Book Recommendation System Using Deep Learning (GPT3)

Harsh Dubey[1], Dr. Suma Kamalesh Gandhimathi[2]

Under The Guidance of, Dr. Suma Kamalesh Gandhimathi, Assistant Professor, Lovely Professional University, Punjab, School Of Computer Science and Engineering

---***---

Abstract

Books have been an important part throughout different phases of peoples' life. They serve as a source of knowledge, entertainment, stress relief, and most importantly motivation to do better in one's life. Even though books are so important and choosing the correct book can be the difference between having a good and productive time reading it to getting frustrated at the halfway mark over the unnecessary expense made at purchasing that book and the time lost over getting to that point, from getting to actually learn something and get inspired to work towards your goals to getting back to square one i.e., going to the book-choosing step. This is where recommender systems step in. The main aim of this system from the very beginning has been to provide a simple, minimalistic yet highly functional, accurate, and information-rich interface for the user. This book recommender not only returns with book names and their cover page but also tries to embed various other useful information which can be the deciding factor in choosing a correct book, information such as a small description, preview of the book (if available), number of pages, etc.

Keywords: Book Recommendation System, GPT3, Google Books, Streamlit, Python, Deep Learning

Introduction

Recommender Systems have been a successful way of tackling issues where a suggestion is required for the viewer, listener, or reader. They even assist buyers by suggesting the products that they might be interested in. Through this project we have tried to create a recommendation system that will help readers by suggesting a book that they have already read and would like to read a similar book. Many recommender systems take user ratings into account for suggesting a book for the user but then those books are not highly related to the book that customer read, they are the books that other users who read the same book liked. Booksellers and libraries often struggle with space for books and with correct categorization of the book as they get many books which are not even in demand for their readers.

This Book Recommendation System using Deep learning (GPT3) project has been an effort to take all these situations into account and then search for a better solution. Much like every other recommendation system available online for different tasks, this project serves as a tool to recommend books like what the user has already read and input its value into the form. This project can serve as the main tool for the customer, shopkeeper, librarian, and the people visiting the library to help them suggest, search, organize and even buy books.

Literature Review

This recommender system uses GPT3 at its core and is assisted by python and streamlit to handle the creation of front-end and to establish the connection between front-end and the openai api along with the connection for google books api. GPT-n is developed, maintained, and offered by researchers at openai who came out with an idea that combining the transformer into a model which would be pretrained with a lot of data points and could be fine-tuned for later usage in specific models. It involves the usage of the transformer, which is an architecture used for transforming one type of sequence into any other with help of decoders and encoders. It also involves a self-attention mechanism which means that it weighs differently the significance of different parts of input data and can learn the context of actual sentences by using this method.

GPT3 is the latest iteration of the generative pre-trained transformer. It is an autoregressive language model, and it can produce human-like texts using deep learning. It has trained over more than 1.75 billion data points and offers a lot of functionalities out of the box such as chatbot, currency or language conversion, questions and answers, text completion such as when asked about writing a tagline for an ice cream shop, it may respond something like 'Serving scoop full of smile'.

Problem Definition

To create a book recommendation system for users through which users can enter a book and get similar book recommendations.

Aims and Objectives:

The book recommendation system should:

a. suggest similar books to the book or topic entered by user
b. provide a clean and minimalist user-interface to avoid any distractions
c. should suggest at least three books for every book entered by user
d. provide various information such as author name, page number, preview link for the book etc.
e. it should also allow users to such other books without refreshing the page

Also, it should be:

a. portable to use on any browser across multiple devices
b. efficient and require less resource and time to suggest books
c. scalable to accommodate newer features
d. reusable to make it as a base for other recommender systems
e. fluid to make transitions from one view and page to another not look jittery and laggy.

Proposed System

This project is made to provide accurate book recommendations based on the content of books that the user enters. It fetches at least three books along with other details associated with these books such as page number, a brief description of the book, name of the author, cover page, and a link to view the preview of the book.

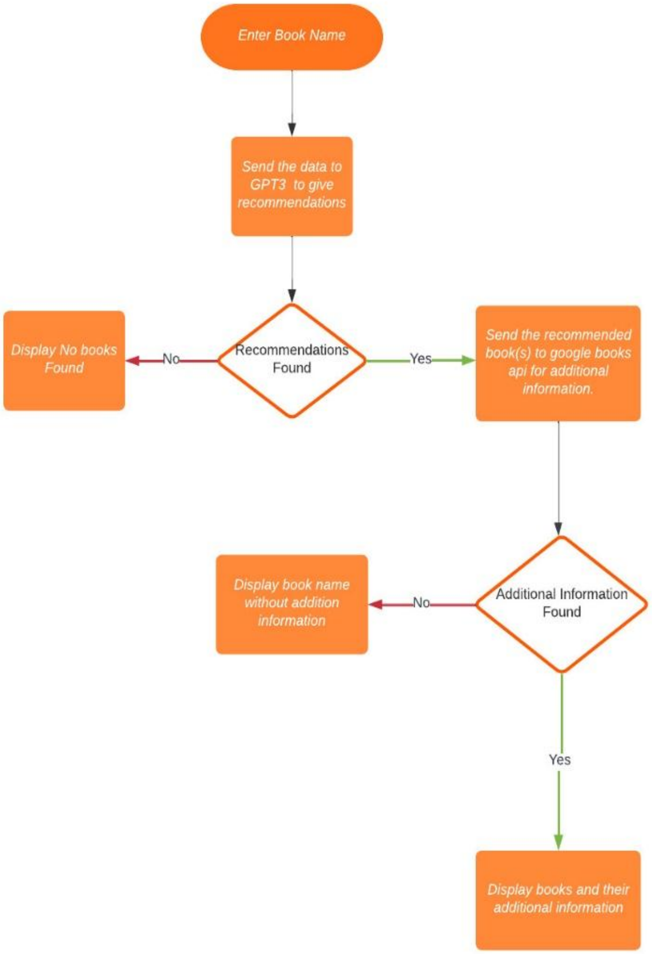
For this task, we proposed a system taking suggestions from the last research paper we went through in the literature survey where the author suggested the usage of GPT (Generative Pre-trained Transformer) for better semantic analysis and recommendation. On the landing page of our app, the user is asked to enter the name of the book that he read recently, or he liked the most, then we pass that information through the API generated by finetuning of the GPT model. This results in getting at least three recommended books. These books are then passed through another API, this time google books API to get additional information about those books. Then all the data is structured in form of expanders (dropdowns) to show the information in a user-friendly manner. The user can again search for another book as the search books is kept intact on the front page without a refresh.

Methodology

Modules used in Implementation

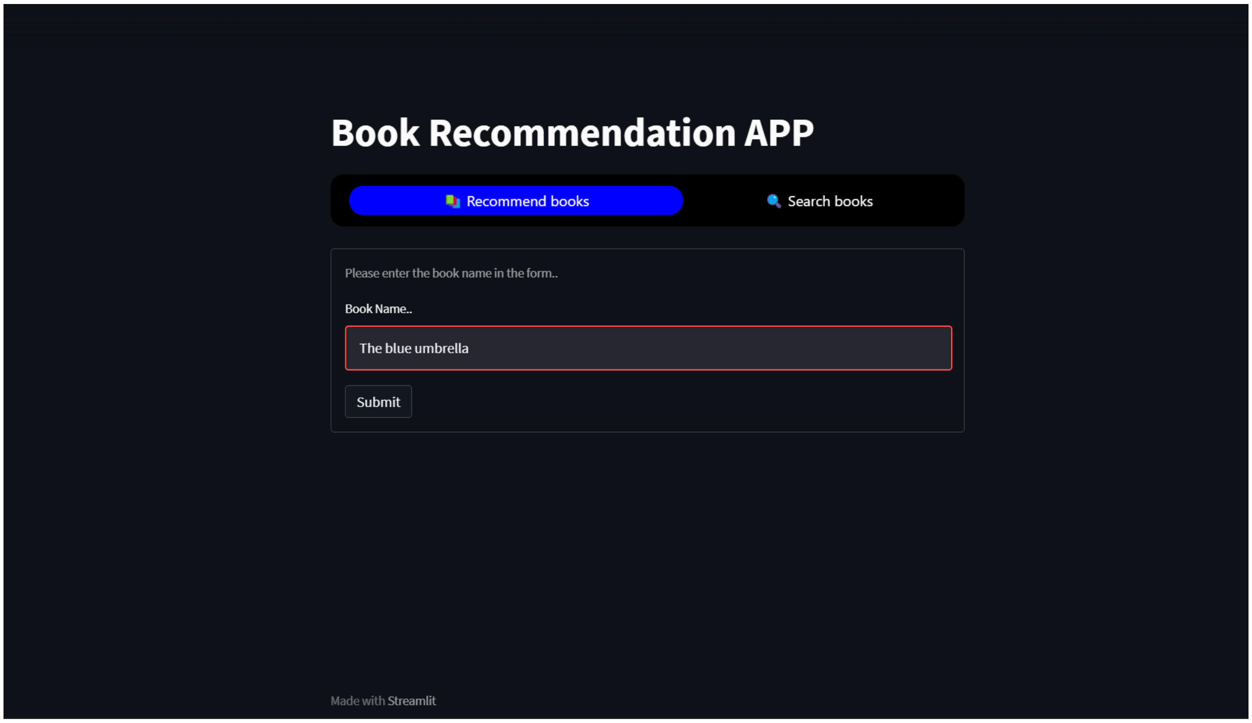
1. Streamlit (front-end)
2. Hydralit (front-end)
3. Python (for backend)
4. Openai API (for recommendation)
5. Google Books API (for addition information on books)

Flow-chart of the proposed system:


Enter Book Name
Send the data to GPT3 to give recommendations
Recommendations Found
No
Display No books Found
Yes
Send the recommended book(s) to google books api for additional information.
Additional Information Found
No
Display book name without addition information
Yes
Display books and their additional information

Results and Discussion

This project results in a great tool for searching a similar book to a book that the user has enjoyed reading. It eliminated many issues which are present in almost every legacy book recommendation system which is the result of them sticking to ratings instead of semantic analysis of books or analysis of genre of books that they are recommending. Some Screenshots of how the model recommends books is attached below:



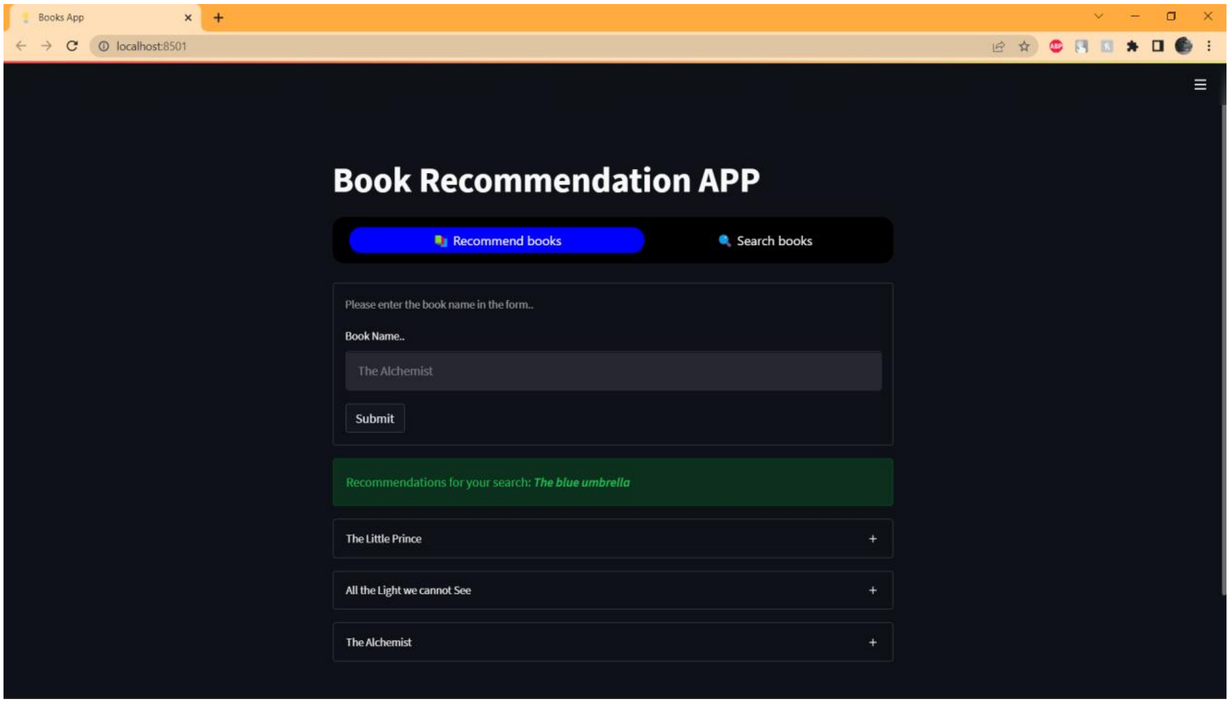
Book Recommendation APP

Please enter the book name in the form..

Book Name..

The blue umbrella

Figure 1. First Screen



Book Recommendation APP

Please enter the book name in the form..

Book Name..

The Alchemist

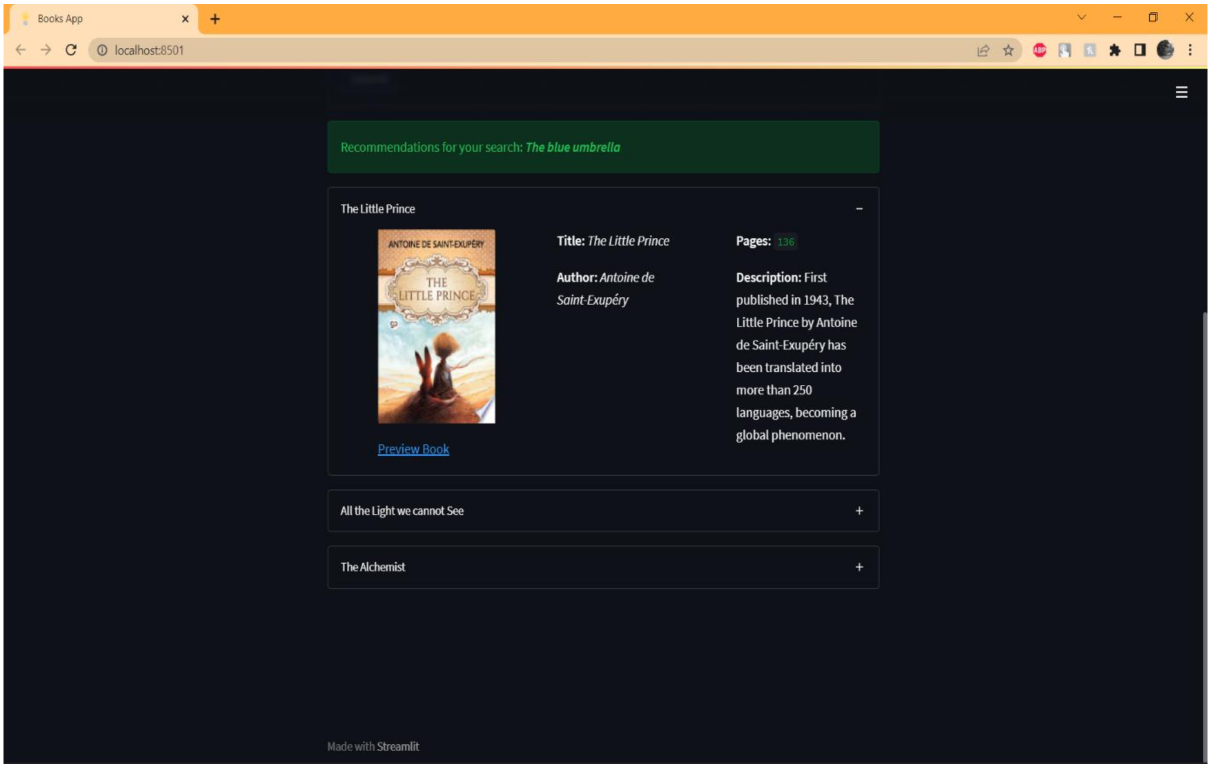
Recommendations for your search: *The blue umbrella*

The Little Prince +

All the Light we cannot See +

The Alchemist +

Figure 2. Results in minimized window



Recommendations for your search: ***The blue umbrella***

The Little Prince

Title: *The Little Prince*

Author: *Antoine de Saint-Exupéry*

Pages: 136

Description: First published in 1943, The Little Prince by Antoine de Saint-Exupéry has been translated into more than 250 languages, becoming a global phenomenon.

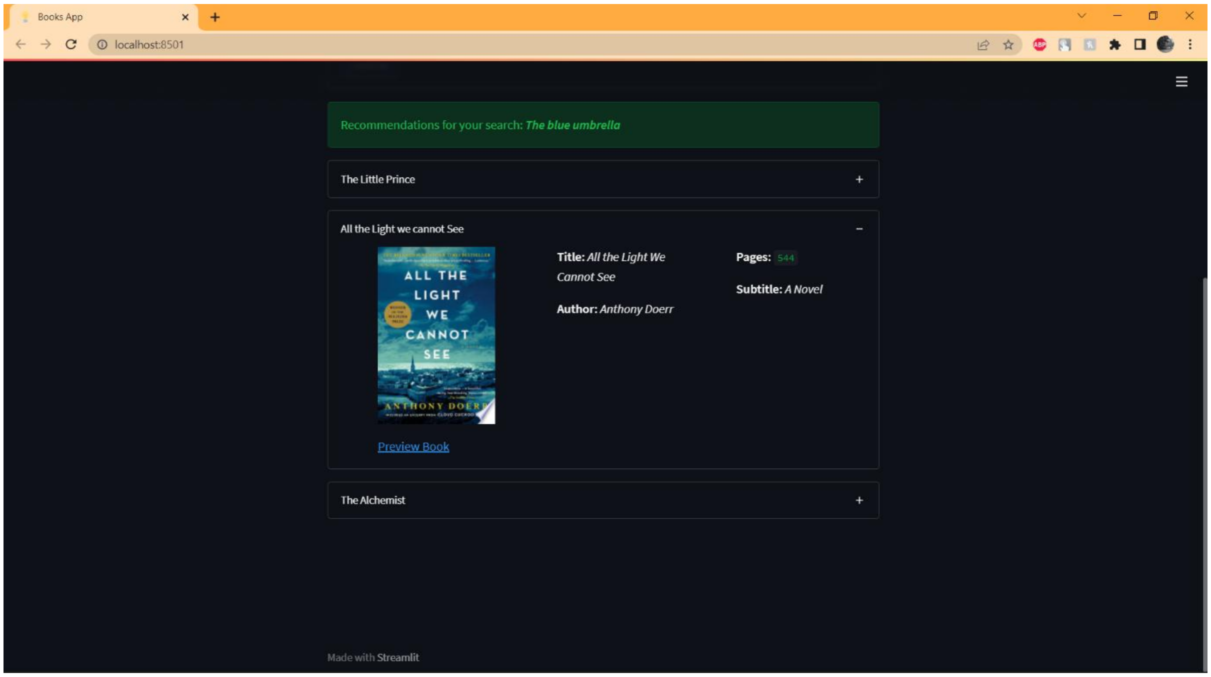
Preview Book

All the Light we cannot See

The Alchemist

Made with Streamlit

Figure 3. First Book Details



Recommendations for your search: ***The blue umbrella***

The Little Prince

All the Light we cannot See

Title: *All the Light We Cannot See*

Author: *Anthony Doerr*

Pages: 544

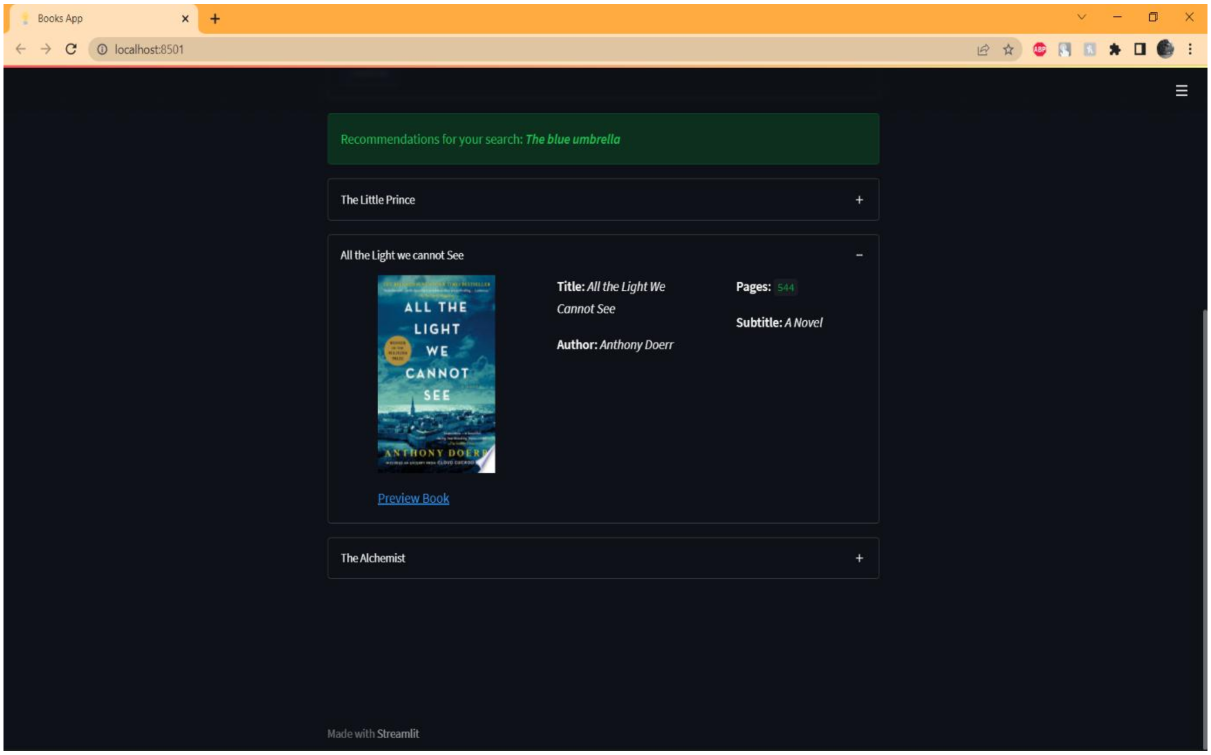
Subtitle: *A Novel*

Preview Book

The Alchemist

Made with Streamlit

Figure 4. Second Book Details



Recommendations for your search: *The blue umbrella*

The Little Prince +

All the Light we cannot See −

Title: *All the Light We Cannot See*

Author: *Anthony Doerr*

Pages: 544

Subtitle: *A Novel*

Preview Book

The Alchemist +

Made with Streamlit

Figure 5. Third Book Details

Conclusion

This recommendation system has been extremely reliable as during experimental testing, we found the range of ratings for the shown recommendations were never less that 5 out of 10 and sometimes the model even achieved a full rating of 10. This shows that the model is capable of being a standalone project on its own which will serve users reliably and consistently, but the major drawback is the cost incurred in the usage of GPT3. While we can reduce the token that we feed the model to correctly, this may also result in hampering the accuracy of model. Developing an indigenous solution for this problem like scraping a large book dataset and categorizing it in a genre-based manner could be a solution but that would need to be updated on a regular basis for better recommendation.

One other drawback that the system is currently facing is the usage of google books api for searching book information. While this gives a good amount of information related to the books, many times important data related to book are not available like cover photo, pages. Getting this data from Amazon books api could have yielded some more data such as price of the books, better customer ratings and with a lot more books available for selection.

Even after all these drawbacks and suggestions that can be incorporated to the project, this project has the potential to provide a great platform for people and institutions which require a lot of book recommendations to work and to sell better books to the customers. It can even help libraries to keep relevant books for better storage and to minimize the cost incurred in buying these books. People can buy books in the same content and genre range that they enjoy without getting stuck with a book that may not fulfil their requirements

Acknowledgement

I would like to take this opportunity to thank my mentor Dr. Suma Kamalesh Gandhimathi for assigning me this capstone project and helping throughout to complete it.

I have not only gained an exposure in a project development, but I have also learned a lot of new developing technologies in my area and how to do research for development of a project.

References:

1. H. Zhang, T. W. S. Chow and Q. M. J. Wu, "Organizing Books and Authors by Multilayer SOM," in IEEE Transactions on Neural Networks and Learning Systems, vol. 27, no. 12, pp. 2537-2550, Dec. 2016, doi: 10.1109/TNNLS.2015.2496281.

2. Kurmashov, Nursultan & Latuta, Konstantin & Nussipbekov, Abay. (2015). Online book recommendation system. 1-4. 10.1109/ICECCO.2015.7416895.

3. E. Giannopoulou and N. Mitrou, "An AI-Based Methodology for the Automatic Classification of a Multiclass Ebook Collection Using Information From the Tables of Contents," in IEEE Access, vol. 8, pp. 218658-218675, 2020, doi: 10.1109/ACCESS.2020.3041651.

4. Aygun, Celalettin & Yildiz, Oktay. (2016). Development of content based book recommendation system using genetic algorithm. 1025-1028. 10.1109/SIU.2016.7495917.

5. H. Zhao, H. Wu, J. Li, H. Zhang and X. Wang, "Dtree2vec: A High-Accuracy and Dynamic Scheme for Real-Time Book Recommendation by Serialized Chapters and Local Fine-Grained Partitioning," in IEEE Access, vol. 8, pp. 23197-23208, 2020, doi: 10.1109/ACCESS.2020.2968220.

6. E. Giannopoulou and N. Mitrou, "An AI-Based Methodology for the Automatic Classification of a Multiclass Ebook Collection Using Information From the Tables of Contents," in IEEE Access, vol. 8, pp. 218658-218675, 2020, doi: 10.1109/ACCESS.2020.3041651.

7. https://arxiv.org/pdf/1706.03762.pdf- attention is all you need

8. https://www.analyticsvidhya.com/blog/2021/06/build-book-recommendation-system-unsupervised-learning-project/

9. https://thesai.org/Downloads/Volume12No1/Paper_26-Personalized_Book_Recommendation_System.pdf

10. https://cs.carleton.edu/cs_comps/1617/book_rec/final-results/paper.pdf