

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

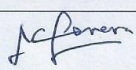
2024

Statement of Declaration

Declaration

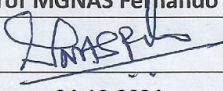
Name of the student: P. N. C. Perera
Registration number: 2019/MCS/067
Name of the Degree Programme: Master of Computer Science
Project/Thesis title: Content and Collaborate based Sinhala Book Recommendation system

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.

Signature of the Student	Date (DD/MM/YYYY)
	24-10-2024

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.

	Supervisor 1	Supervisor 2	Supervisor 3
Name	Prof MGNAS Fernando		
Signature			
Date	24-10-2024		

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

ACKNOWLEDGEMENTS

Completing the Master's degree program while managing the demands of a high-stress work environment and fulfilling responsibilities as a father was a significant challenge. It was a valuable life experience to balance these responsibilities effectively and meet all deadlines. I would like to extend my heartfelt gratitude to everyone who supported and assisted me throughout this journey, contributing to the successful completion of my degree.

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 growing availability of digital texts in Sinhala language presents an opportunity for the development of effective book recommendation systems tailored to Sinhala-speaking users. This thesis explores the implementation of hybrid technique combining collaborative filtering and collaborative based filtering to enhance the accuracy and relevance of book recommendations in Sinhala. The study begins with an extensive analysis of existing book recommendation systems, identifying their limitations and areas for improvement. Subsequently, it delves into the principles of collaborative filtering, investigating its applicability and effectiveness within the context of the literature.

A novel collaborative filtering algorithm, optimized for Sinhala text processing, is proposed and rigorously evaluated against benchmark datasets. The algorithm leverages user-item interaction data to generate book recommendations, taking into account user preferences, book attributes, and rates specific to Sinhala books. Furthermore, the thesis explores various strategies for enhancing the scalability and efficiency of the recommendation system, considering factors such as computational resources and data sparsity by integrating content-based filter.

The evaluation results demonstrate significant improvements in recommendation accuracy compared to existing approaches, affirming the efficacy of the proposed collaborative filtering and content-based algorithm. Additionally, user feedback and qualitative analysis provide insights into the usability and user satisfaction of the system.

Overall, this thesis contributes to the advancement of Sinhala book recommendation systems, offering valuable insights and practical solutions to address the unique challenges posed by the Sinhala language and its literary ecosystem.

Keywords: Book recommendation, Collaborative filtering, Content based filtering, Hybrid model, Sinhala

TABLE OF CONTENTS

ACKNOWLEDGMENT.....	iii
ABSTRACT.....	iv
TABLE OF CONTENT.....	v
LIST OF FIGURES.....	vii
LIST OF TABLES.....	viii
LIST OF ABBREVIATIONS.....	ix
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	3
1.6 Scope.....	4
1.6.1 In scope	4
1.6.2 Out scope.....	4
1.7 Chapter Summary	5
CHAPTER 2.....	6
LITERATURE REVIEW	6
2.1 Chapter Overview	6
2.2 Problem Domain	6
2.2.1 Natural Language Processing	6
2.2.2 Machine Learning Models.....	7
2.3 Literature Review.....	8
2.4 Chapter Summary	13
CHAPTER 3.....	14
METHODOLOGY	14
3.1 Chapter Overview	14
3.2 Data Set.....	14
3.2.1 Validate Data Set.....	14
3.2.2 Prepare Data Set	16
3.3 Architectural Diagram.....	17
3.4 Apply Machine Learning Models	19
3.5 Web Application	19
3.6 Technology Selection.....	20
3.7 Preprocessing.....	20
3.7.1 Convert Sinhala review to English	21
3.7.2 Sentiment Analysis.....	21
3.7.2.1 Using Libraries	22
3.7.2.2 Using Own Mechanism	24
3.7.2.3 Get Sentiment Analysis Rate	31
3.8 Collaboration based filter	31
3.9 Content based filter	35
3.10 Web Application	37
3.10.1 User Interface	37

3.10.2 Authenticate.....	43
3.11 Database.....	44
3.12 Chapter Summary	45
CHAPTER 4.....	46
EVALUATION AND RESULTS	46
4.1 Chapter Overview	46
4.2 Evaluation Metrics	46
4.2.1 Accuracy Metrics	46
4.2.2 User Engagement Metrics	47
4.2.3 User Feedback	47
4.2.4 Online Evaluation.....	48
4.3 Mean Absolute Error (MAE)	48
4.4 Test Results	49
4.4.1 Self-Evaluation.....	49
4.4.2 Verification of Functional Requirements	50
4.4.3 Qualitative Evaluation	51
4.4.3 Quantitative Evaluation.....	56
4.4.3.1 System Calculation	56
4.4.3.2 Online Survey	58
4.5 Chapter Summary	59
CHAPTER 5.....	60
CONCLUSION AND FUTURE WORK	60
5.1 Chapter Overview	60
5.2 Conclusion	60
5.3 Limitation.....	60
5.4 Future Enhancements.....	61
5.5 Chapter Summary	61
Appendix – A.....	I
Appendix - B	III
Appendix - C	IX
REFERENCES	XII

LIST OF FIGURES

Figure 1. 1 : Questions Users ask from groups.....	2
Figure 2. 1 : Machine Learning Models.....	7
Figure 2. 2 : precision of different algorithm.....	10
Figure 3. 1 : Collected data sample	16
Figure 3. 2 : Expected format to apply algorithms.....	16
Figure 3. 3 : Architecture Diagram.....	17
Figure 3. 4 : Review after converting to English.....	21
Figure 3. 5 : VADER result	23
Figure 3. 6 : TextBlob result.....	23
Figure 3. 7 : Kindle Review Data sample.....	24
Figure 3. 8 : The way how the stemming is applied	27
Figure 3. 9 : Build the vocabulary.....	27
Figure 3. 10 : Vectorization 02.....	27
Figure 3. 11 : Vectorization 01.....	27
Figure 3. 12 : Balance Dataset.....	30
Figure 3. 13 : All models related to recommend system.....	32
Figure 3. 14 : User based Collaborative filter	33
Figure 3. 15 : Item based collaborative filter	33
Figure 3. 16 : User vs Rate matrix	34
Figure 3. 17 : User vs book rate matrix	34
Figure 3. 18 : Collaborative filter result	35
Figure 3. 19 : Content based recommended book list	37
Figure 3. 20 : Login Page.....	40
Figure 3. 21 : Top Rated Books.....	40
Figure 3. 22 : Popular Book List.....	41
Figure 3. 23 : List of books user can select	42
Figure 3. 24 : Selected book details with reviews and rate	42
Figure 3. 25 : Recommended Books	43
Figure 3. 26 : User stored data	44
Figure 3. 27 : Saved book details	44
Figure 4. 1 : Email Confirmation - Dileepa Jayakody	52
Figure 4. 2 : Email Confirmation - Shamel Jayakody	53
Figure 4. 3 : Facebook respond after sharing the YouTube link of the implements system	54
Figure 4. 4 : Email Confirmation - Nethindu Warapitiya	54
Figure 4. 5 : MAE Formula	56
Figure 4. 6 : Invoke the MAE method for all data	57

LIST OF TABLES

<i>Table 3. 1 : Google form with fields details</i>	<i>15</i>
<i>Table 3. 2 : Technology Stack.....</i>	<i>20</i>
<i>Table 4. 1 : Functional Requirements</i>	<i>50</i>
<i>Table 4. 2 : Famous Authors list contacted</i>	<i>51</i>
<i>Table 4. 3 : Feedback from Technical experts.....</i>	<i>55</i>
<i>Table 4. 4 : Feedback from Normal Book Readers</i>	<i>56</i>
<i>Table 4. 5 : Accuracy calculated for selected book</i>	<i>58</i>
<i>Table 4. 6 : Feedback of online survey</i>	<i>59</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

On social media like Facebook, there are lots of groups for almost anything you can think of. As an example, if you have a special kind of car, there's likely a group for owners like you. These groups are helpful because you can learn a lot from them, and if you have any questions, you can ask the group and get answers.

For people who love reading Sinhala books, there are many groups on Facebook too. In these groups, you can share your thoughts on a book you've read, or find out about new books if you follow the groups. Many people ask for book recommendations, like books similar to ones they've enjoyed, or books about Sri Lankan history or World War. Some people even ask if a book is worth reading by posting a picture of it. After seeing these questions, it would be a good idea to have a system that shows what other users think about a particular book, how good it is, and recommend similar books. This could motivate us to start a similar system for Sinhala book readers. Below image show what kind of questions users ask from the groups.

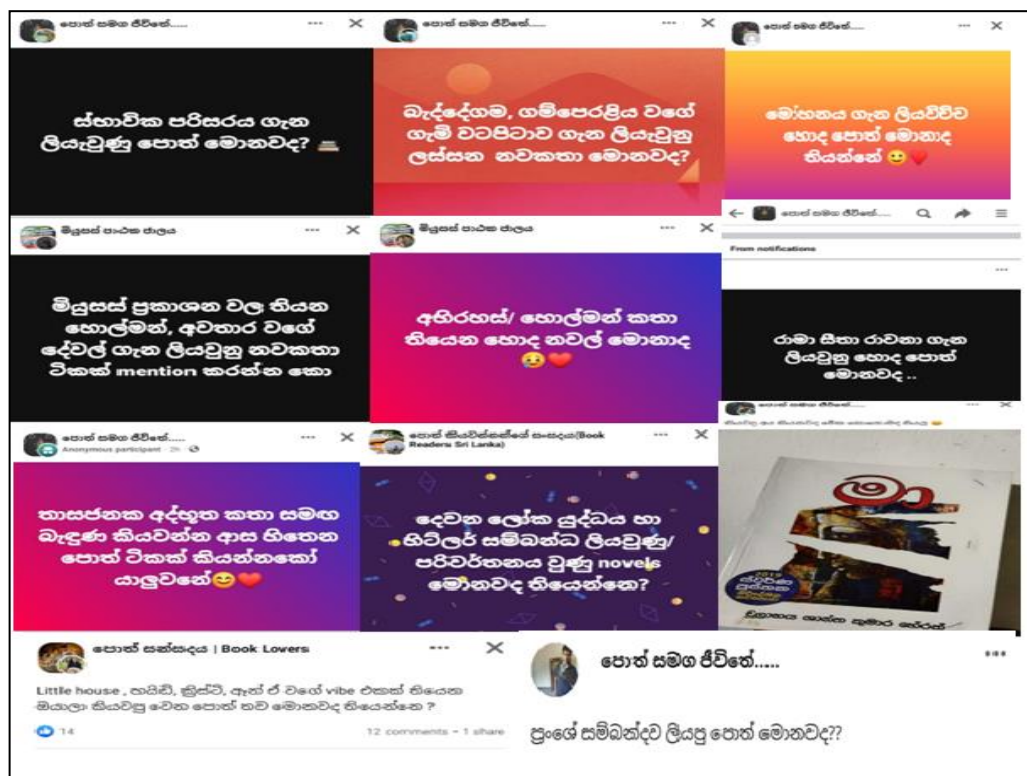


Figure 1. 1 : Questions Users ask from groups

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

Implementing an effective Sinhala book recommendation system for online users that rated a book list using the content and collaboration (hybrid) method 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 aim of this research is to analyze, design, develop and evaluate a Sinhala book recommendation system that effectively combines collaborative filtering and content-based methods to provide personalized recommendations to users.

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. Following are the objectives identified for the content and collaborative-based hybrid solution for Sinhala recommendation system.

1. Literature Survey – The literature survey is conducted to identifying existing systems implemented and their limitation. Many algorithms and methodologies along with evaluation criteria are studied to select the best approach to implement and evaluate the system.

2. Requirement analysis – study on the requirement to identify what is the outcome of the system and user expectation.
3. Design – based on the requirement the design is carried out along with architectural diagram. study of technologies that can be adopted to develop the system. Preparing dataset and design the web application is considered.
4. Implementation – a prototype is implemented based on the requirement and design. It is used tools and technologies identified in design phase. apply appropriate methodologies to recommend books
5. Testing – process of verifying the implemented system meets the end user expectation.
6. Evaluation – a proper evaluation criteria is identified. Both qualitative and quantitative approaches are conducted to evaluate the system.

1.6 Scope

1.6.1 In scope

1. Finding a data set which have user data along with rates and reviews for books.
2. Translating Sinhala reviews collected to English and applying sentimental analysis.
3. Identifying the textual features of Sinhala books that will be used in the content-based recommendation component.
4. Incorporating user-item interaction data for collaborative filtering.
5. Implementing a web application to interact with the system.

1.6.2 Out scope

1. Books other than Sinhala language will not be considered.
2. Books having less than 20 reviews and rate less than 5 will not be considered for recommendation.

1.7 Chapter Summary

The chapter included 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 chapter studies 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. As you all know 'Siri' in Apple utilize NLP to respond for user commands. As per the research (Shah, 2019) it explains that this type of recommendation system works with the data that is being provided by the user either by rating given to a product or by determining the nature of the sentence by using natural language processing (NLP).

Sentiment analysis is one of the main NLP based application that are widely used. It is the process of determining the sentiment or emotional behind a text. 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. Machine translation (MT) is another NLP application which is a 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. It is not believable that Spam detection also 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.

2.2.2 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.

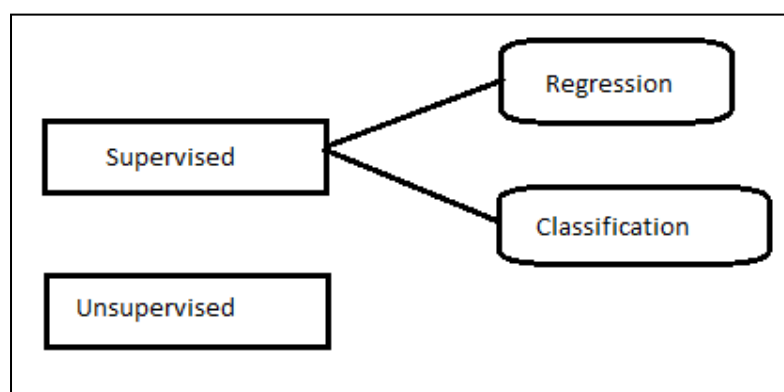


Figure 2. 1 : Machine Learning Models

In Supervised 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.

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.

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.

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.

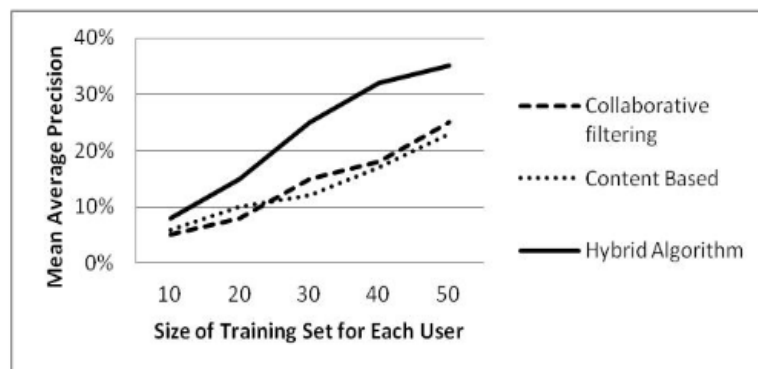


Figure 2. 2 : precision of different algorithm

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 drawbacks 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 included the existing system available along with tools and technologies used in these systems. As per the literature, most of the system have implemented specially for English books using either collaboration based or content based or combining both as hybrid in order to increase the accuracy and address the problems which encountered when implemented with only one model. The main limitations of the literature were less accuracy and the user reviews have not been considered for recommendation process. By addressing these limitations, a system will be designed and implemented with high accuracy which will be benefitted by all Sinhala book readers.

CHAPTER 3

METHODOLOGY

3.1 Chapter Overview

This chapter is discussing the methodologies that is used to implement the system. Initially the data set will be analyzed and preprocessed in order to make use in the application. The architectural diagram will be explained next along with the technologies selected for the implementation. As a part of preprocessing, step wise own implementation of sentiment analysis is discussed as there are limitations in available libraries. It will further explain the implementation details individually for identified modules such as item based collaborate filtering, content-based filtering and integration with the web application.

3.2 Data Set

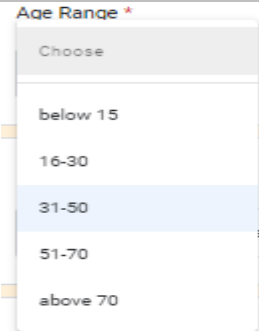
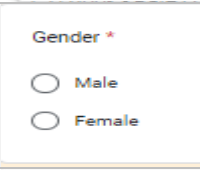
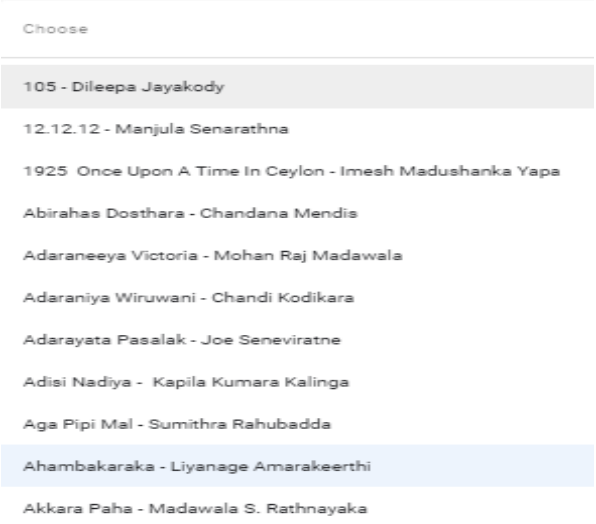
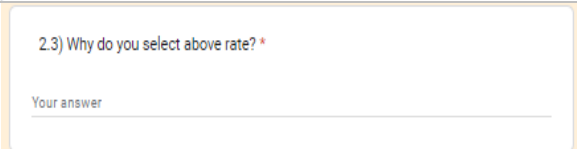
For any machine learning application, a dataset plays a significant role. While going through some initial process there were no data set found for Sinhala books from popular dataset providers like Kaggle. All those data providers have lot of datasets related to English language, but as mentioned there are no dataset available for Sinhala language. Therefore, a google form is created and shared in all book readers groups in Facebook and able to collect fair amount of data which can be used to build a model.

3.2.1 Validate Data Set

In the google form there are fields like rate and review along with books. 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 valid and suitable for train the model and predict the books.

Following fields are listed in the google form to be filled by the book readers. The table contains the usage of each field and the image of the field which is represented in the form. The full Questionnaire will be shown in Appendix A.

Table 3. 1 : Google form with fields details

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	

Initial stage of the implementation, an input text filed 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 additionally, 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.

3.2.2 Prepare 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 why?
20	nc21perer	Male	31-50	Rana Rala	9	It explains how to face problems and win as a t	Lohitha Pa	10	This is one of best detective series	Adaraneer	9	A great story in a village and our history	Senkottan	10	a story about a village and the way it written is	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 3. 1 : 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 3. 2 : Expected format to apply algorithms

3.3 Architectural Diagram

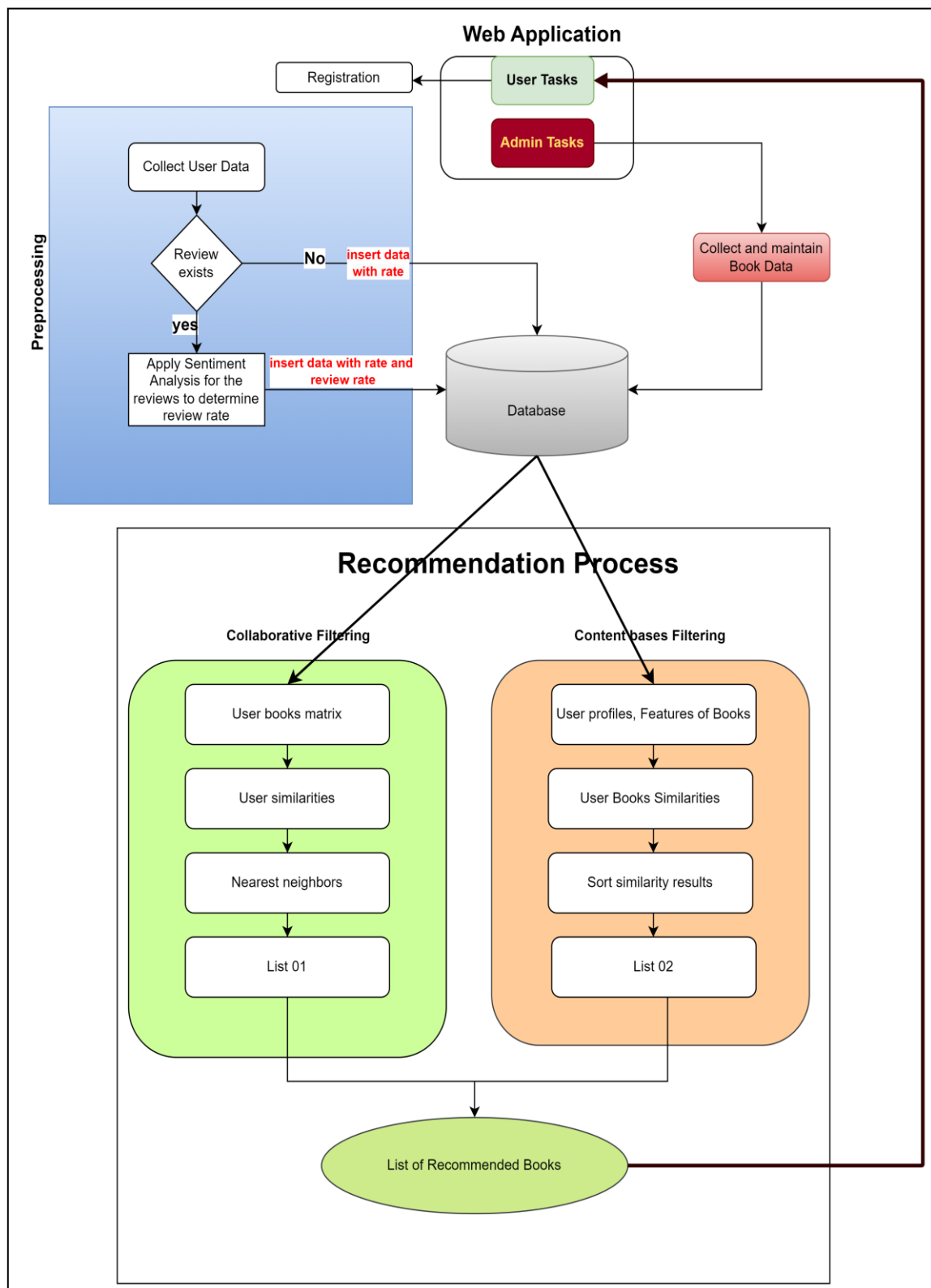


Figure 3.3 : Architecture Diagram

The architectural blueprint outlines a systematic process for handling data within a book recommendation system. Initially, data is gathered and subsequently categorized into

three distinct tables: Book_Details, User_Details, and Rate_Details. User-related information is stored in the User_Details table, while Book_Details contains additional book data such as ISBN, publisher, and year to facilitate book details retrieval during user searches. Ratings provided by users for books are stored in the Rate_Details table. In cases where a book review exists, a separate review rating is assigned post-sentiment analysis and incorporated into the Rate_Details table as a distinct column. The final rating calculation involves averaging the review rate and the normal rate, with the normal rate being utilized if no review is available.

Before commencing the recommendation process, data preprocessing is essential to eliminate unwanted entries like null rate values and instances where a maximum rate is assigned to just one book, as these books should not be recommended. Subsequently, algorithm application ensues, with one of the primary methods being Collaborative Filtering. This method aims to recommend books to users based on similarities identified from other users' preferences and ratings. User-based collaborative filtering involves constructing a matrix of users and their corresponding book ratings, followed by computing cosine similarity for each user in the matrix. Cosine similarity, a technique within the K Nearest Neighbor (KNN) algorithm, identifies clusters of similar users based on shared book ratings. Recommended books are then generated based on this similarity.

Alternatively, the Content-Based approach leverages book descriptions and user preferences to recommend similar books to those previously rated by the user. Initially, the system abstracts features like title, author, and genre of books, then constructs a user preference vector based on the books read and rated by the user. Candidate items are compared with the user's previously rated books to recommend the best matches.

While hybrid methods combining both Collaborative Filtering and Content-Based Filtering are prevalent, a strategy of applying these methods separately and then merging their results has shown to be more effective. Consequently, the combined book lists generated from both approaches are presented to users as the final recommended book list.

3.4 Apply Machine Learning Models

Even though many applications have been implemented using either collaborate based or content based, few applications were implemented using both in order to increase the accuracy.

As explained the data set contains user data, book detail and rate given by users. As the first step, the data set should be converted to user-based mastics. Then cosine similarity can be applied to find the recommend books. The process is known as applying collaborative filtering.

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. This is the process known as applying content-based filtering.

There are multiple ways we can apply hybrid methodology for the dataset. Finding books separately for collaborate and contend based and combine both results 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.

3.5 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 content 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.

3.6 Technology Selection

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

Table 3. 2 : Technology Stack

Front End	Logic	Data Persistence
   	    	 

3.7 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.

3.7.1 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	ඇත්තටම ආදරණීය කතාවක්.. ඒ වගේම පොඩි 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 හිතෙන 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 3. 4 : Review after converting to English

3.7.2 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 Crowdfunder 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.

3.7.2.1 Using Libraries

- **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 3. 5 : VADER result

▪ 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 3. 6 : TextBlob result

▪ 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.

3.7.2.2 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.

<code>data = pd.read_csv('kindle_reviews.csv')</code>				
<code>data.head()</code>				
	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...	Wiggelicious/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 3. 7 : Kindle Review Data sample

- **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()))
```

- **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()))
```

- **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)
```

- **Remove Numbers**

There were numbers also added in the review text 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)
```

- **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))
```

- **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.

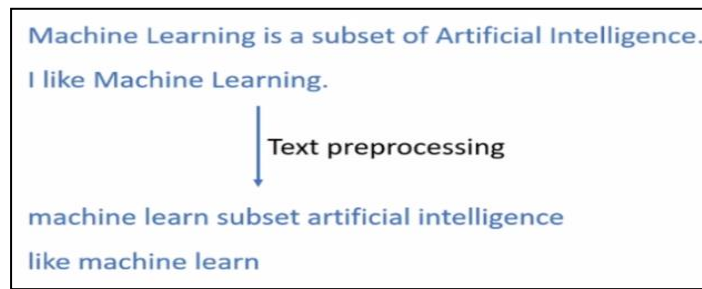


Figure 3. 8 : The way how the stemming is applied

▪ **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.

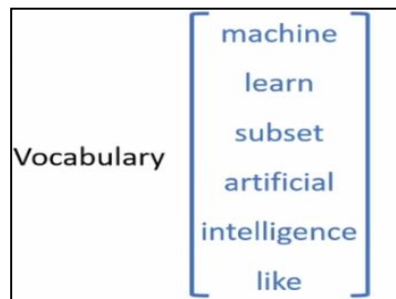


Figure 3. 9 : Build the vocabulary

▪ **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.

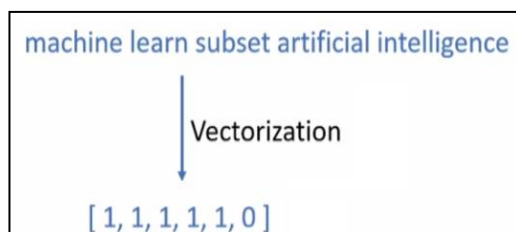


Figure 3. 11 : Vectorization 01

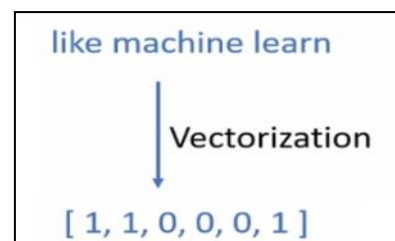


Figure 3. 10 : Vectorization 02

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. The function goes through all the vocabulary and perform the vectorization.

```
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. As in the diagram below each rate have fair amount of data divided

```

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 3. 12 : Balance Dataset

▪ **Model Training and Evaluation**

The next stage of the sentiment analysis process is building a model and evaluation. First it imports the libraries required and defined two functions for training and evaluation with accuracy, precision and recall values.

```

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}")

```

▪ 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)
```

3.7.2.3 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.

3.8 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.

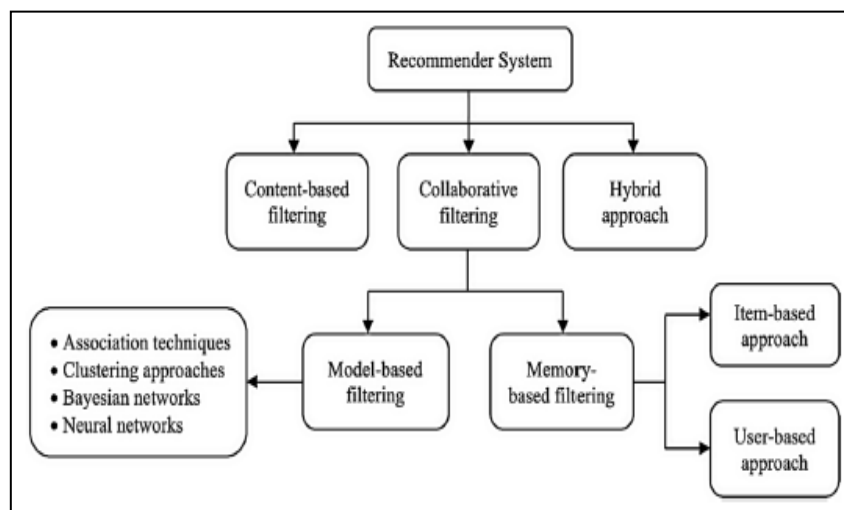


Figure 3. 13 : All models related to recommend system

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)

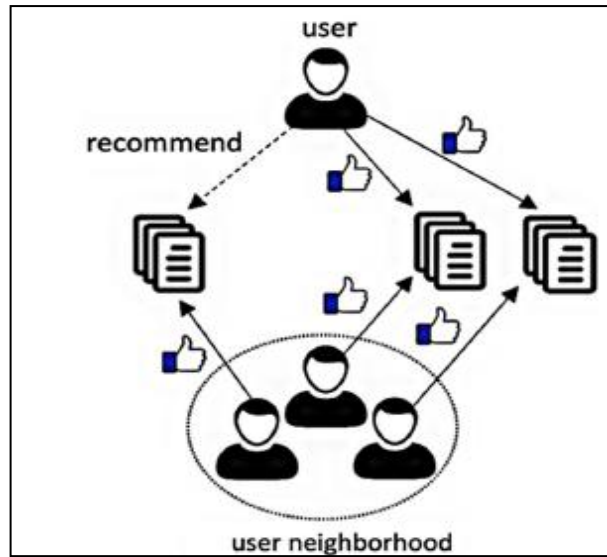


Figure 3. 14 : User based Collaborative filter

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)

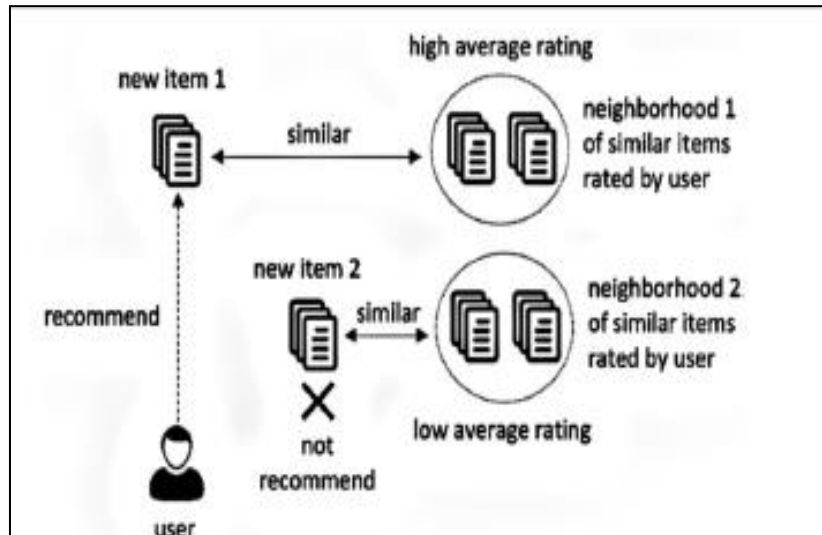


Figure 3. 15 : Item based collaborative filter

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

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

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.

```
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 as below.

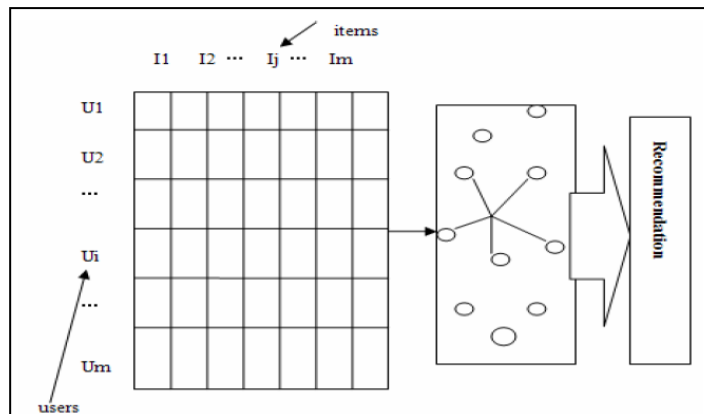


Figure 3. 16 : User vs Rate matrix

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

Above is the code snippet to implement the matrix and the result is displayed as below. It represents item-based matrix.

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 3. 17 : 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, as an example if we pass the book ‘apoiyawa’ it recommends list of books like ‘Adaraneeya Victoriya’, ‘Charitha Thunak’, ‘Ape Gama’ which are similar to provided book.

```
recommend("Apoiya - 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 3. 18 : Collaborative filter result

3.9 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
```

Figure 3. 19 : Content based recommended book list

3.10 Web Application

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

3.10.1 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 3. 20 : Login Page

When a user visit to the site, it first displays a login page as above. If the user has not registered yet, there is an option to register and make use of the system.

Top Rated Books

 ගොළු හඳවත කරුණාසේන ජයලත් කතර්‍යා ජරකාශන Votes - 26 Rating - 9.096	 සැරි මිනිසෙකුගේ කතාවක් දැඩිම මි රුද්‍රිල කෙසෙලි ප්‍රකාශකයෝ Votes - 20 Rating - 8.825	 කරු හඳවත ලියනගේ අමරකිකරී විද්‍යාර්තා ප්‍රකාශකයෝ Votes - 22 Rating - 8.795	 සිරි රහස් සලකුණ විතදන මෙන්ඩිස් විතදන මෙන්ඩිස් ප්‍රකාශන Votes - 27 Rating - 8.778
 විරිත කුනක් කේ ජයතිලක කතර්‍යා ජරකාශන Votes - 31 Rating - 8.565	 බිහිසුණු නිමිතය විතදන මෙන්ඩිස් විතදන මෙන්ඩිස් ප්‍රකාශන Votes - 30 Rating - 8.550	 මළුගිය ඇත්තෝ එදිරිවීර සරච්චන්ද්‍ර ගොඩගේ ප්‍රකාශකයෝ Votes - 40 Rating - 8.512	 විස්සාන සිහිනය එසුල් ශාන්ත සන්නස්ගල සංකිල් ප්‍රකාශකයෝ Votes - 23 Rating - 8.500

Figure 3. 21 : Top Rated Books

Above image displays the list of books having high average ratings. ‘Golu Hadawatha’ book has the highest and the rest of books are displayed as descending order.

Popular Books



12.12.12

මි.ජී.ඒ. ජයරත්න

මි.ජී.ඒ. ජයරත්න ප්‍රකාශන

Votes - 219

Rating - 7.970



අඟි යහලුවෝ

ඩී.ඩී. ඉලංගරත්න

සරසවි ප්‍රකාශන

Votes - 141

Rating - 8.110



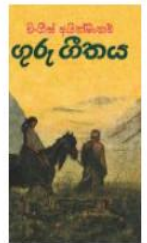
සෙංකොට්ටන්

මිහින්ද ප්‍රසාද මස්ඉඩුල

සන්තති ප්‍රකාශනය

Votes - 127

Rating - 8.280



ගුරු ගීතය

දැදිගම වී රුද්‍රිගු

ප්‍රගති ප්‍රකාශකයෝ

Votes - 98

Rating - 8.270



ඇමසෝනියා

මි.ජී.ඒ. ජයරත්න

මි.ජී.ඒ. ජයරත්න ප්‍රකාශන

Votes - 86

Rating - 8.233



ආදරණීය චිත්තෝර්ධයා

මොහාන් රාජ් මඩ්ඩල

බීසෝ ප්‍රකාශන

Votes - 86

Rating - 7.558



අසනග වැසි

දඟරන ශමී වීජේතිලක

සුසර ප්‍රකාශකයෝ

Votes - 79

Rating - 8.456



1925

ඉමේෂ් මඩුලංක යාපා

මි.ජී.ඒ. ජයරත්න ප්‍රකාශන

Votes - 71

Rating - 7.528

Figure 3. 22 : Popular Book List

Above image displays the list of books having high votes which is calculated by counting the number of users selected the book. '12-12-12' book has the highest vote and the rest of books are displayed as descending order. The book list contains an image of the book along with the Author, Publisher, number of votes and number of ratings. The popular book list does not consider only the number of votes but the rating above 5 is also another factor considered.

Recommend Books

-- Select --

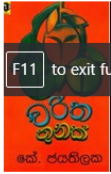
-- Select --

- 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 3. 23 : List of books user can select

Above image displays the list of books which can be selected in order to get similar recommended books.



වරිත කුනක්

Author- කේ ජයතිලක

Publisher - කතර්නා ජරකාගන

Categories - Novel, Fiction, Village

Description - මගේ වැටහීම අනුව, කළු අවිචිතයේ ජයතිලක, ඔහුගේ පාඨකයන් වඩාත් හොඳින් දන්නා පරිදි කේ. ජයතිලක, අද ඉහළම තලයේ සිටින නිමර්ශනාත්මක ලේඛකයන් හා බුද්ධිමතුන් අතරෙහි ලා ගිණිය හැකි වේ. ප්‍රමුඛත්වය අතින් ඔහු දෙවෙනි වන තෙක් වන්නේ මාට්ටින් වික්‍රමසිංහට පමණකි. ජයතිලකගේ නවකතා අතුරෙන් සියලු පාඨකයන්ගේ හඳවන සසල කළා වූත්, ඔහුගේ ශ්‍රේෂ්ඨතම කෘතිය හැටියට එයට සම්භාවිත වූත් නිමර්ශනය වරිත කුනක්ය. 1963 දී මුලින්ම පළ කරන ලද එහි ජනප්‍රියත්වය කොතරම් විශාල වූවත් එහි තේමාව හෝ අන්තර්ගතය පිළිබඳ කිසිදු අද්විතයක් මෙන්ම ප්‍රකාශයට පත් වී නොමැත. එය කියවන කියවන වාරයක් පාසා මුලින් ඇස හැසුණාට වැඩි යමක් දක්නට ලැබෙයි.

Review	Rate
Best novel	10
I like books that touch the subtle parts of the mind. This book is also a book that made me feel a lot. The writer has developed the character of Isa in an excellent way. It's nice how Isa's mind changes. Isa's character should be read and understood like Aravinda's character in Viraga. Also, this book shows the influence of traditional village ethics on human thoughts.	10
I love the story	10
එකම පවුලක සහෝදරයින් තුන්දෙනෙකුගේ විවිධාකාර වූ මනෝභාවයක් පෙන්වූ කතර කතාවක් නිසා	10
Kind of a sad story about a men who scarifies his live on his entire family and nothing gain at the end	10
සරල හැමි ජීවිතයේ සංකීර්ණත්වය පිළිබිඹු වන කෘතියක් බැවින්	10
කුඩාකල සිට පවුලේ සහෝදර සහෝදරියන් සමඟ ගෙවියන ජීවිතය වැඩිමහල් සහෝදරයාගේ සිතූම සැතපුම් හා අවසානය	10
Life story from another corner	10

Figure 3. 24 : Selected book details with reviews and rate

Above image displays the details of the selected book along with review and rate other users given for the book user selected. So, a user can view the review and rate and decide whether the book is as per his preference.

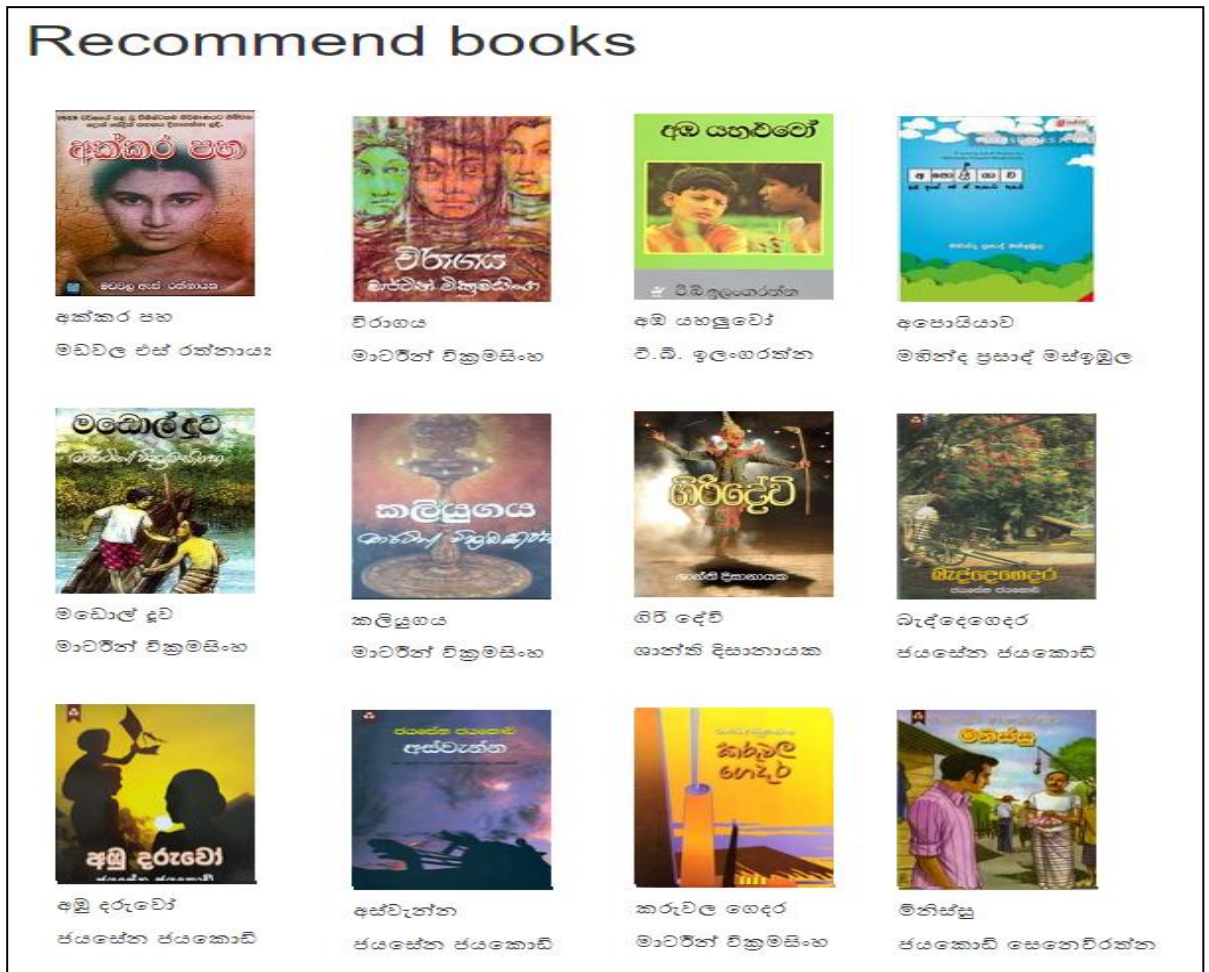


Figure 3. 25 : Recommended Books

The main output of the entire application is depicted in above image. When ‘Charitha Thunak’ book is selected as an example, it gives above book list which are selected by applying machine learning algorithms. Since the selected book is category type of village, fiction and novel, all the novel, fiction and related to village books are displayed as a result.

3.10.2Authenticate

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.

3.11 Database

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.

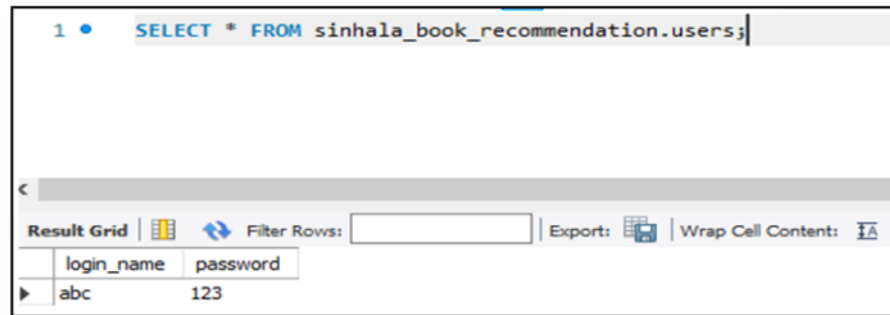


Figure 3. 26 : User stored data

The above user table contain all the data related to users who are registered to the system. The column contains username and password which is used to login to the system.

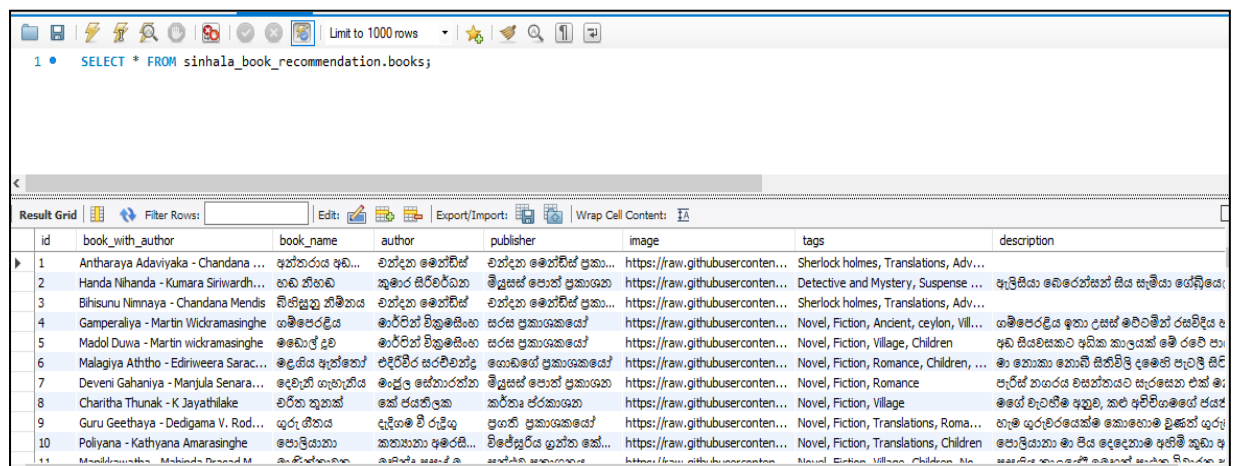


Figure 3. 27 : Saved book details

The above book table contain all the data related to books. It has book name, author, image url, publisher as well as tags which are used for content-based algorithm to predict the book list.

3.12 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 recommendation system.

4.2.1 Accuracy Metrics

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

- **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.
- **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.
- **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.

- 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.

- **Click Through Rate (CTR)** - It calculates the percentage of users who clicked on a recommended book.
- **Conversion Rate** - It calculates the percentage of users who buy and read the recommended book
- **Bounce Rate** - It calculates how many users ignore or not accept the recommended books without interacting the any results.

4.2.3 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.

4.2.4 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.

4.3 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, user data will be used and a standard machine learning training-test split will be applied in order to learn and train the model for the evaluation. Mean Absolute Error (MAE) or Root Mean Square Error (RMSE) will 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 implemented system do not have access to the live system, it will be focused 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 the target is to get as close as possible.

In order to validate the model further, expert authors will be contacted. The author 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.

4.4 Test Results

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

4.4.1 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 to be read.

4.4.2 Verification of Functional Requirements

Table 4. 1 : Functional Requirements

Id	Requirement	Priority	Status
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

4.4.3 Qualitative Evaluation

In order to evaluate the quality of the application, the system is shown to the domain experts (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. 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.

▪ 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.

Table 4. 2 : Famous Authors list contacted

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

1. The Famous Author Confirmation – Dileepa Jayakody

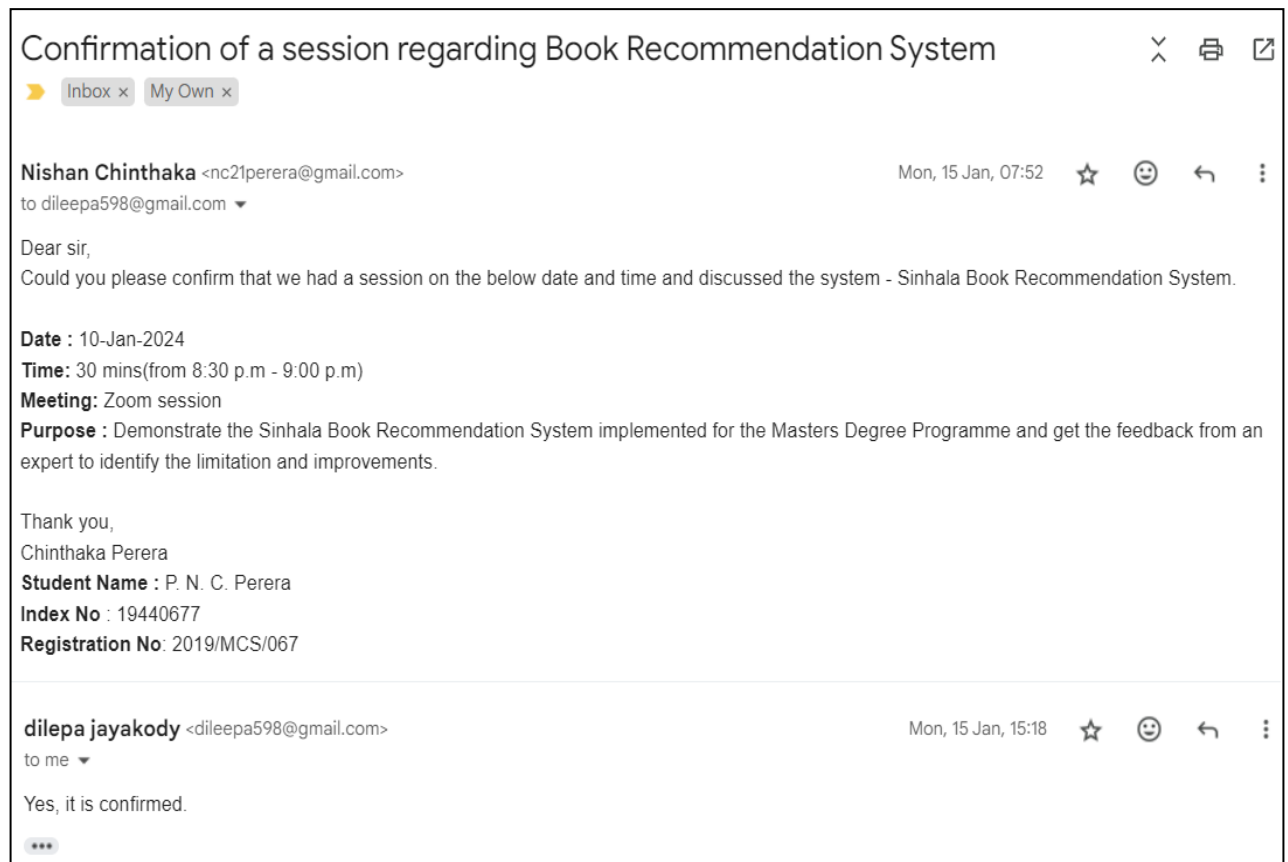


Figure 4. 1 : Email Confirmation - Dileepa Jayakody

The above email confirmation received from the famous Author Dileepa Jayakody. A zoom meeting was scheduled and the application which have been implemented was demonstrated. He was excited about the system and highlighted some features to be include to the system.

2. The Famous Author Confirmation – Shamel Jayakody

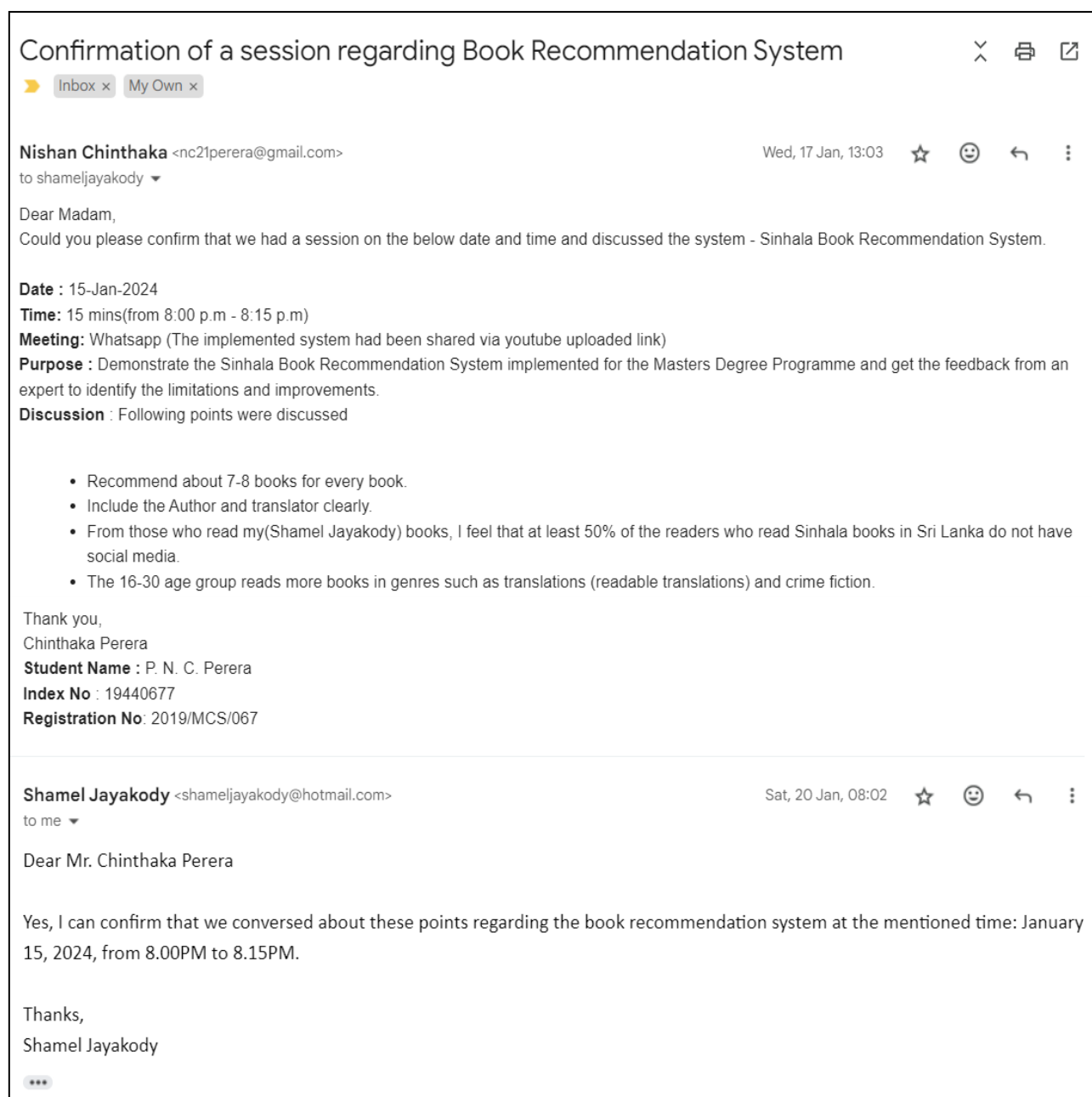


Figure 4. 2 : Email Confirmation - Shamel Jayakody

The above is the email confirmation received from the famous Author Shamel Jayakody. A recorded demonstration about the system implemented has been shared with her to check. She also pointed out some features that should be available in the system.

3. The Famous Author Confirmation – Nethindu Warapitiya



Figure 4. 3 : Facebook respond after sharing the YouTube link of the implements system

Nethindu Warapitiya gave the feedback for the system implemented via Facebook chat as above.

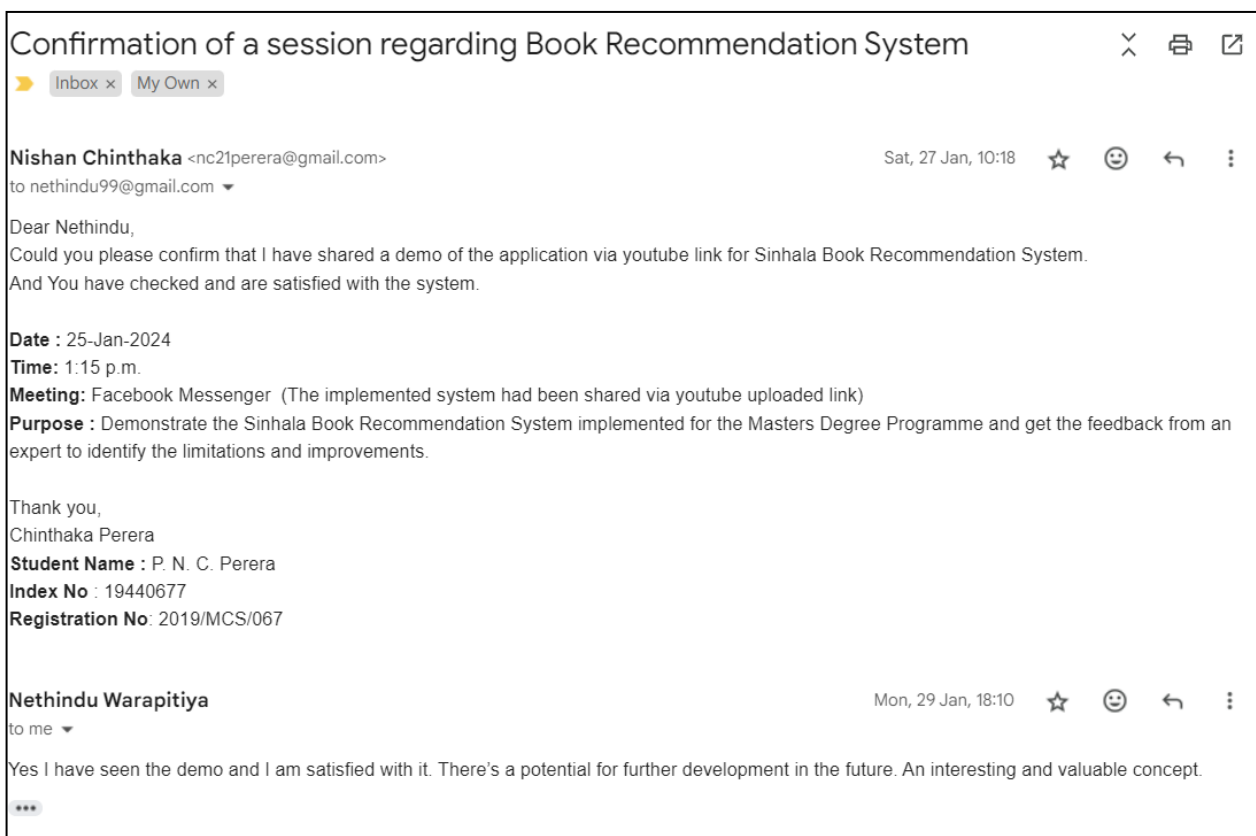


Figure 4. 4 : Email Confirmation - Nethindu Warapitiya

The above is the email confirmation received from the famous Author Nethindu Warapitiya. He also gave a good feed back about the system implemented

▪ 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.

Table 4. 3 : Feedback from Technical experts

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

- **Feedback from Book Readers**

Table 4. 4 : Feedback from Normal 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 displays already read books. if you can improve to track and remove books the user already read

4.4.3 Quantitative Evaluation

4.4.3.1 System Calculation

- **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.

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

Figure 4. 5 : 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.

```

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 4. 6 : Invoke the MAE method for all data

- **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.

- **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

Table 4. 5 : Accuracy calculated for 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 Karunaratna	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%
Total Average		80%

4.4.3.2 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.

Table 4. 6 : Feedback of online survey

	Question	Positive Count	Total	Percentage
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

4.5 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

Around the world, there are so many items available and recommended systems provide a facility for users to recommend item they prefer. Even though there are many English book recommendation system implemented, Sinhala book recommendation system has not been implemented for Sinhala book readers.

The proposed system can be used by all book readers to recommend Sinhala books along with reviews and rate. Based on the machine learning techniques, the system was implemented and the evaluation results show that this collaborative and content-based book recommender system is effective and accurate for Sinhala book recommendations.

The aim of the research was ‘to analyze, design, develop and evaluate a Sinhala book recommendation system that effectively combines collaborative filtering and content-based methods to provide personalized recommendations to users’ and that was successfully achieved and further the implemented system met all the objectives defined to a greater extend.

5.3 Limitation

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.

5.4 Future Enhancements

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

1. The system is implemented as a web-based application in local machine; for further work, this system can be made available for all users by deploying to a web server.
2. create user roles for admin user. An admin user would be able to enter and update book details and all manipulate data which only administrator role authorized to perform.
3. Implement a mobile application that offer Sinhala book recommendations on-the-go, with features such as barcode scanning to identify physical books and receive relevant suggestions.
4. Enhance the application to ensure that the recommendation system is accessible to users with disabilities, with features such as text-to-speech support for visually impaired readers.

By incorporating these enhancements, you can create a robust and user-centric Sinhala book recommendation system that effectively serves the needs and preferences of Sinhala book readers.

5.5 Chapter Summary

The conclusion chapter discussed whether all of the aim and objectives which were defined have been successfully achieved. Furthermore, the chapter discuss the limitations and future enhancements that can be included to the project.

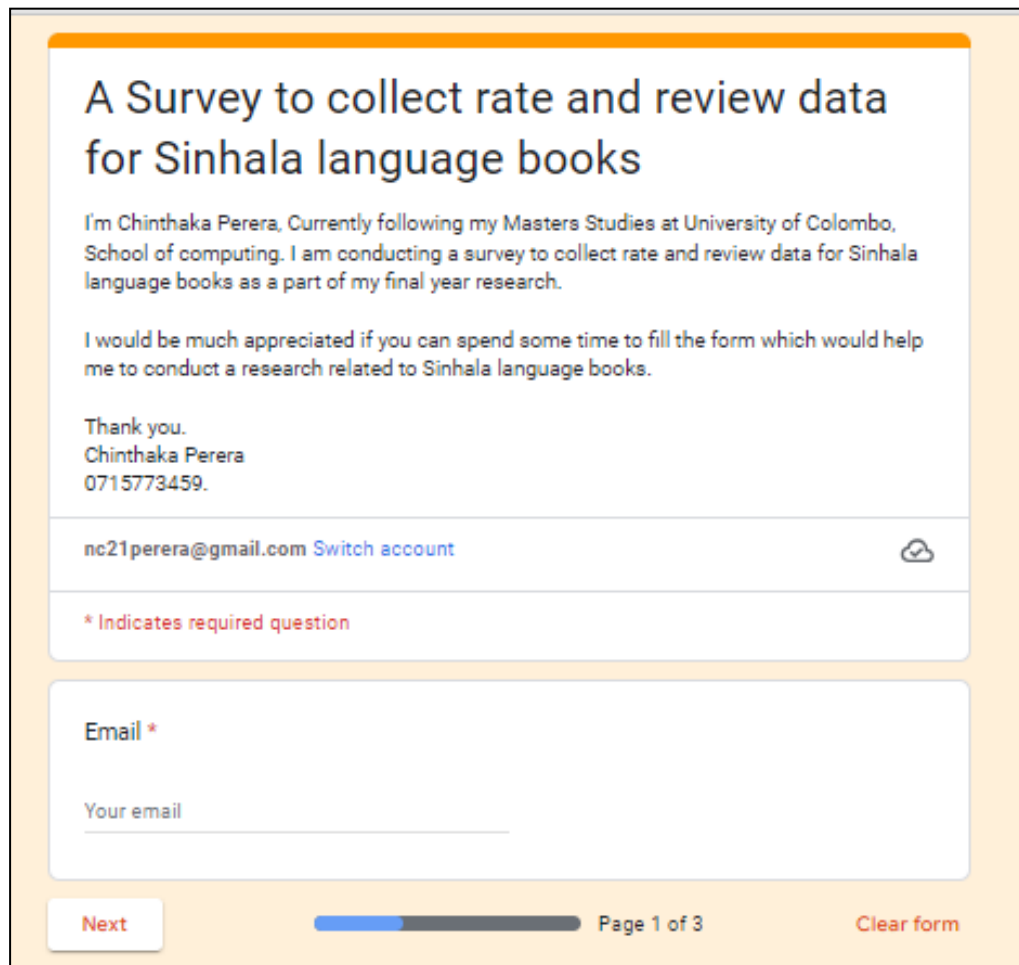
APPENDICES

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-
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 the form header, there is a text box containing "nc21perera@gmail.com" and a "Switch account" link with a small cloud icon. Below the header, there is a red asterisk followed by the text "* Indicates required question". The main content area of the form has a white background and contains a question labeled "Email *" in bold. Below the label is a text input field with the placeholder text "Your email". At the bottom of the form, there is a "Next" button on the left, a progress bar in the center showing "Page 1 of 3", and a "Clear form" link on the right.

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

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

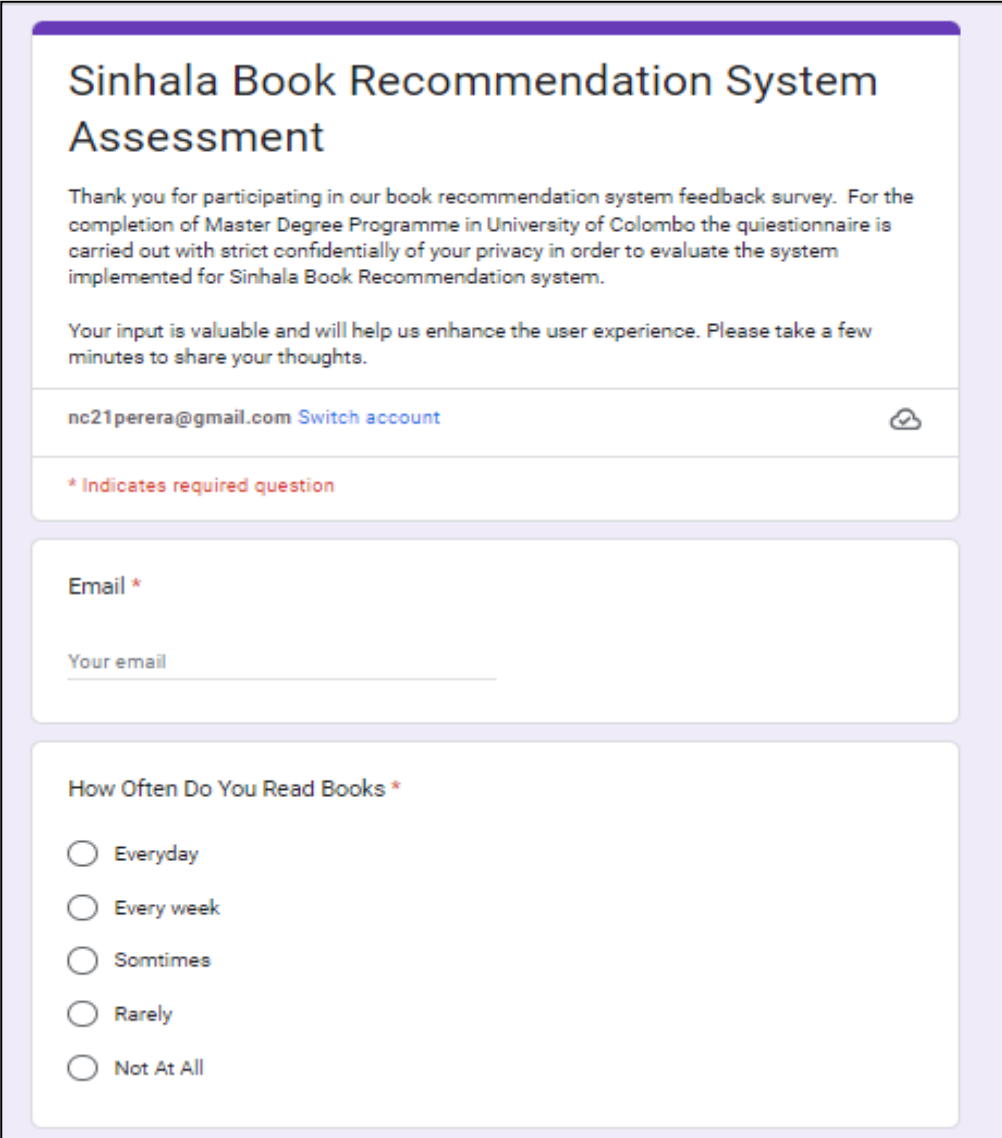
[Clear form](#)

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

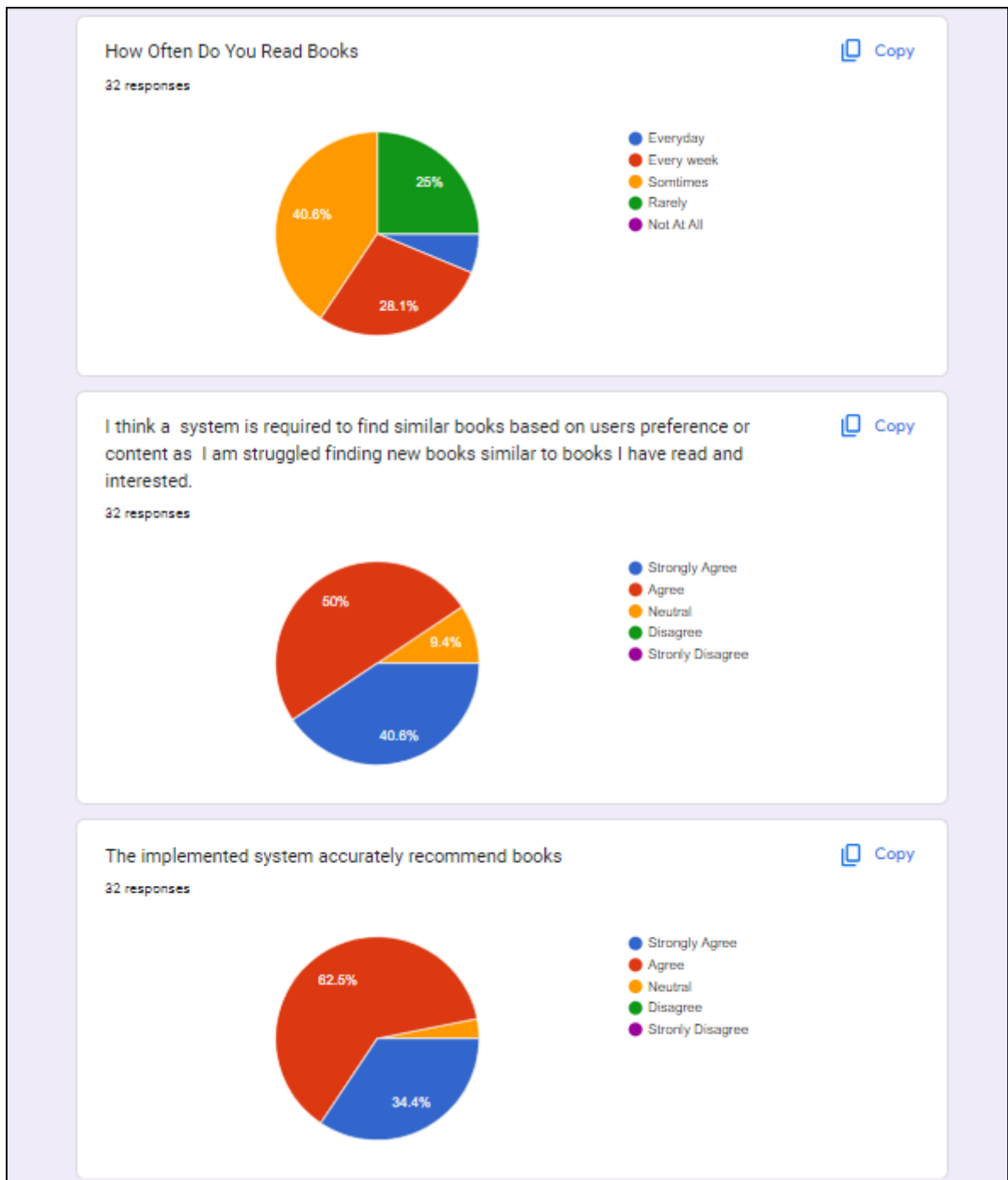
Thank you for completing our feedback questionnaire. Your input is valuable in helping us enhance the book recommendation system. If you have any additional comments or suggestions, please feel free to share them. Your feedback is greatly appreciated!

Submit

Clear form

Never submit passwords through Google Forms.

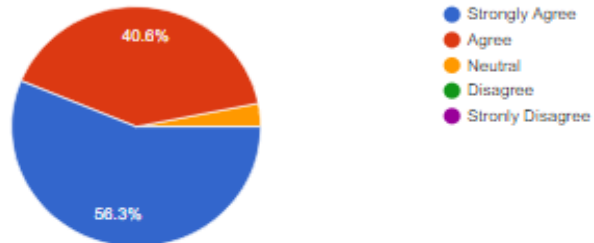
Appendix - C



The system helped me to find new books

 Copy

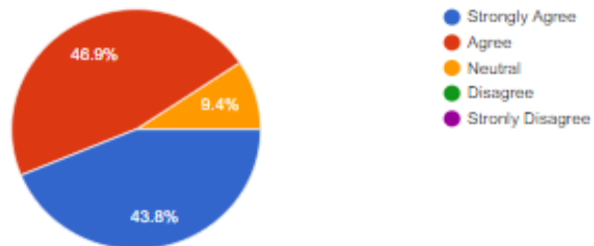
32 responses



It is easy to navigate and use the system

 Copy

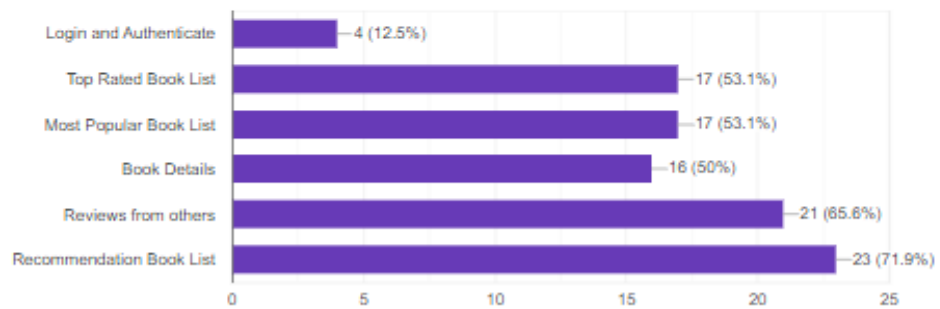
32 responses



What is the most attractive feature of the application

 Copy

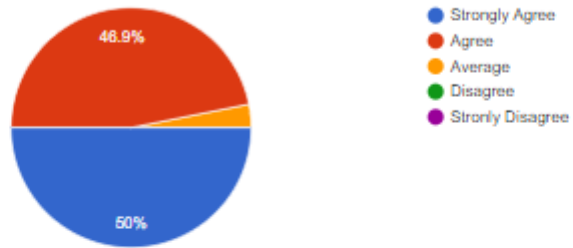
32 responses



Overall, I'm satisfied with the recommender system

 Copy

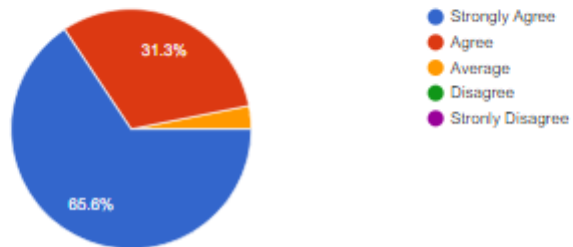
32 responses



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 Recommendation 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.