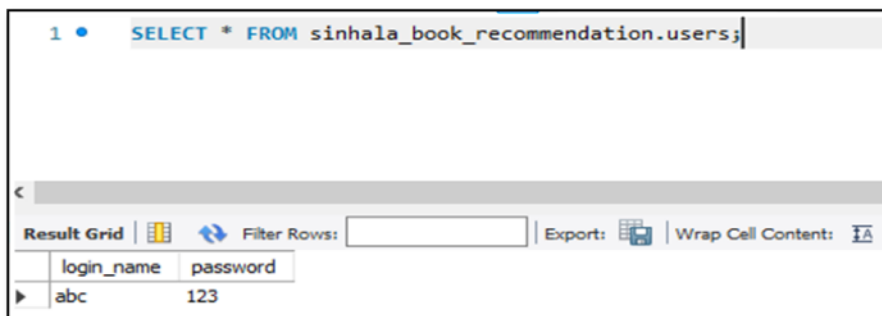


Figure 30 : User stored data

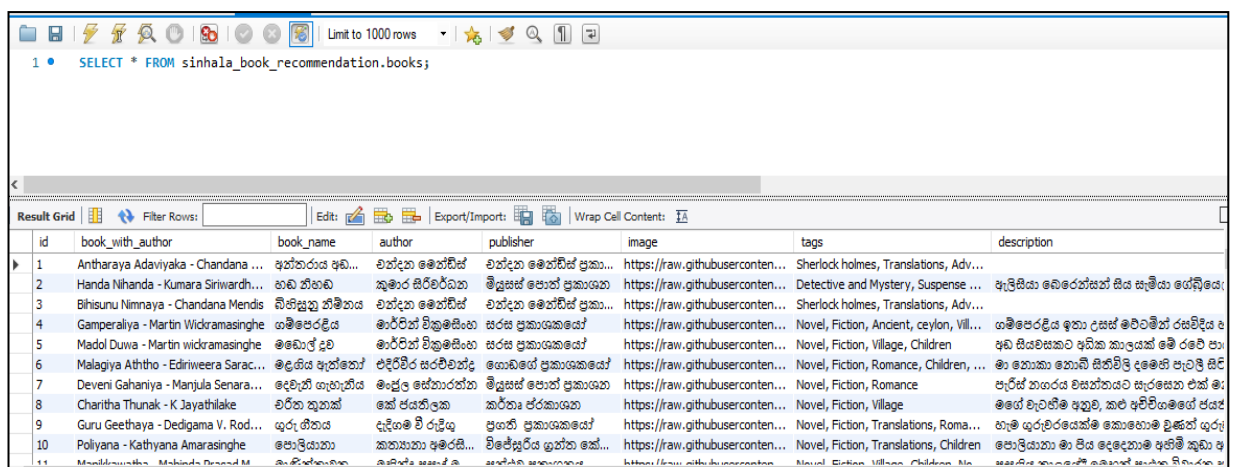


Figure 31 : Saved book details

### **3.13 Chapter Summary**

The chapter explains the main architecture of the system along with the technology which will be used to implement the system. Furthermore, technology stack and the how the evaluation will be conducted explained in the chapter in details. The clarification between the methodologies and why the hybrid method is used is also described in this chapter. Further it explained how the implementation was done mainly using python programming language. Initially the data needed to be preprocessed to remove unwanted data. Then it describes what the methods available for sentiment analysis and how the own model was build and trained. The implementation of UI and authentication was described with screen shots. Finally, how the database is connected to the application was explained.

# CHAPTER 4

## EVALUATION AND RESULTS

### 4.1 Chapter Overview

Implementing any application without proper testing or evaluation is considered as incomplete system. In industry also when we implement the application, after the QA test we hand over Client for User Acceptance Test (UAT) and get the feedback of the client. The feedback is kind of evaluation of what we have implemented.

### 4.2 Evaluation Metrics

Evaluating a book recommendation system is essential to ensure that it provides meaningful and useful suggestions to users. There are several key metrics and methods available to evaluate the accuracy and the performance of a book recommendation system.

#### 4.2.1 Accuracy Metrics

In terms of accuracy, we can use metrics such as precision, recall, and F1 score.

##### 4.2.1.1 Precision

It calculates the proportion of recommended books which are actually relevant to the user's preference. High precision means that the system provides relevant and accurate recommendations. It can be calculated by formula bellow.

$$\text{Precision} = (\text{No of relevant recommendation} / \text{Total no of})$$

##### 4.2.1.2 Recall

It calculates the proportion of a user's preferred books which were correctly recommended by the system. A high recall means that the system captures a significant portion of the user's preferences. It can be calculated by following formula.

$$\text{Recall} = (\text{No of relevant recommendations} / \text{Total no of user's preferred})$$

### 4.2.1.3 F1 Score

The F1 score is the harmonic mean of precision and recall. (“How do you calculate the F1 score in machine learning evaluation metrics?,” n.d.) It provides a balance between these two metrics, considering both false positives and false negatives. A higher F1 score indicates a well-balanced system that both accurately recommends relevant books and captures a significant portion of the user's preferences. It can be calculated by following formula.

$$\text{F1 Score} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

- If the precision is high but the recall is low, it means the system provides very accurate recommendations but may miss many relevant books. This will be suitable for users who prioritize quality over quantity.
- If the recall is high but the precision is low, it means the system captures many relevant books but also recommends a lot of irrelevant books. This will be suitable for users who want a broad range of recommendations.
- A high F1 score indicates a well-rounded system that balances precision and recall, offering both accuracy and coverage.

It's important to set an appropriate threshold for relevance when calculating these metrics. What is considered as relevant may differ from one recommendation system to another and depend on user preferences.

The accuracy metrics should be considered along with other evaluation metrics, such as user engagement, diversity, user feedback, and online or offline evaluation, to provide a more reasonable assessment of the recommendation system's performance.

## 4.2.2 User Engagement Metrics

It is the measurement getting by user actions like Click Through Rate (CTR), Conversion Rate and Bounce Rate.

### 4.2.2.1 Click Through Rate (CTR)

It calculates the percentage of users who clicked on a recommended book.

#### **4.2.2.2 Conversion Rate**

It calculates the percentage of users who buy and read the recommended book

#### **4.2.2.3 Bounce Rate**

It calculates how many users ignore or not accept the recommended books without interacting the any results.

#### **4.2.3 Diversity Metrics**

The main categories are novelty and serendipity. Novelty measures how many diverse and non-redundant book were recommended. It ensures that the system should not give same type of books repeatedly. On the other hand, serendipity measures how often the system recommends books that are not expected but appreciated by users.

#### **4.2.4 User Feedback**

In order to collect feedback from users, surveys, reviews or direct interaction can be used to understand their preferences or how satisfy they are about the system. This metrics will be used for the implemented solution to calculate the accuracy and the performance.

#### **4.2.5 Online Evaluation**

When the application is deployed to the server and getting the feedback from users is considered online evaluation. Deploying the application in Heroku kind of cloud service requires to be registered and the service is not free, the online evaluation will not be performed.

#### **4.2.6 Benchmarking**

It is the comparison the application with other popular recommend system and verify how the application fits to the industry standards.

#### **4.2.7 Mean Absolute Error (MAE)**

Even though there are many methods available to evaluate the system, the paper (Raval and Khedkar, 2019) used item based collaboration filtering recommendation system and

the Root Mean Square Error (RMSE) and Mean Absolute Error (MSE) methods were used for evaluation respectively. In the final results, their proposed method outperforms all the state-of-art methods. To align with the above research, the system implemented by (Shah, 2019) also used Mean Absolute Error (MSE) for the evaluation.

But the research (Kurmashov et al., 2015) implemented a book recommendation system which gives fast result based on collaborative filtering and use online survey for the evaluation because they realized that there is no database suitable for their task to evaluate the results. Therefore, they have selected independent readers and ask to provide a score from 1 to 10 based on the parameters like quality, convince and ease of use of the recommending system implemented. The higher score indicates the relevance of the recommendation.

To validate the system, two options are available as offline validation and online validation. For offline validation, I will be having user data and performing a standard machine learning training-test split in order to learn and train the model for the evaluation. Mean Absolute Error (MAE) or Root Mean Square Error (RMSE) or any other evaluation function could be used.

For online validation, a recommender model will be created based on information taken from other domains which is also called as cross-domain recommender system and test the system with live data. Since the implementing system do not have access to some live system, I will be focusing on finding a data set that is more than enough for the offline validation. In order to perform offline validation for the application, we can make use of the concept of precision-recall. Recall describes, what ratio of items that a user like will be actually recommended. And the precision describes out of all recommended items, how many items user actually will like. The main idea of any recommending system is recommending only items user likes. This is the optimal recommender and My target is to get as close as possible.

In order to validate the model further, expert authors will be contacted. He would check the recommended book list is matches with searched book or the list contain books which is preferred by the user.

According to the proposed solution, expecting accuracy level would be more than 80%. We can increase the accuracy by collecting and allocating more data for training. At the end of the project a user can find a best recommend books according to his preference



and the rates given by other users. Once the recommended book is read the user may realize the accuracy of the application and no need to waste time on finding the books in everywhere. Once the model is developed, we can use it to make recommendation for that we need to save the desired model and restore it when we need to do recommendation through it.

## **4.3 Test Results**

To enhance the accuracy of the recommender system both system and user feedback results will be calculated and analyzed.

### **4.3.1 Self-Evaluation**

When the researcher going through the process of implementing final year application, following categories were tested and evaluated in order to verify the implemented system is useful and most users make use of it.

#### **1) Main goal of Research Topic**

The main goal of the application is to help Sinhala book readers by recommending Selected Sinhala books based on user preference.

#### **2) Scope**

Discussed whether the scope sufficient for MCS project and is achievable during the time period.

#### **3) Design**

Designed and reviewed the architectural diagram before start the implementation of the application.

#### **4) Implementation**

A web-based application was implemented along with latest technologies and libraries. Features like top rated books, most popular books, selected book details along with user reviews and recommended books were provided.

#### **5) Testing**

How the system is tested and evaluate is discussed in details. Basic functionalities were verified and checked recommended books are related to selected book.

#### 6) Limitations and future enhancement

While testing and evaluating, some limitations were identified such as new books are not available, age range were not considering, the application is not deployed. These limitations can be addressed and enhance the application further for Sinhala book readers to make use of the application and get suggested books next to be read.

#### 4.3.1.1 Verification of Functional Requirements

Following functional requirements has been evaluated

<b>Id</b>	<b>Requirement</b>	<b>Priority</b>	<b>Status</b>
FR - 01	As a user I should be able to login to the system	High	Completed
FR - 02	As a user I should be able to see the most rated book list	High	Completed
FR - 03	As a user I should be able to see the most popular book list	High	Completed
FR - 04	As a user, I should be able to see the book details when I select a book from drop down	High	Completed
FR - 05	As a user, I should be able to see the review and the rate given by other users for a selected book	High	Completed
FR - 06	As a user, I should be able to see the book list recommended by the system for a selected book	High	Completed

*Table 3 : Functional Requirements*

#### 4.3.2 Qualitative Evaluation

In order to evaluate the quality of the application, the system will be shown to the domain experts in this case book authors and technical and industry experts. Showing the system and obtaining their feedback will be important not only to evaluate the system but also to identify the limitations and improve the system further as they have the expert knowledge in the fields. By conducting the qualitative evaluation, following criteria are captured and analyzed.

- ❖ To evaluate the novelty of the proposed application concept.
- ❖ To evaluate the scope of the project.
- ❖ To evaluate whether the system provides a solution to the existing problem.

- ❖ To identify the limitations of the project
- ❖ To evaluate how authors are benefited from the project.
- ❖ To identify the look and feel of the application in a UI/UX point of view.

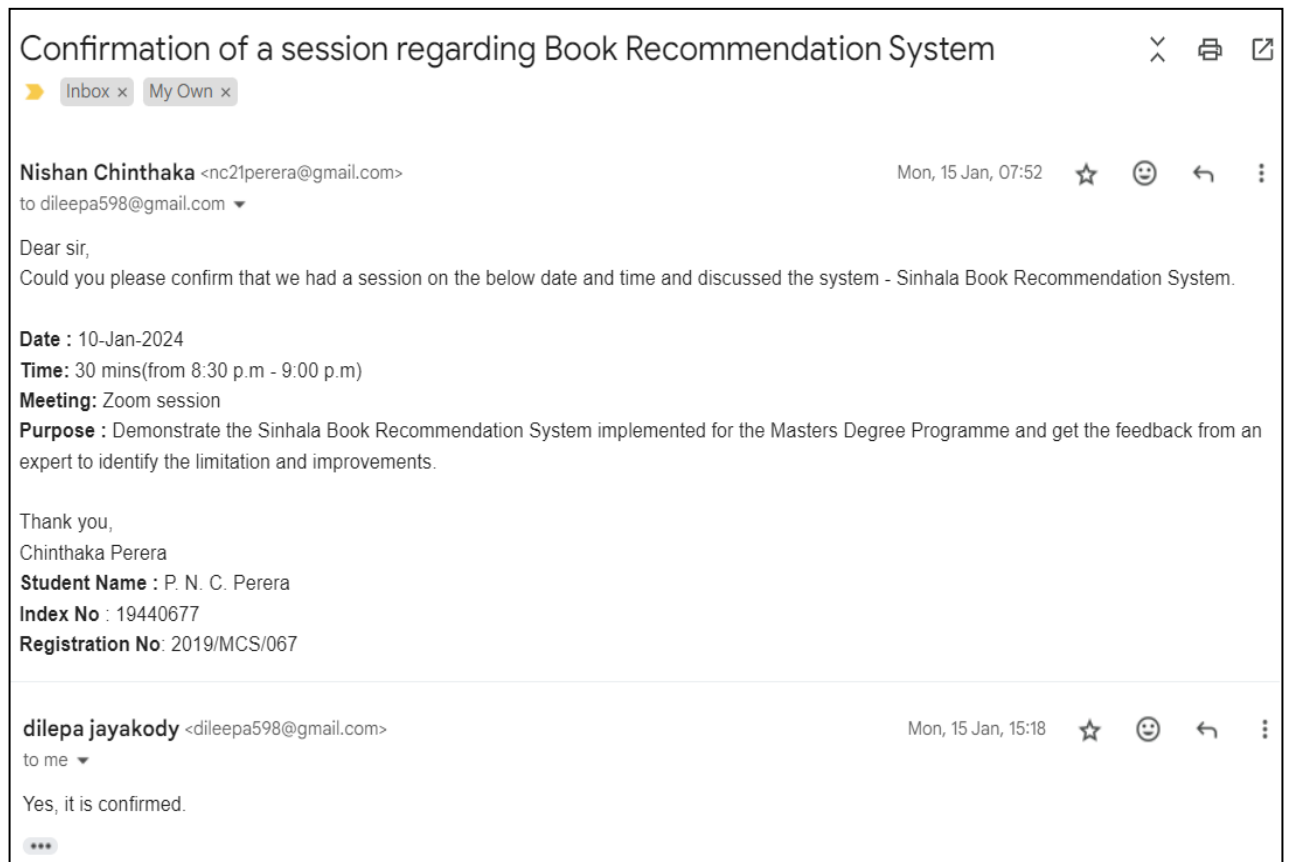
#### 4.3.2.1 Feedback from Domain experts - Famous authors

The researcher has been trying to connect famous authors in order to show the application and get the feedback to identify the limitation and improvements. Following famous authors responded for the request made for evaluate the system.

Author	Mode of shared the implemented System	Feedback
Dileepa Jayakody	Meeting via zoom	Showed the system via zoom meeting and got the feedback with area to be improved
Shamel Jayakody	Shared a demo of the implemented system via YouTube link	Showed the system via Youtube and got the feedback with area to be improved
Nethindu Warapitiya	Shared a demo of the implemented system via YouTube link	Showed the system via Youtube and got the feedback with area to be improved
Sudath Rohan	Shared a demo of the implemented system via YouTube link	Due to a technical difficulty could not get the response
Norbert Ayagamage	Shared a demo of the implemented system via YouTube link	Due to a technical difficulty could not get the response
Lasitha Raveen Umagiligy	Shared a demo of the implemented system via YouTube link	Due to a limitation of time and writing a new book, was not able to get the response
Mahesh Prasad Masimbula	Shared a demo of the implemented system via YouTube link	Due to a limitation of time and writing a new book, was not able to get the response
Mohan Raj Madawala	Shared a demo of the implemented system via YouTube link	Due to a limitation of time and writing a new book, was not able to get the response

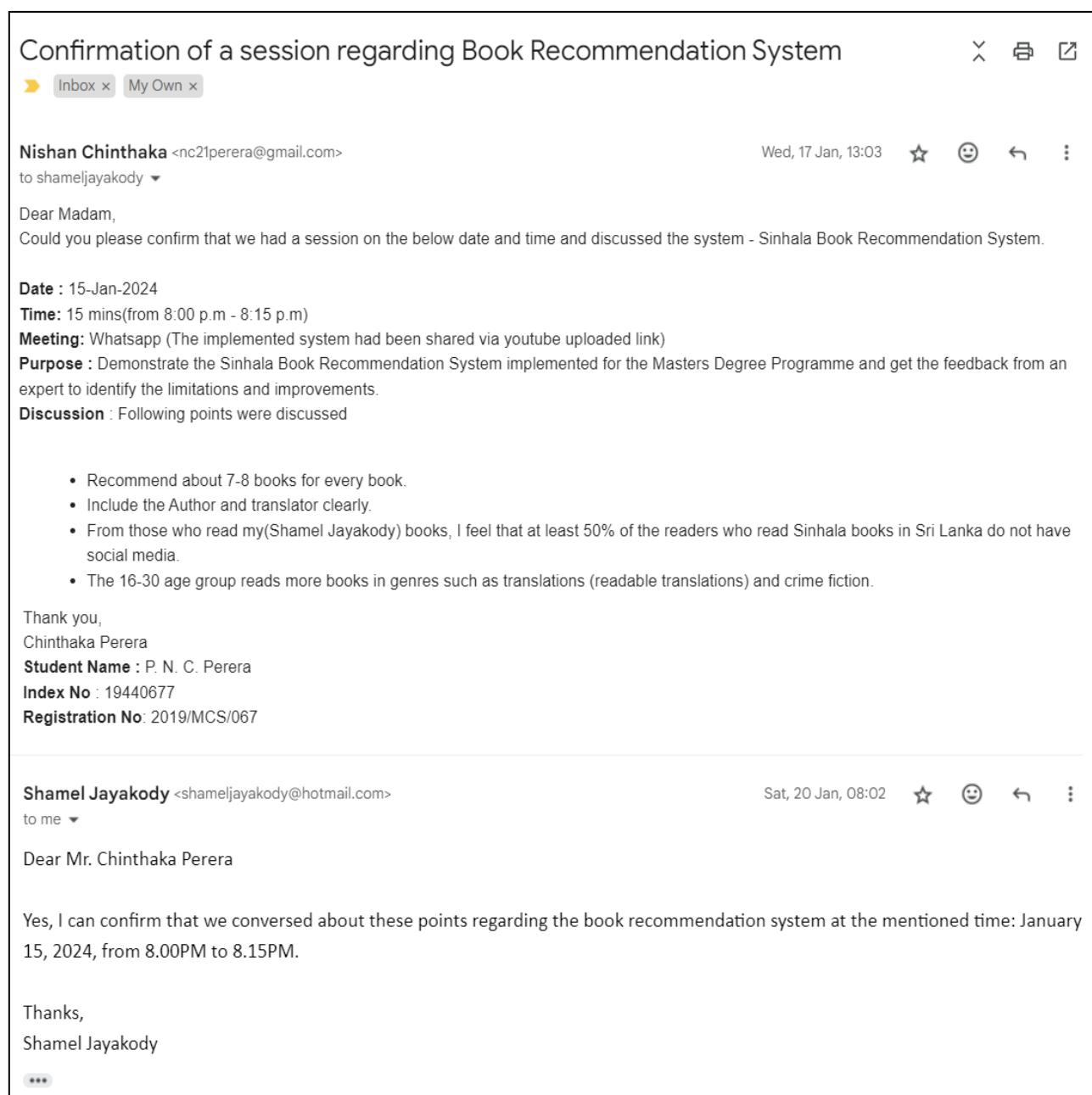
*Table 4 : Famous Authors list contacted*

## 1. The Famous Author Confirmation – Dileepa Jayakody



*Figure 32 : Email Confirmation - Dileepa Jayakody*

## 2. The Famous Author Confirmation – Shamel Jayakody

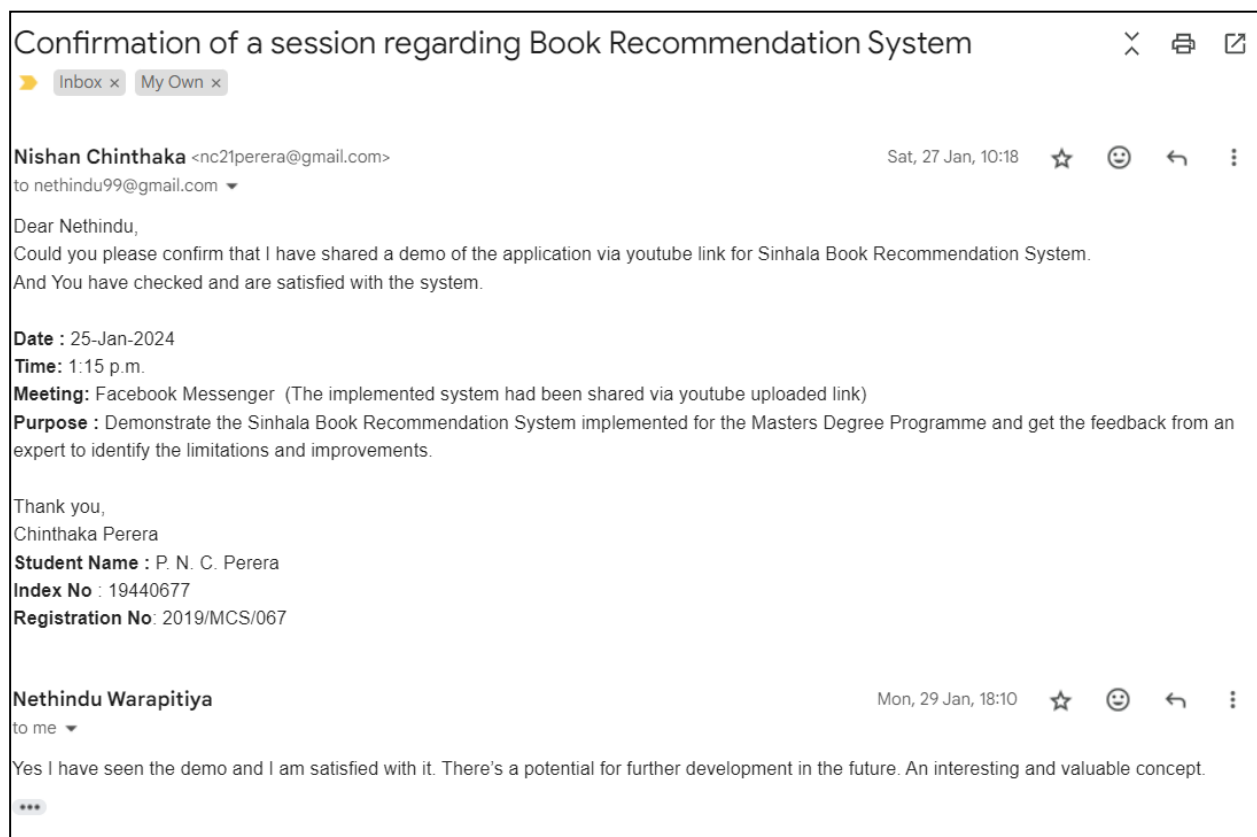


*Figure 33 : Email Confirmation - Shamel Jayakody*

### 3. The Famous Author Confirmation – Nethindu Warapitiya



*Figure 35 : Facebook respond after sharing the YouTube link of the implements system*



*Figure 34 : Email Confirmation - Nethindu Warapitiya*

### 4.3.2.2 Feedback from Technical and industry experts

Other than domain experts, system demo video has been shared among industry experts in order to get the feedback in a technical point of view as they have the experience of understand the client feedback once a feature is demonstrated to the customer. Also, they have the experience of how the user interface should be displayed to the customer to use the application ore friendly way.

Designation	Feedback
Lecturer	Exploring this domain for research is interesting due to the scarcity of existing studies in the Sinhala language. It would be valuable to incorporate review ratings or classifications on comments, facilitating users in swiftly understanding reviews according to their needs. Despite some room for improvement, it is a commendable effort, and I anticipate its availability for public use.
Senior Software Engineer	Thanks a lot for develop this system. It will help me to find new books. Great idea
Network Administrator	If you can just make it bit more UI friendly and mobile support
Fronnd end developer	Great if you can suport responsiveness, but overall it user friendly
Developer - Robotics	Rather than using a tabular format, use some symbolic system to show the ratings. E.g.stars or graph. Then, add a Link as "show more". Then you can display the first few good and bad comment seperate and your table below. I hope users will find this way much easier to sort out the book reccomandation they needed rather than analyzing data in a tabular format.
Student of Master degree programme	Your book search web system is a valuable tool for efficient book discovery. Consider enhancing it by incorporating translation features for each book, expanding its accessibility on a global scale. This would greatly benefit users worldwide, making your platform even more indispensable.
Quality Assurance Lead	Good to add book prices
Database Administrator	Regular readers looking for their own taste books always. Sometimes they r willing to read whatever they have. End of the day reading makes a full man
Architect	Location of buying options from a store or library close by. Maybe even a book borrows or swap feature. A common place for all Sinhala book readers.
Business Intelligence Lead	Use of generative AI to enhance recommendations

*Table 5 : Feedback from Technical experts*

### 4.3.2.3 Feedback from Book readers

Feedback
Great system to find new sinhala books
Personally this depends on person to person, I'm more inclined to reading English books only. I have a bit of difficulty in reading sinhala text. But it's just me. But for sinhala readers, I think this would be beneficial
Try to include latest books
This system is the best, very useful
very useful application. make it available online so every can make use of it.
even it display already read books. if you can improve to track and remove books the user already read

*Table 6 : Feedback from Normal Book Readers*

## 4.3.3 Quantitative Evaluation

### 4.3.3.1 System Calculation

#### 4.3.3.1.1 Collaborative Filter Evaluation

Mean Absolute Error (MAE) is a type of statistical accuracy metrics that is widely used to determine the quality of the recommender system specially when use collaborative filtering. The statistical based approach calculates a numerical score which is then compared with actual rating given by users. The MAE can be easily calculated by using the `mean_absolute_error()` function from Scikit-learn library. Following displays the formula for MAE.

$$\frac{1}{n} \sum_{i=1}^n \text{abs}(y_i - x_i)$$

*Figure 36 : MAE Formula*

As per the formula, it calculates the absolute different for each pair and then finally get the mean value as the result. The lower value means a better accurate results while high value means the different of predicted and actual is high.

Following method calculate the MAE and return the value.



```

pt = final_ratings.pivot_table(index='Username', columns='Book', values='Book Rate')
pt.fillna(0, inplace=True)
df_ratings = pt.copy()
similarity_matrix = cosine_similarity(pt, pt)
similarity_matrix_df = pd.DataFrame(similarity_matrix, index=pt.index, columns=pt.index)

def calculate_ratings(movie, user):
    if movie in df_ratings:
        cosine_scores = similarity_matrix_df[user] #similarity of id_user with every other user
        ratings_scores = df_ratings[movie] #ratings of every other user for the movie id_movie
        index_notRated = ratings_scores[ratings_scores.isnull()].index
        ratings_scores = ratings_scores.dropna()
        cosine_scores = cosine_scores.drop(index_notRated)
        ratings_movie = np.dot(ratings_scores, cosine_scores)/cosine_scores.sum()
    else:
        return 2.5
    return ratings_movie

calculate_ratings('12.12.12 - Manjula Senarathna','91iahadarshi@gmail.com')

```

```

def score_on_test_set():
    npArr=[]
    new_df = final_ratings[['Book', 'Username']]
    for index, row in new_df.iterrows():
        npArr.append(calculate_ratings(row['Book'], row['Username']))
    predicted_ratings = np.array(npArr);
    true_ratings = np.array(final_ratings['Book Rate'])
    score = np.sqrt(mean_absolute_error(true_ratings, predicted_ratings))
    return score.round(5)
test_set_score = score_on_test_set()
print('***** >', test_set_score)

```

```
***** > 2.52076
```

*Figure 37 : Invoke the MAE method for all data*

#### 4.3.3.1.2 Content based Filter Evaluation

The above MAE was used to calculate the accuracy of the approach which has a numeric field in our case 'Book Rate'. The data set used for collaborative filtering have the rate field. But as per the data set of books having tags does not have any numeric field and therefore MAE cannot be applied for Content based filtering. But since the accuracy gives the correctness of the implemented application, a different approach which only works with text should be used. We integrate Artificial Neural Network (ANN) for the application and predict the recommended book list. The results can be compared and accuracy can be calculated with implemented Content based model.

#### 4.3.3.1.3 Artificial Neural Network (ANN)

The Artificial Neural Network is a connected network which takes an input value and computes the desired output. The book with tags dataset can be considered as input data and recommended book list is the output. The reason behind selecting the ANN is it's not just giving the recommended books but also compare the results with accuracy percentage.

Following table shows the accuracy for the selected book

User Selected Book	Number of books recommended by Content based and are listed in the list recommended by ANN out of 20 Books	Accuracy
Oliver Twist - M. M. Piyawardana	17	85%
Hari Puduma Iskole - Leelananda Gamachchi	19	95%
Bhayanaka Miniha - Chandana Mendis	16	80%
Rathu Rosa - Kumara Karunarathna	14	70%
Sanda Wiyaruwa - Bhadraraj Mahinda Jayathilaka	13	65%
Iti Pahan - Sumithra Rahubadda	13	65%
Gahanu Lamayi - Karunasena Jayalath	20	100%
105 - Dileepa Jayakody	16	80%
Apuru Iskole Apuru Dawas - Sudath Rohan	16	80%
Bindunu Bilinda - Dileepa Jayakody	13	65%
Emily 01 - Manel Jayanthi Gunasekara	18	90%
Anne 01 (Arabe Gedara Anne) - Premasiri Mahingoda	17	85%
<b>Total Average</b>		<b>80%</b>

*Table 7 : Accuracy calculated for selected book*

#### 4.3.4 Online Survey

This study will use 32 test subjects. A questionnaire was prepared to determine user satisfaction and the quality of the suggested book list for the users. Five-point Likert-scale survey questions were asked. Likert scales have become an essential survey tool to get feedback on a person's opinion or attitude regarding an item. It ranges from polar opposites to complete satisfaction to complete dissatisfaction. Questions were structured to be asked under the categories of accuracy, familiarity, novelty of the book recommendations, and interactivity of the system. An optional question was asked if the user wanted to give any suggestions or feedback for further improving the system. This questionnaire determines whether the implemented system has met the objectives and met the user's requirements and needs.

The Questionnaire for the evaluation is listed in Appendix B.

The summary of the results is as bellow. More details for the results for the online evaluation is attached in Appendix C.

	<b>Question</b>	<b>Positive Count</b>	<b>Total</b>	<b>Percentage</b>
1	I think a system is required to find similar books based on users' preference or content as I am struggled finding new books similar to books I have read and interested.	29	32	90,63
2	The implemented system accurately recommends books	31	32	96,88
3	The system helped me to find new books	31	32	96,88
4	It is easy to navigate and use the system	29	32	90,63
5	Overall, I'm satisfied with the recommender system	31	32	96,88
6	I would recommend this Sinhala Book Recommendation system to others.	31	32	96,88

*Table 8 : Feedback of online survey*

## 4.4 Chapter Summary

Evaluate the implemented system was discussed in this chapter with test results. In order to evaluate the system, some experts will be contacted and get their feedback. The accuracy of the evaluation will also be discussed in the phrase.

## **CHAPTER 5**

### **CONCLUSION AND FUTURE WORK**

#### **5.1 Chapter Overview**

The conclusion chapter presents the final part of the application and discuss whether the aim and the objectives have been achieved successfully. Furthermore, the limitations and future work will be discussed in order for someone to add the missing feature and enhance the application.

#### **5.2 Conclusion**

The idea to implement an application for Sinhala Book recommendation came to the mind by surfing Facebook book related groups. Many people ask so many questions like is this a good book to read, I like books related to history and romance and please suggest me some books. The process of implementing such system began by finding any data set. Kaggle and many dataset providers have not provided any data set related to Sinhala books, thus a google form was created and shared among Facebook groups to collect the data which was a challenge. Even the form was share among groups, not able to collect data set as expected. Therefore, each member of groups was contacted and shared the form to be filled.

In parallel of collecting data some research papers were read and understand what are the system available and limitation of them. In literature review chapter all the details were discussed. Then the implementation started by learning Python programming language. In order to apply the hybrid model, First Collaboration filter was applied and get recommended book list and the Content based filtering was applied and get the suggested book list, finally both lists were combined and display the result to the users. A User interface was implemented to display book details, reviews and recommended book list in an attractive way. Mean Absolute Error was used to evaluate the collaborative filter and same application was implemented in Artificial Neural Network and compare the result to evaluate the Content based filtering. Based on the feedback provide by the authors and technical experts the final result of the system was evaluated.

### 5.3 Challenges and Solutions

Some of the challenges while implementing a recommendation for Sinhala Books are listed down as bellow.

1. There was no any dataset

To implement any machine learning application, dataset plays a significant role. The researcher was finding a dataset from Kaggle, and many dataset providers. Even they have so many datasets for English books, none of them have a dataset related to Sinhala books. Since the interest of implementing the system, a google form was created and shared among groups as a first step to collect the dataset. Even the form was shared in many groups, sufficient data set was not able to be collected. Therefore, each individual reader was contacted and shared the form to fill. And the researcher visited international book fair and physically collected data by considering the value readers will be getting after implementing the system.

2. Finding a technology and implementing the system

Finding a technology to implement the system was not an easy as the researcher mainly working with Java technology in the industry. Learning Python and UI related technology like css, bootstrap was a challenge and watching YouTube until implementing the system was an achievement.

3. Evaluation

Even there is a method to evaluate the collaboration filter, no method found to evaluate content-based filter which no rate attached to it. If there was a rate in Content based, same method could be implemented. Therefore, the same implementation was done using Artificial Neural Network and compared the result to evaluate the content-based filter.

### 5.4 Limitation

Even most of the features were able to completed as per the proposal, some limitations were found to be implemented as an enhancement.

1. The age and gender fields were not considered for recommendation.
2. The latest books have not been considered as the book list was created in 2022.

3. If user have not read any books, he is not able to use the system without select at least a book from the list.
4. Since there is a drop down to select books, most books were selected from the top of the list.

## **6.1 Future Enhancements**

The application was implemented as per the proposed system and there are some features that could be added as enhancements for the features of the system.

1. The drop-down book list was created in English. Most people entered data, suggested to display them in Sinhala as the project is related to Sinhala books.
2. The user should be able to select any books based on some categories like author and genre like history, romance, detective.
3. The dataset was limited like only around 4500 were able to be collected.
4. Reviews were displays as the way the users entered in the google form. But since the application is implemented for Sinhala book readers, could use a google translator and display all the reviews in Sinhala.

## **6.2 Chapter Summary**

The main goals and objectives of the application were defined at the introduction chapter and the conclusion chapter discussed whether all of them have been successfully achieved. The limitation and future enhancements were discussed.

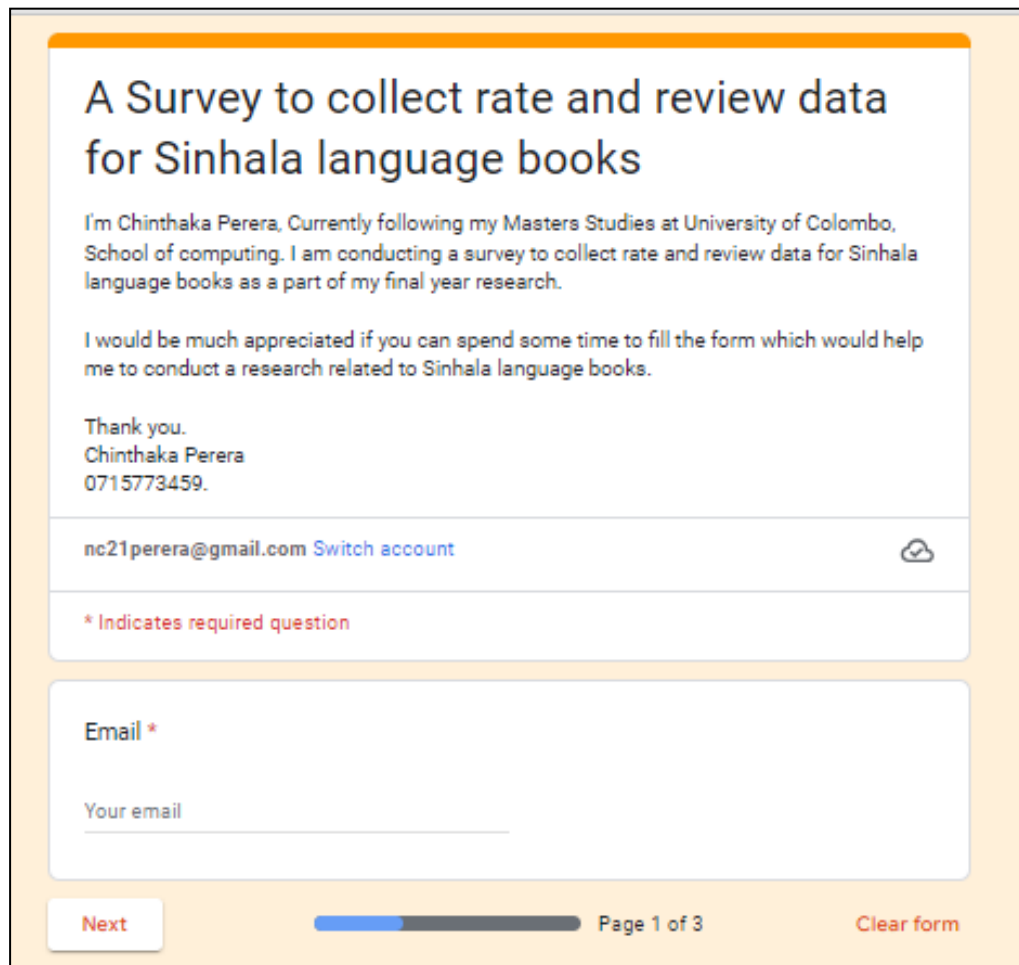
## APPENDICES

### Appendix – A

1. The url for the data collection form

[https://docs.google.com/forms/d/e/1FAIpQLSd1UaBYtuAcuYqOIyeSOttzw2N-  
iu\\_HbvGSAzGOSp-XrxLFOQ/viewform](https://docs.google.com/forms/d/e/1FAIpQLSd1UaBYtuAcuYqOIyeSOttzw2N-<br/>iu_HbvGSAzGOSp-XrxLFOQ/viewform)

2. The Questionnaire for data Collection



The screenshot shows a Google Form titled "A Survey to collect rate and review data for Sinhala language books". The form is set against a light orange background. The title is in a large, bold, dark blue font. Below the title, there is a paragraph of text in a smaller, dark blue font: "I'm Chinthaka Perera, Currently following my Masters Studies at University of Colombo, School of computing. I am conducting a survey to collect rate and review data for Sinhala language books as a part of my final year research." This is followed by another paragraph: "I would be much appreciated if you can spend some time to fill the form which would help me to conduct a research related to Sinhala language books." Below this, there is a "Thank you." message and the name "Chinthaka Perera" with a phone number "0715773459." At the bottom of this section, there is a line with the email "nc21perera@gmail.com" and a "Switch account" link. Below this is a red asterisk followed by the text "\* Indicates required question". The main content area of the form is a white box with a light orange border. Inside this box, the word "Email" is followed by an asterisk. Below this, there is a text input field with the placeholder text "Your email". At the bottom of the form, there is a "Next" button, a progress bar showing the current page as "Page 1 of 3", and a "Clear form" link.

**A Survey to collect rate and review data for Sinhala language books**

I'm Chinthaka Perera, Currently following my Masters Studies at University of Colombo, School of computing. I am conducting a survey to collect rate and review data for Sinhala language books as a part of my final year research.

I would be much appreciated if you can spend some time to fill the form which would help me to conduct a research related to Sinhala language books.

Thank you.  
Chinthaka Perera  
0715773459.

nc21perera@gmail.com [Switch account](#)

\* Indicates required question

**Email \***

Your email

[Next](#) [Clear form](#) Page 1 of 3

## A Survey to collect rate and review data for Sinhala language books

nc21perera@gmail.com [Switch account](#)



\* Indicates required question

Gender \*

- ☐ Male  
☐ Female

Age Range \*

Choose ▼

1.1) Book \*

Choose ▼

1.2) Book - Rate (?/10) \*

Worst 1 2 3 4 5 6 7 8 9 10 Best  
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

1.3) Why do you select above rate? \*

Your answer

2.1) Book \*

Choose ▼

2.2) Book - Rate (?/10) \*

Worst 1 2 3 4 5 6 7 8 9 10 Best  
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

2.3) Why do you select above rate? \*

Your answer

[Back](#)

[Next](#)

Page 2 of 3

[Clear form](#)



#### Untitled Section

3.1) Book \*

Choose

3.2) Book - Rate (?/10) \*

1 2 3 4 5 6 7 8 9 10  
Worst ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ Best

3.3) Why do you select above rate? \*

Your answer

4.1) Book \*

Choose

4.2) Book - Rate (?/10) \*

1 2 3 4 5 6 7 8 9 10  
Worst ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ Best

4.3) Why do you select above rate?\* \*

Your answer

5.1) Book \*

Choose

5.2) Book - Rate (?/10) \*

1 2 3 4 5 6 7 8 9 10  
Worst ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ Best

5.3) Why do you select above rate? \*

Your answer

6) What are books not listed in the list and you think to be included

Your answer

☐ Send me a copy of my responses.

[Back](#)

[Submit](#)

Page 3 of 3

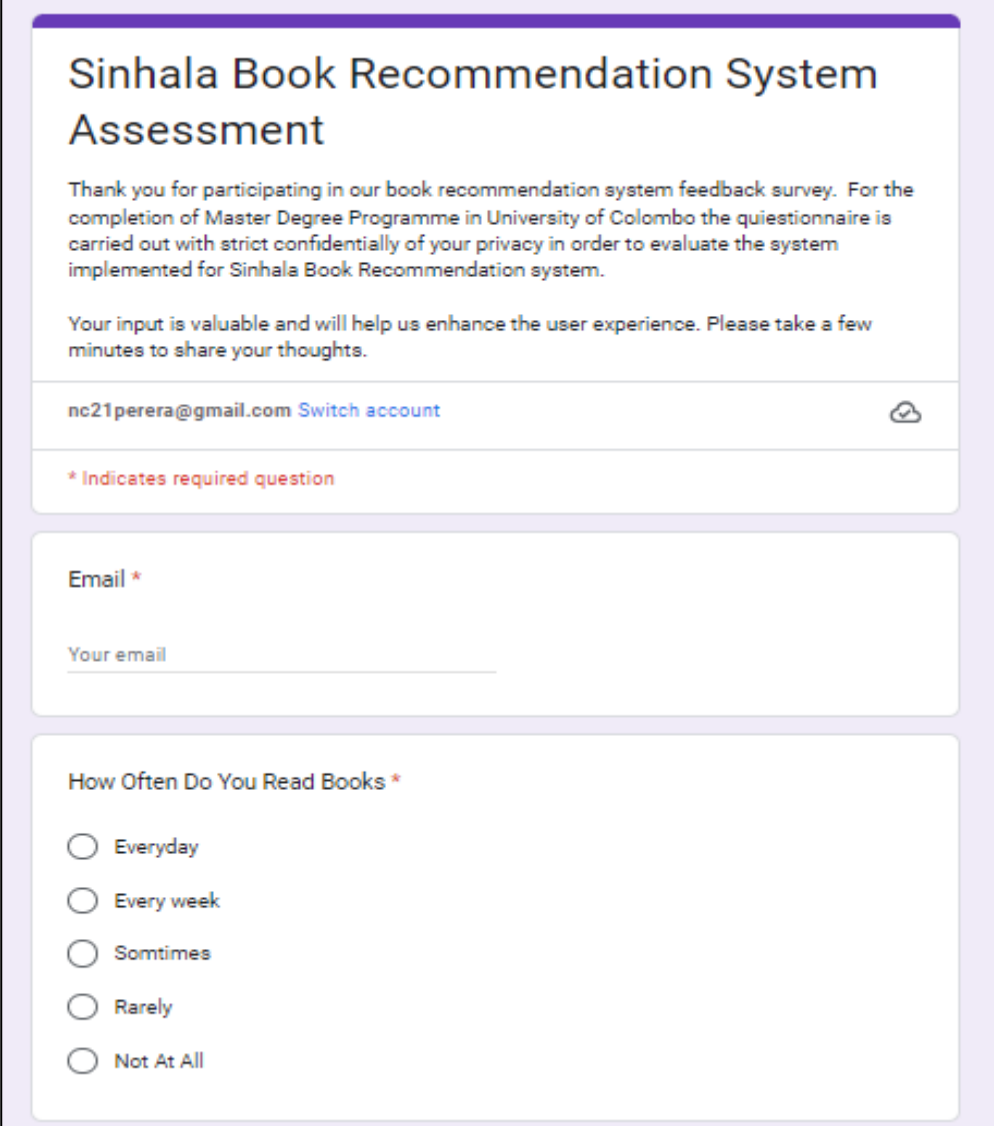
[Clear form](#)

## Appendix - B

1. The url for the data evaluation form

<https://docs.google.com/forms/d/e/1FAIpQLSdmIYoCEFIGNpNEUIwq3sM-etMaiAOgsMyVNe2A8wQdoOL6JQ/viewform>


2. The Questionnaire for data Evaluation



**Sinhala Book Recommendation System Assessment**

Thank you for participating in our book recommendation system feedback survey. For the completion of Master Degree Programme in University of Colombo the questionnaire is carried out with strict confidentiality of your privacy in order to evaluate the system implemented for Sinhala Book Recommendation system.

Your input is valuable and will help us enhance the user experience. Please take a few minutes to share your thoughts.

nc21perera@gmail.com [Switch account](#) 

\* Indicates required question

**Email \***

Your email

**How Often Do You Read Books \***

☐ Everyday

☐ Every week

☐ Sometimes

☐ Rarely

☐ Not At All

I think a system is required to find similar books based on users preference or content as I am struggled finding new books similar to books I have read and interested. \*

- ☐ Strongly Agree
- ☐ Agree
- ☐ Neutral
- ☐ Disagree
- ☐ Strongly Disagree

The implemented system accurately recommend books \*

- ☐ Strongly Agree
- ☐ Agree
- ☐ Neutral
- ☐ Disagree
- ☐ Strongly Disagree

The system helped me to find new books \*

- ☐ Strongly Agree
- ☐ Agree
- ☐ Neutral
- ☐ Disagree
- ☐ Strongly Disagree

It is easy to navigate and use the system \*

- ☐ Strongly Agree
- ☐ Agree
- ☐ Neutral
- ☐ Disagree
- ☐ Stronly Disagree

What is the most attractive feature of the application \*

- ☐ Login and Authenticate
- ☐ Top Rated Book List
- ☐ Most Popular Book List
- ☐ Book Details
- ☐ Reviews from others
- ☐ Recommendation Book List

Overall, Im satisfied with the recommender system \*

- ☐ Strongly Agree
- ☐ Agree
- ☐ Average
- ☐ Disagree
- ☐ Stronly Disagree

I would recommend this Sinhala Book Recommendation system to others. \*

- ☐ Strongly Agree
- ☐ Agree
- ☐ Average
- ☐ Disagree
- ☐ Strongly Disagree

Please mention any suggestions or feedback if you have

Your answer

#### Thank You

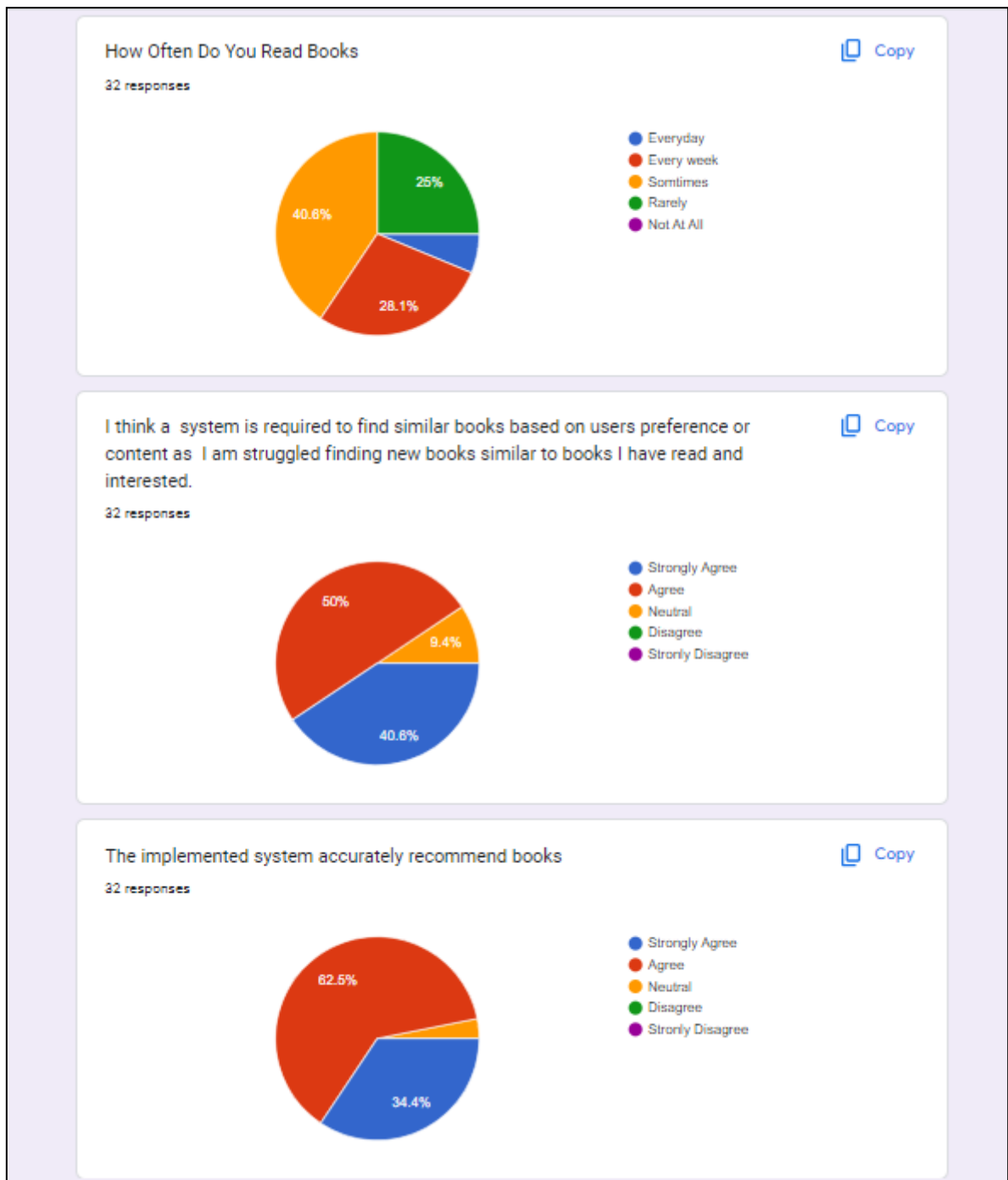
Thank you for completing our feedback questionnaire. Your input is valuable in helping us enhance the book recommendation system. If you have any additional comments or suggestions, please feel free to share them. Your feedback is greatly appreciated!

Submit

Clear form

Never submit passwords through Google Forms.

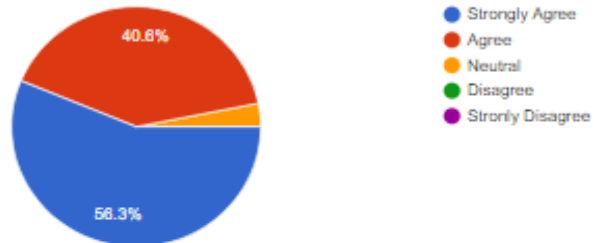
## Appendix - C



### The system helped me to find new books

 Copy

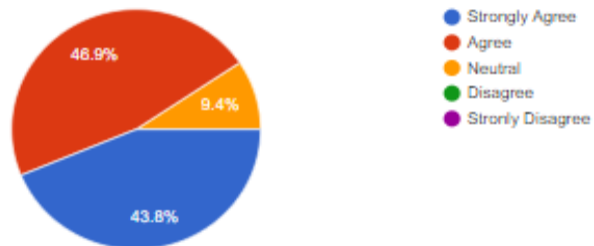
32 responses




### It is easy to navigate and use the system

 Copy

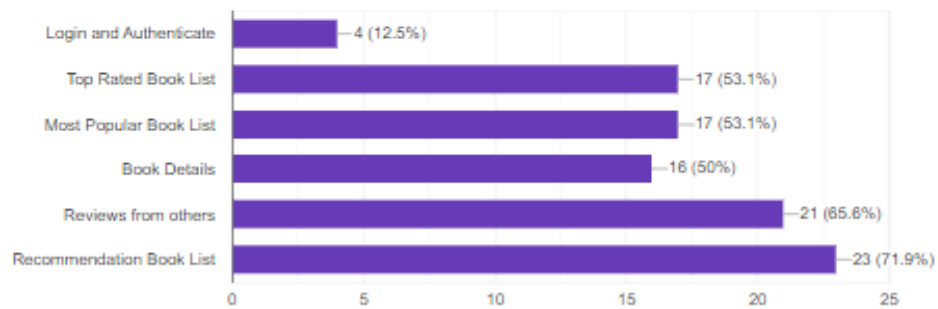
32 responses



### What is the most attractive feature of the application

 Copy

32 responses

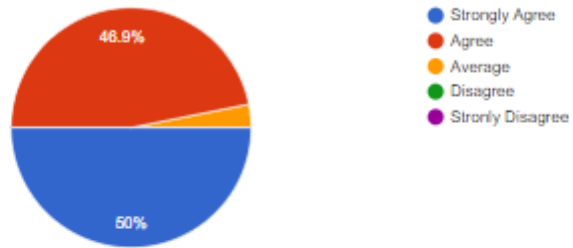





Overall, I'm satisfied with the recommender system

 Copy

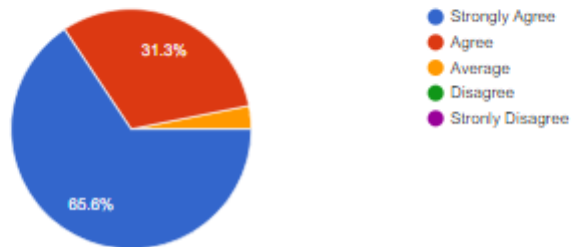
32 responses



I would recommend this Sinhala Book Recommendation system to others.

 Copy

32 responses



## REFERENCES

1. Barai, M.K., 2021. Sentiment Analysis with TextBlob and Vader. Analytics Vidhya. URL <https://www.analyticsvidhya.com/blog/2021/10/sentiment-analysis-with-textblob-and-vader/> (accessed 10.30.23).
2. Caren, N., 2019. Word Lists and Sentiment Analysis [WWW Document]. Neal Caren. URL <https://nealcaren.org/lessons/wordlists/> (accessed 2.13.24).
3. Chandrasekaran, G., Antoanela, N., Andrei, G., Monica, C., Hemanth, J., 2022. Visual Sentiment Analysis Using Deep Learning Models with Social Media Data. Applied Sciences 12, 1030. <https://doi.org/10.3390/app12031030>
4. Dhanda, M., Verma, V., 2016. Recommender System for Academic Literature with Incremental Dataset. Procedia Computer Science 89, 483–491. <https://doi.org/10.1016/j.procs.2016.06.109>
5. How do you calculate the F1 score in machine learning evaluation metrics? [WWW Document], n.d. URL <https://www.linkedin.com/advice/3/how-do-you-calculate-f1-score-machine-learning-6ngoe> (accessed 2.13.24).
6. Ijaz, F., n.d. Book Recommendation System using Machine learning.
7. Kurmashov, N., Latuta, K., Nussipbekov, A., 2015. Online book recommendation system, in: 2015 Twelve International Conference on Electronics Computer and Computation (ICECCO). Presented at the 2015 Twelve International Conference on Electronics Computer and Computation (ICECCO), IEEE, Almaty, Kazakhstan, pp. 1–4. <https://doi.org/10.1109/ICECCO.2015.7416895>
8. Machine Learning Project | Classification | Sentiment Analysis | Sinhala - YouTube [WWW Document], n.d. URL <https://www.youtube.com/playlist?list=PL495mke12zYDPRGhXd6JGY5EUoksIVwYU> (accessed 10.29.23).
9. Marappan, R., 2022. Create a Book Recommendation System using Collaborative Filtering. IJMEBAC 1, 44–46. <https://doi.org/10.31586/ijmebac.2022.341>
10. Mercy Milcah Y, Moorthi K, Jansons Institute of Technology, 2020. AI based Book Recommender System with Hybrid Approach. IJERT V9, IJERTV9IS020416. <https://doi.org/10.17577/IJERTV9IS020416>
11. Murali, M.V., Vishnu, T.G., Victor, N., 2019. A Collaborative Filtering based Recommender System for Suggesting New Trends in Any Domain of Research, in: 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS). Presented at the 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), IEEE, Coimbatore, India, pp. 550–553. <https://doi.org/10.1109/ICACCS.2019.8728409>
12. Raval, N., Khedkar, V., 2019. A Review Paper On Collaborative Filtering Based Moive Recommedation System 8.
13. Roy, D., Dutta, M., 2022. A systematic review and research perspective on recommender systems. J Big Data 9, 59. <https://doi.org/10.1186/s40537-022-00592-5>

14. Sallam, R.M., Hussein, M., Mousa, H.M., 2020. An Enhanced Collaborative Filtering-based Approach for Recommender Systems. *IJCA* 176, 9–15. <https://doi.org/10.5120/ijca2020920531>
15. Sarma, D., Mittra, T., Shahadat, M., 2021. Personalized Book Recommendation System using Machine Learning Algorithm. *IJACSA* 12. <https://doi.org/10.14569/IJACSA.2021.0120126>
16. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S., n.d. 9 Collaborative Filtering Recommender Systems.
17. Shah, K., 2019. Book Recommendation System using Item based Collaborative Filtering 06.
18. Tian, Y., Zheng, B., Wang, Y., Zhang, Y., Wu, Q., 2019. College Library Personalized Recommendation System Based on Hybrid Recommendation Algorithm. *Procedia CIRP* 83, 490–494. <https://doi.org/10.1016/j.procir.2019.04.126>
19. Tripathy, A., Agrawal, A., Rath, S.K., 2015. Classification of Sentimental Reviews Using Machine Learning Techniques. *Procedia Computer Science* 57, 821–829. <https://doi.org/10.1016/j.procs.2015.07.523>
20. Wadikar, D., Kumari, N., Bhat, R., Shiroadkar, V., 2020. Book Recommendation Platform using Deep Learning 07.
21. Wang, D., Liang, Y., Xu, D., Feng, X., Guan, R., 2018. A content-based recommender system for computer science publications. *Knowledge-Based Systems* 157, 1–9. <https://doi.org/10.1016/j.knosys.2018.05.001>
22. Wassan, S., Chen, X., Shen, T., Waqar, M., Jhanjhi, N., 2021. Amazon Product Sentiment Analysis using Machine Learning Techniques.
23. What is Natural Language Processing? An Introduction to NLP [WWW Document], n.d. . Enterprise AI. URL <https://www.techtarget.com/searchenterpriseai/definition/natural-language-processing-NLP> (accessed 11.12.23).