BCIS 5140 Final Project Report

E-book Recommendation A.I. System for an Online Library Pay-to-Read Platform

By Project Team 9:

Yusuf Saadu

David Ajiga

Soundarya Nidhi Sabanayagam

Prabandha Naraboiena

Table of Contents

Executive Summary

This project tackled the problem of subscriber retention and site traffic faced by an online novel library platform based in Singapore called Webnovel. They possess a vast and growing library of e-books that they license and provide avid readers access to while generating revenue through ads, monthly subscriptions, and site currency purchases to unlock chapters. The advent of the internet has pushed forward the e-book as a viable medium to profit from and the company wished to secure its share of the growing market, however, had a user retention issue.

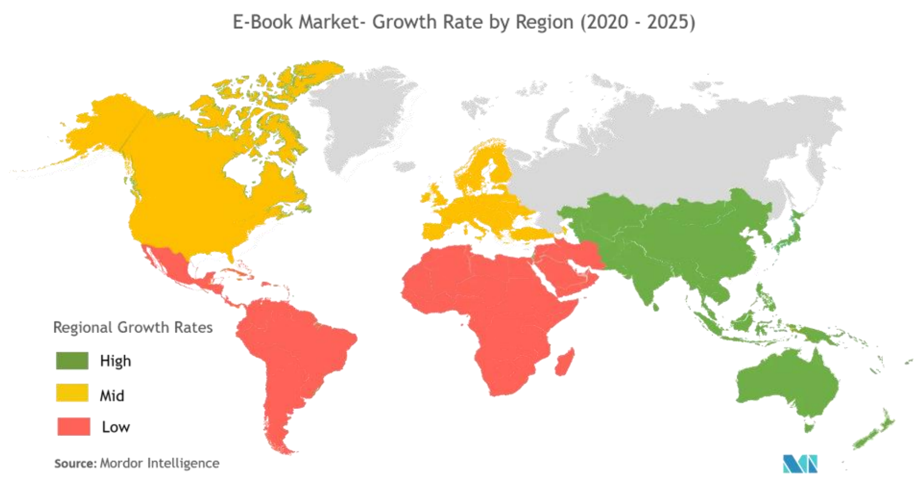
The problem they faced was in securing site traffic and retaining subscribers, an issue we felt could be solved by raising the customer experience by providing engaging novel suggestions for them to read that would keep them on the site. The more the readers enjoyed the novel recommended to them, the more they'd want to stay on the site and possibly spend money on subscriptions, chapter purchases, and platform currency exchanges. To solve this issue, our team undertook the project of developing a book recommender A.I. algorithm to replace the subpar one they currently had. The algorithm was based on users' historical reading habits and stated preferences among other factors such as book content genre and info. Revenue was lost in the number of users who shortly perused the site before leaving and the ad sense revenue that the company lost out on due to it, the loss in monthly subscription revenue due to users unsubscribing because they couldn't find anything compelling to read, and finally the decrease in revenue generated from the site's currency purchases to unlock chapters. When recommendations didn't hit the mark, users didn't feel anything was interesting to read on the site and therefore stopped buying site currency to unlock chapters, unsubscribed, or left altogether. This process waste was measured in dollars.

The Recommender system was developed and employed to develop a customized and better reading experience for the customer. It utilizes the approach of Hybrid filtering using both collaborative filtering and content-based filtering to make novel predictions using ML algorithms that will earn and sustain reading customers on the site. The website and mobile application collect the demographic data of the customers and add them to information from previous purchases, novel ratings, user behavior, etc. This data will be used in the algorithm to build the final recommender. A database that's a fraction of the size was used to build a prototype to show what the final product will be capable of.

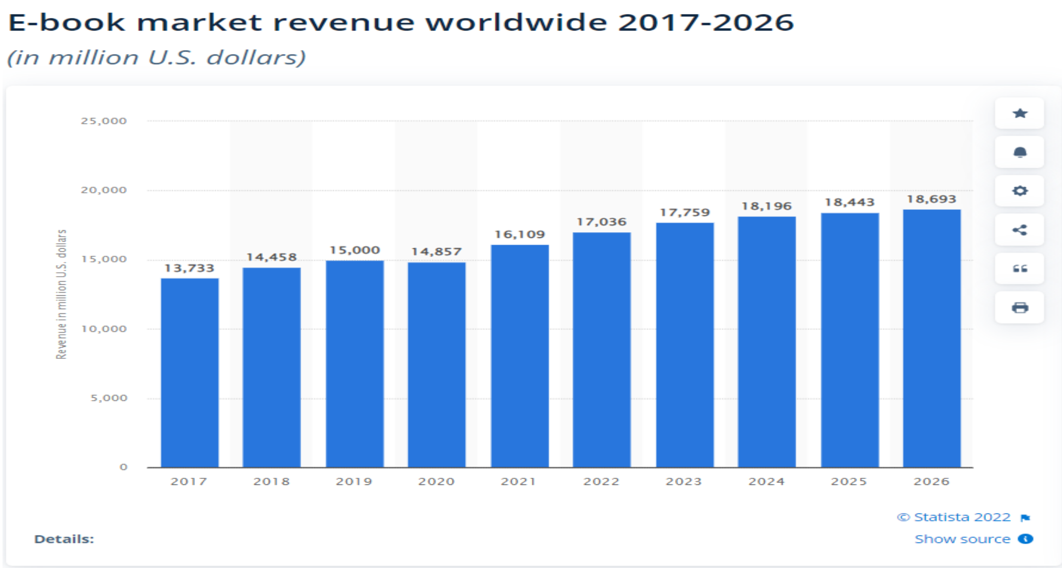
We produced an A.I. novel recommender capable of automatically choosing and displaying the right novels to readers on Webnovel's platform. In doing so, we aimed to keep readers engaged on the platform, boosting revenue for the company and solving the company platform's user retention issue. We developed this A.I. through the use of Hybrid filtering machine learning. We used a combination of the users' data, book data, and user-book interaction data to train our algorithm and make decisions on recommendations. The algorithm's purpose is to interact with and provide recommendations to the user so its ultimate markers of success will be measured by how many click on the recommendations, how many start them, how many pay to read through the paid/sponsored chapters, and how many finish the recommended novel. These can be achieved by setting up tracked hyperlinks on the site. The data was analyzed, cleaned, and used to build a working prototype. We trained the model using a portion of the dataset and tested/validated it using the rest. We've built a working model and further steps would be to augment and implement it into the platform. A custom algorithm platform will be utilized for better flexibility and control while data will be stored in a warehouse under user and book ID

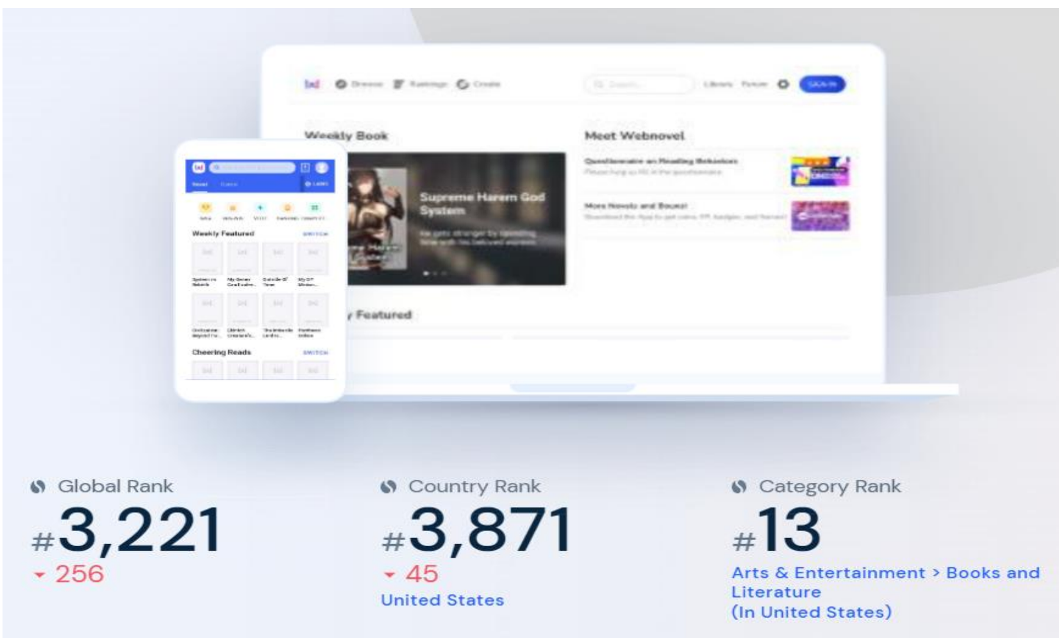
Industry & Client Organization Description

Due to the advent of the internet, there has been a decline in the popularity of brick-and-mortar bookstores as well as sales of print books. With a new pipeline through which to consume literary works, avid readers are turning towards cheaper, more sustainable options when it comes to reading a new book. There has been a boom in the market for alternatives such as audiobooks and especially e-books. While the demand or revenue has yet to overtake the more traditional print, it would be wise to notice the slow industry shift from the traditional form of words printed on paper to words printed on a digital format meant to be read on a screen. Mainstream publications are shifting online and rolling back on printing while consumers are finding it easier, faster, and sometimes cheaper to find their books online rather than going to a store or having them delivered. While it has yet to exceed traditional print, the demand for and use of e-books is growing and that presents an opportunity in the market, regardless of the genre.



While traditional print had a simple business model in selling and renting hardcopy books to make money, the advent of e-books utilizes more diverse revenue-generating avenues. No longer burdened by the cost of printing every unit and dealing with costs such as factory maintenance, transportation, or storage, e-books can be sold at a lesser price for a larger profit margin. The literary work is easily distributed on the internet. People can purchase a digital copy, rent it for some time or even read it for free under certain conditions. The industry is growing since consumers don't have to pay as much to find or read their favorite books.





Our client for this project is Webnovel. They are a growing Singapore-based online platform founded in 2017 that provides access to an extensive library of novels published online ranging across different genres. They reportedly have 200-500 employees. They do not produce or sell any physical copies. They rely on their extensive catalog to attract avid readers to their website where they provide access to their novels on a chapter-by-chapter basis. They utilize multiple avenues to generate profit from site traffic AdSense to monthly subscriptions to the sale of a website currency called spirit stones that can be used to access new novel chapters. The company relies on its user-friendly online library platform to increase and sustain traffic/engagement through which it can generate revenue. The more people click on and stay on their site, the better. Utilizing the internet's global reach, they have a customer base in various countries.

Problem Description

The company generates revenue through the website and mobile app user subscriptions to premium books, purchases of novels through the platform's unique currency, which is run similarly to a point-based system, the monetization of copyrights, ad revenue, and royalty sharing with authors contracted to the platform

To improve the customer experience, and sustain revenue, the platform implements business processes that revolve around increasing user traffic, and subscriptions and the implementation of programs that reward customers for loyalty, and authors for their writing services.

A pivotal business problem that platform faces is accurately building and implementing a system or algorithm that can accurately recommend novels to the platform's users based on their historical reading habits (existing users) and specified novel preferences (new users). By building an accurate algorithm, the platform can potentially retain its current subscribers and increase user traffic to the platform website and mobile application while also prioritizing every user's unique preference and reading habits.

In building and implementing a recommender algorithm/system, a potential waste of resources and computational power could occur by building an algorithm that only targets users that the platform believes are at a higher risk of unsubscribing. By falling into this pitfall, any marketing and promotional value put into these users would result in a loss for the company because certain high-risk users would have unsubscribed regardless of the algorithm's recommendations.

Process waste can be measured through the number of novel subscribers lost over a specified period. It can also be measured by monitoring the overall user traffic to the website and mobile application over specified time periods.

A.I. Solution Description

Recommendation systems or engines have the potential to change the way how sites interact with users and can maximize the ROI(Return on Investment) of the companies based on the information on each customer's preferences and purchases that the system gathers.

Personalized Content: Helps to enhance the in-person experience by generating personalized suggestions for various audiences, much like Netflix does.

Better Product search experience: Assists in classifying products according to their attributes. Example: Season, material, etc.

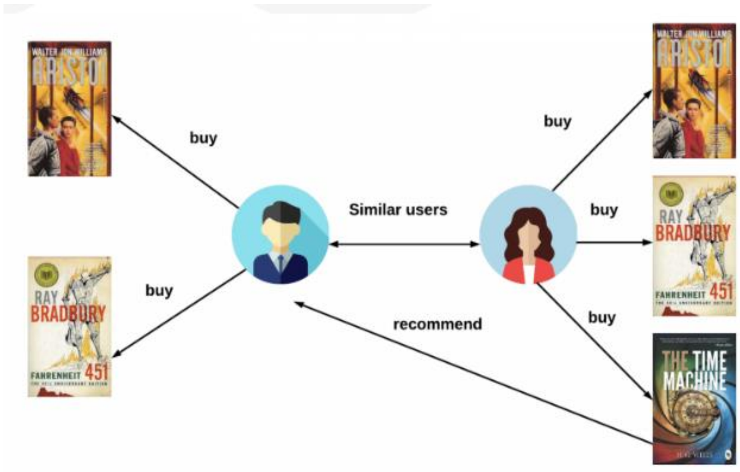
Action

The approach could be brought into action by understanding the science behind each of these approaches to recommendation systems.

- **Collaborative filtering** -This system draws a comparison between one user and another. For example: If person X likes Harry Potter books and person Y like the Harry Potter series and Twilight series of books, then person X might like the Twilight series as well

- **Content-based filtering**- This focuses on the books of themselves and recommends other books that have similar attributes. So, it doesn't rely on other users to make recommendations. E.g.: Pinocchio and Peter Pan

Similar approaches like **Demographic**, **Utility**, **Knowledge based,** and **Hybrid filtering** makes predictions using ML algorithms where AI could role play and change the system into an automated platform that could earn and sustain more customers.



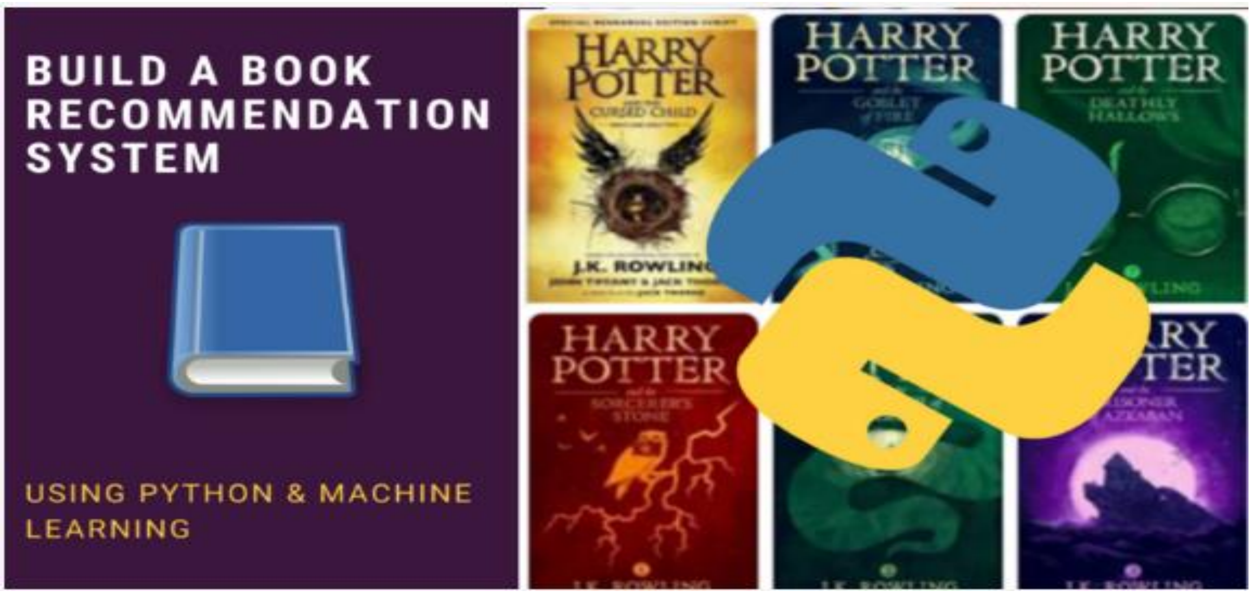
Sensing process/input data

AI-based approaches to E-Book recommendation systems could be known for personalized customer experiences. The web/mobile application will collect and analyze the demographic data of the customers and add it to information from previous purchases, product ratings, and User behavior.

Decision model/agent program

These details that are gathered by the sensing process are then used to research, train and test the model that is built based on the system that is active to predict the minds of the customer on how they will rate sets on a similar book, or how likely a customer would open the book that is additionally recommended, how often they visit and check for new recommendations, how long they stay active on the website, how likely they'll extend their subscription on the e-book website. These are the components of the ML model where some of these features would be fitted as training data and testing data.

A book recommendation engine is essentially a tool that enables marketers to provide pertinent book recommendations to their clients in real-time. Recommendation systems employ algorithms and data analysis techniques as effective data filtering tools to suggest the most pertinent product or things to a specific consumer. Any



recommendation engine's primary objective is to increase demand and involve users in the process. Recommendation engines are primarily a part of an eCommerce personalization strategy that dynamically adds different products to websites, applications, or emails to improve the user experience. These different and cross-channel recommendations are based on a variety of information, including consumer preferences, historical transaction history, qualities, and contextual information.

The company will use the description of the solution as the development process's objective is to help them identify the problems with their business solutions. The degree of client involvement in the process is the primary determinant of whether or not business problem solutions are developed successfully. When creating the needs and specifications for upcoming business items, they will employ the solution. By helping clients achieve their objectives, the generated customized solutions are catered to their company expectations.

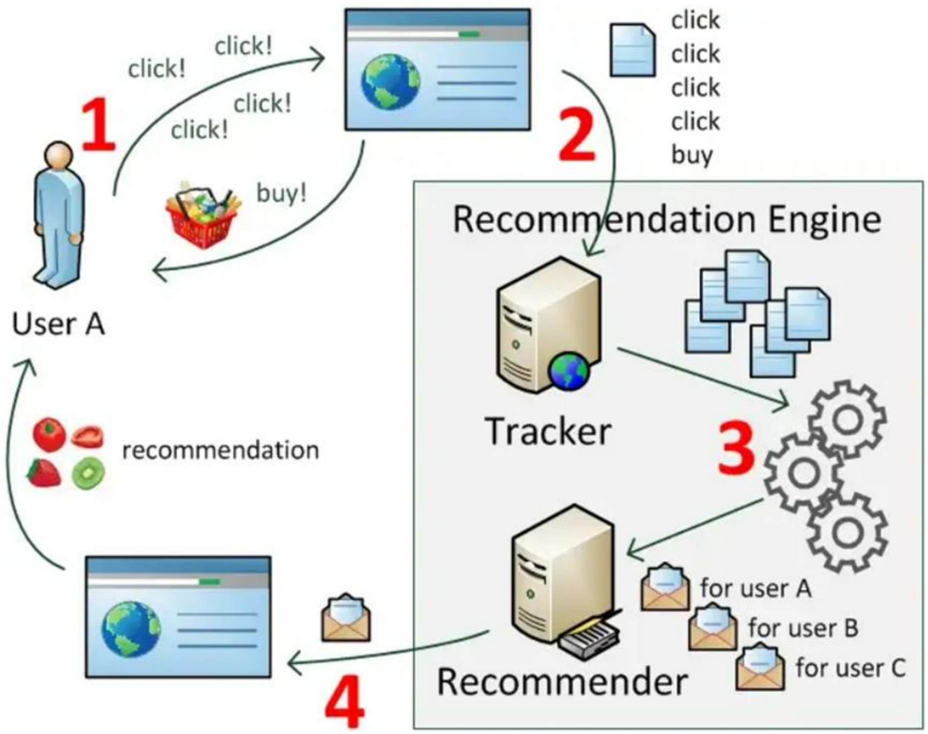
Using input variables as inputs, machine learning algorithms estimate the target function (f) to predict the output variable (Y) (X). Regarding the shape of the function being learned, such as whether it is linear or nonlinear, various representations make varying assumptions. Assumptions regarding the structure and shape of the function and how to most effectively optimize a representation to approximate it vary among machine learning methods. Because we cannot predict which strategy will be most effective at guessing the structure of the underlying function we are trying to approximate, it is crucial to test a variety of algorithms on a machine-learning task. Yes, outputs will be used as inputs in another system to anticipate the outcome accurately. By involving staff members from every department in purpose-driven research, Output is intended to advance your creativity. Administrators can create challenges that focus on high-value issues or opportunities, establishing precise requirements for concept submissions along with supplementary materials like pictures, videos, datasheets, and more. By enabling administrators to control end-user permissions, organize evaluations, and establish precise deadlines for each stage of the creative process, Output also ensures that projects advance effectively. Along the route, the platform gives administrators the ability to control engagement, view, arrange, and export idea submissions, as well as enhance their repetitive procedures. The Challenge participants can collaborate cross-functionally on submissions and get clear feedback on why their ideas were chosen to advance or not thanks to output, which promotes transparency and collaboration. As time goes on, Output assists your company in making better innovations by giving you the freedom to enhance your problem-solving processes by utilizing historical data and analytics.

Learning, reasoning, problem-solving, perception, and language understanding are the five fundamental facets of artificial intelligence. NLP is the general field of employing machines to interpret human language. NLU and NLG are two of the many subfields that make up NLP. NLU focuses on understanding what has been expressed in natural language. The goal of NLG is to simulate human response by creating artificial text that appears to be what a human would have said. By obtaining more values from NLG and NLU, we may use NLP. The Recommendation Engine is then brought into play because NLP may be a part of a recommender system, extending from that. Either for feature extraction or text classification, recommender systems may be built atop an NLP module.

NLP is the best element in our system for content-based filtering, which we will use for the recommendation.

Collaborative filtering, also known as the personality-based approach, and content-based filtering, as well as other systems like knowledge-based systems, are frequently used in recommender systems. In this approach, the things are described using keywords, and a user profile is created to show the kinds of items this user is interested in. These algorithms attempt to suggest products that are comparable to those that a user has previously enjoyed or is now looking at. The user's prior ratings of various potential goods are specifically compared, and the best-matching things are then recommended. Research on information filtering and retrieval is the basis of this strategy.

1. A simulation of user preferences.
2. A record of how the user has interacted with the recommendation engine.



Solution Data Requirements

The behaviors of various types of users, including Anonymous users, new users, and current users, will be used to collect data for the book platform. The recommendation algorithm works to make it as easy as possible for users to locate books or audio podcasts to enjoy whenever they access the book platform. We also predict the user's propensity to search for a specific title in our catalog based on a variety of variables, such as:

- User interactions with our service in the book platform such as viewing history
- Number of hours spent on reading a particular book and the book searches that are done in the engine
- Initial subjects like “Sci-fi”, “comedy”, “feel-good”, “horror” which are chosen by the users, and grouping the users based on the similarity of taste

For example:

To monitor and record the title and genre of books picked or liked by the user who chose the category as “comedy” or “feel-good” which they have in their profile

- Information about the titles, such as genre, categories, author, published year, etc.

The same information that was used to describe books, as well as the user's preferences gleaned from their profile and their actions on the platform. When a user accesses the Webnovel service, our recommendation system analyzes what they have read on the platform and uses that information as input to better personalize the recommendations. We also use the inputs listed below to determine the solution at runtime, and the same data is also used for scoring.

- The time the reader last read
- The technology they possessed for reading books and listening to podcasts
- How many pages or how long the book is read

These many types of information are all used as inputs by our algorithms. A series of procedures called an algorithm are used to solve problems.

Although data quality has been acknowledged as the most crucial element in the larger information systems study, recommender systems have paid little attention to it. Data will be of higher quality if user-specific information is removed. The effect of item content data completeness on recommender system prediction accuracy as moderators for this impact, we specifically look at the rise in completeness per item, per user, and feature. Ten hypotheses are derived from the theoretical model we provide and tested against two real-world datasets, one from a movie review portal and the other from two of the top book review websites.

The overall outcome will demonstrate that greater completeness has a beneficial impact on prediction accuracy. Contrary to what is already known, we can also see from the data that adding features to a recommender system that are more complete but notably different from those already present does not improve prediction accuracy. Ad-hoc cleaning techniques, which remove noisy and unreliable records from the data, address data quality issues in recommendation systems. Frequently, the cleansing parameters are chosen at random without giving the data's properties much thought. In this study, we focus on two major issues with data quality in recommendation systems.

1) Sparsity
2) Redundancy

We remove redundant columns, convert some variables to integers, scale the data using MinMaxScaler, and create models for determining thresholds and sample levels that are dependent on the data. We test these models using a variety of open-source and commercial datasets. Then, we will see that the models have little impact on the accuracy of the recommendation generation while precisely predicting the parameters for data purification. The decision-making method for this recommendation system excludes the consideration of demographic data like age or gender.

Data transformation pipeline requirements

- We required the ability to maintain a snapshot of all the data used to create our models
- The computational solution for the engine is required to be reliable and scalable to handle increasing data. Whether in terms of processing speed, memory size, or both
- The computing solution had to be economical
- For the compute solution to maintain an effective load distribution, a good monitoring tool was required
- It was necessary to have access to the models produced by each run of the pipeline afterward
- To quickly access user recommendations using an API, we wanted to be able to store them somewhere

Acquisition Strategy/Platform Requirements

Under one digital roof, a platform enables the connection of tools, teams, data, and processes. It serves as the foundation of all systems and enables smooth interfaces across all preferred tools.

A programming interface called an artificial intelligence API enables programmers to include AI features in their works. Through APIs, software modules can be combined into a single application or enterprise information system. Intelligent systems that serve us at home, at work, in businesses and in offices can be created. These APIs can be used for a variety of commercial operations, including sharing, spam filtering, facial recognition, and position detection.

A proprietary algorithm is a set of instructions or guidelines that are unique to a corporation because its creator has filed for a trademark or patent on them. These instructions or guidelines are utilized to execute a specific task. One commercial web search engine's ranking algorithm is used as an example; while some information may be easily accessible to the public, the source code is not freely available to protect business interests and avoid misuse.
We may practice training and testing in the Open AI and TensorFlow environments. However, for this use case, the majority of organizations do not need an agent. There are many situations where we want to use RL to further our specific goals, and it should go without saying that neither Open AI nor TensorFlow provides the setting for such tasks. This emphasizes the need

for a personalized environment that is suited to our task. It is possible to choose what to do when the reset environment is engaged, how to go forward, and how many trajectory steps will be taken into account.
Our project made use of a custom environment because it provided a higher level of control and flexibility for the function we wanted it to execute. It would also allow us to tailor our model to existing and active users however there is a thought towards the platform's growth and a shift will be made to an A.I. platform to account for the expected larger customer base on the platform. We know the custom algorithm will only be able to handle the user base to an extent.

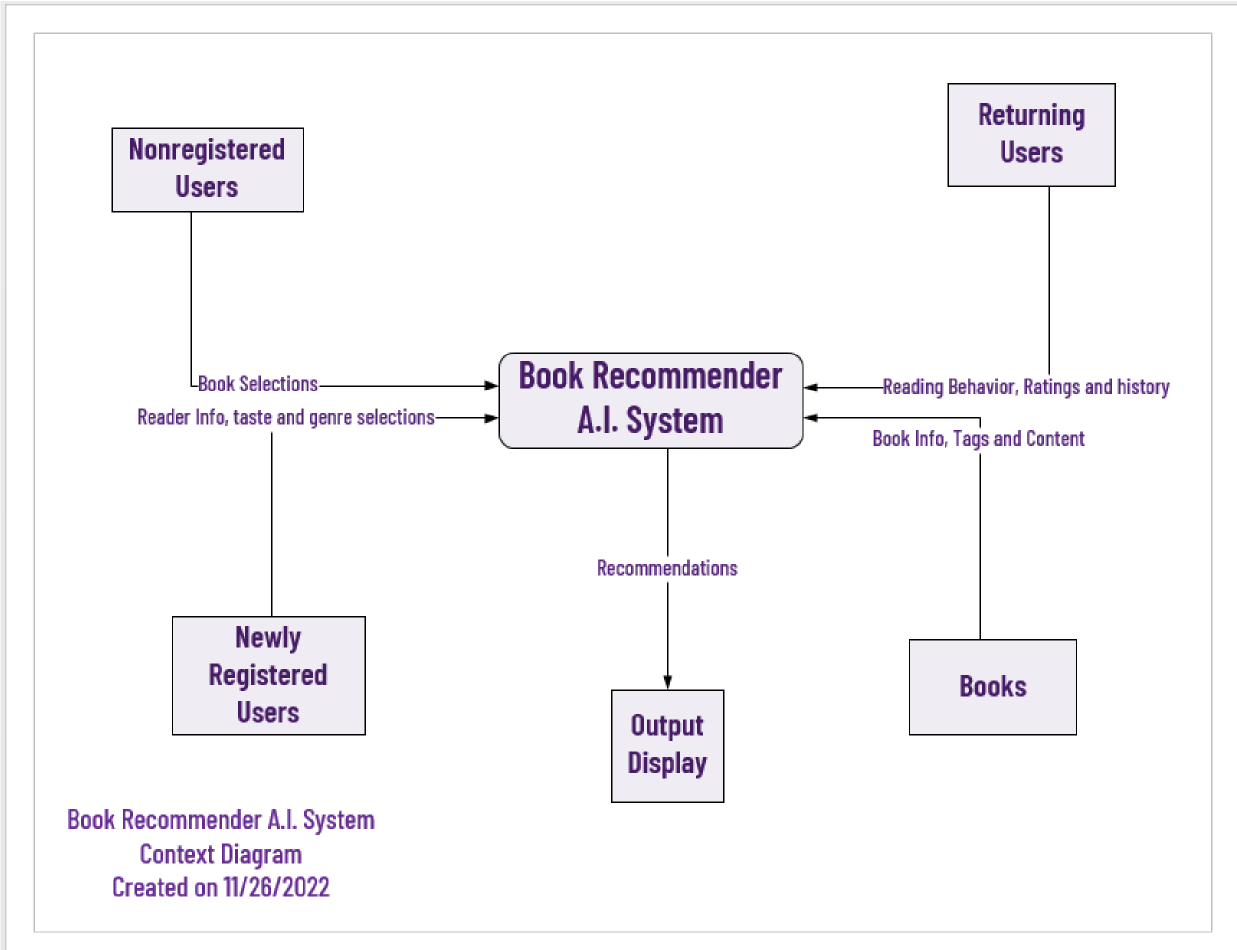
Underlying ML model requirements

Our final solution is a Supervised Machine Learning based Recommender System that intakes all of our inputs (books, users, and ratings) to recommend titles for users based on their unique preferences. These recommendations are the target variable that the model sets out to accurately predict. As earlier mentioned, the final model will be based on a hybrid filtering framework to predict relevant recommendations based on comparisons between similar users (collaborative) and similar titles (content-based).

The model and its performance metric (Mean Average Precision at K score) function effectively by rewarding recommendations from the model that is relevant to a particular user. So naturally, the higher the MAP at K score, the better the model is at recommending relevant items to its users. In mathematical terms, the Precision at K metric can be calculated as the *Number of relevant recommendations / Total number of recommendations*. In this calculation, K represents the number of recommendations the model is tasked with predicting.

As far as the model's framework, it is also important to note that the performance metric also rewards the model on the correct ranking of its recommendation by their importance to the users. The performance metric will never punish the model for adding more recommendations, it will instead reward the model's predictions by its ability to recommend relevant titles to the customer.

Solution architecture



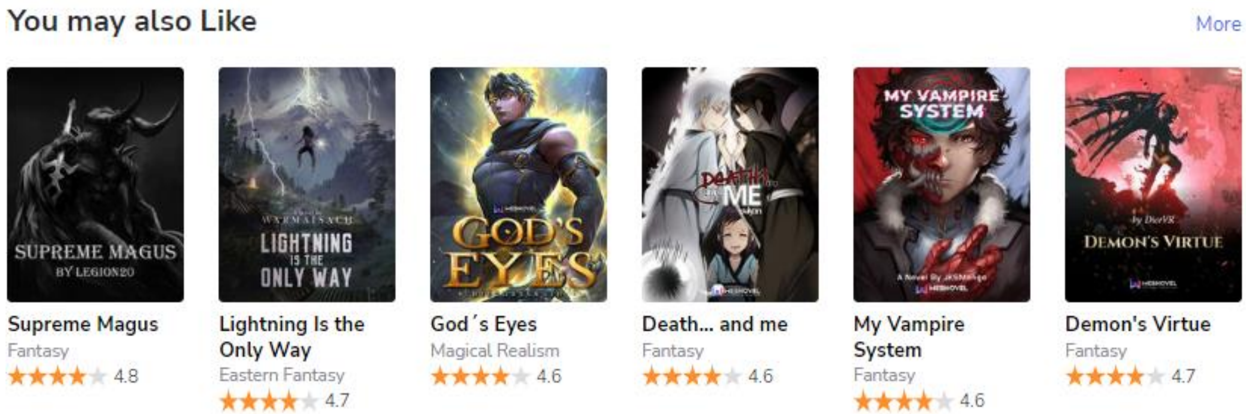
As can be seen above, components of the algorithm can be briefly summarized using the data flow between them. To provide accurate custom recommendations to each reader, the system needs to receive info from the novel library and users of the platform. These users can be broken down into non-Registered users, newly registered users, and existing/returning users. Each user type provides different info as they each use the platform differently and thus would have their contributions weighted differently. Their info along with the book info such as genre, content tags, and author will be funneled into the algorithm to filter out similar novels based on similar content or readers.

The algorithm gets triggered whenever the reader opens up a novel's main page or has just finished all the chapters in a book. At that point, the user's unique ID as well as the book's ID are funneled as input into the algorithm. These IDs contain all pertinent info about the user and book. From the user's history, ratings, tastes, and selections to the book's genre, content tags, and author. Of course, depending on the type of user these might be lacking but that's where the hybrid filter comes in to provide a recommendation the reader should enjoy.

Regarding the cleaning up and transformation of the data, a lot of info is collected about the user when they log in and traverse the platform. Extraneous, useless, identifying, and repetitive data(i.e., Cover art) will be dropped from the recommendation algorithm process while some variables will be converted to integers and strings. In building the prototype the data was cleaned and divided into training and test sets to fully build the algorithm. Upon use, the user and book data will be processed accordingly for the filter and used to generate the recommendations. Once the results have been produced, data is stored in a warehouse for the books and users under their assigned ID, this is for only those who have an account. Non-registered users will experience a somewhat randomized selection fully weighted by the book they've chosen or have just finished. Those with an account will have all their history data stored in a warehouse to be called up again for the next recommendation process.

As mentioned earlier the recommendation process itself is automated and is triggered when the reader opens up a book's main page or comes to the end of a novel's last chapter. At that point, the recommender starts up and calls upon all the information that has been collected under the user and book ID so far to produce the recommendations which will be packaged into a visually appealing section that contains tracked hyperlinks for the reader to click on. All collected data is also automatically stored in the data warehouse linked to the pertinent ID. Also, as new books, ratings, and reader behavior is recorded by the platform those too are automatically added to storage to be called upon later when a recommendation is triggered.

The model process ends with the recommendation output. The process may seem instantaneous but it's part of the process that comes with loading the webpage. When the titles of the novel recommendations have been filtered out, the recommender engine process concludes and the results are then funneled into a visually appealing section of the webpage where the reader can see the cover art, the title, genre, rating, and a correlation percentage based on content or readership. Take the example of a fantasy novel titled "Birth of the Demonic Sword" this would be an example of the output display based on the book ID and User ID. Our recommender will also include a correlation matrix below the rating where the percentage of book similarity or the percentage of similar users reading the book can be displayed.



Prototype

The prototype provides insight into an important framework of the would-be final model. This prototype builds a recommender system using the Collaborative filtering component to create a User-Book matrix that handles all user ratings for (recommendations) every single book in the database. The model then uses correlation to determine the most similar titles to any title of choice based on the ratings given to the title by other users.

The prototype also uses data of users with the most read books (weeding out users with little to no book-reading history) to improve the model's accuracy for recommendations, as fewer active users may not have enough ratings or book history to be included in the prototype and final model.

By creating this framework, we have built a clear blueprint of the components and capabilities of the final model incorporating all of the pertinent information needed to build and improve the final model.

```
[ ] #Authors of 5 books we recommend - final product of prototype
    df_author=df[["Book-Title","Book-Author"]]
    df_author.head()

    df1 = df_author.loc[df_author["Book-Title"].isin(rec_book_list)]

    df2=df1.drop_duplicates(subset=["Book-Author","Book-Author"], keep="first")
    df2
```

	Book-Title	Book-Author
39579	Where the Heart Is (Oprah's Book Club (Paperba...	Billie Letts
41270	The Da Vinci Code	Dan Brown
46418	Dreamcatcher	Stephen King
47952	Ender's Game (Ender Wiggins Saga (Paperback))	Orson Scott Card
108824	The Da Vinci Code	DAN BROWN
129724	The Divine Secrets of the Ya-Ya Sisterhood: A ...	Rebecca Wells
398924	Dreamcatcher	Dinah McCall
540083	Dreamcatcher	Audrey Osofsky

Description: Final prototype output with recommendations for the most similar titles to the "The DaVinci Code"

References

- Rocca, B., & Rocca, J. (2019, June 12). *Introduction to recommender systems*. Medium. Retrieved October 21, 2022, from https://towardsdatascience.com/introduction-to-recommender-systems-6c66cf15ada
- Verma, Y. (2021, October 20). *A guide to building hybrid recommendation systems for beginners*. Analytics India Magazine. Retrieved October 26, 2022, from https://analyticsindiamag.com/a-guide-to-building-hybrid-recommendation-systems-for-beginners/
- Google. (n.d.). *Recommendations: What and why? | machine learning | google developers*. Google. Retrieved October 21, 2022, from https://developers.google.com/machine-learning/recommendation/overview
- Kordik, P. (2019, December 15). *Machine learning for Recommender Systems - Part 1 (algorithms, evaluation, and cold start)*. Medium. Retrieved October 22, 2022, from https://medium.com/recombee-blog/machine-learning-for-recommender-systems-part-1-algorithms-evaluation-and-cold-start-6f696683d0ed
- Stefano, A. D. (n.d.). *Machine learning in recommendation systems: An overview*. Machine Learning in Recommendation Systems: An Overview. Retrieved October 22, 2022, from https://www.itransition.com/machine-learning/recommendation-systems
- Seif, G. (2019, September 4). *An easy introduction to machine learning Recommender Systems*. KDnuggets. Retrieved October 22, 2022, from https://www.kdnuggets.com/2019/09/machine-learning-recommender-systems.html

- Shetty, B. (2019, July 24). *An in-depth guide to how Recommender Systems work*. Built-In. Retrieved October 22, 2022, from https://builtin.com/data-science/recommender-systems
- Dwivedi, R. (2020, April 16). *What are recommendation systems in machine learning?* Analytics Steps. Retrieved October 22, 2022, from https://www.analyticssteps.com/blogs/what-are-recommendation-systems-machine-learning
- Borak, K. (2022, July 25). *An in-depth guide to Machine Learning Recommendation Engines*. Stratoflow. Retrieved October 22, 2022, from https://stratoflow.com/machine-learning-recommendation-engine/
- Villani, P. (2021, November 19). *AI-powered recommendations for media, books, blogs: Algolia*. Algolia Blog. Retrieved October 22, 2022, from https://www.algolia.com/blog/ai/ai-powered-recommendations-for-media-books-blogs-articles-publications/
- Agrawal, R. (2022, July 26). *Book recommendation system: Build A book recommendation system*. Analytics Vidhya. Retrieved October 22, 2022, from https://www.analyticsvidhya.com/blog/2021/06/build-book-recommendation-system-unsupervised-learning-project/
- Gray, K., & Farzindar, A. (n.d.). *Recommender Systems in a Nutshell*. KDnuggets. Retrieved October 22, 2022, from https://www.kdnuggets.com/2020/07/recommender-systems-nutshell.html
- Mahowald, M. (n.d.). *Building a recommender system*. KDnuggets. Retrieved October 22, 2022, from https://www.kdnuggets.com/2019/04/building-recommender-system.html
- Korbut, D. (n.d.). *Recommendation system algorithms: An overview*. KDnuggets. Retrieved October 22, 2022, from https://www.kdnuggets.com/2017/08/recommendation-system-algorithms-overview.html
- *Webnovel - Tech Stack, Apps, Patents & Trademarks*. (n.d.). Crunchbase. Retrieved October 23, 2022, from https://www.crunchbase.com/organization/webnovel/technology
- *Webnovel.com Traffic Analytics & Market Share*. Similarweb. (n.d.). Retrieved October 23, 2022, from https://www.similarweb.com/website/webnovel.com/#overview
- Longo, C. (2018, November 22). *Evaluation metrics for Recommender Systems*. Medium. Retrieved December 2, 2022, from https://towardsdatascience.com/evaluation-metrics-for-recommender-systems-df56c6611093
- Sawtelle, S. (2016, October 25). *Mean average precision (MAP) for Recommender Systems*. Evening Session. Retrieved December 4, 2022, from https://sdsawtelle.github.io/blog/output/mean-average-precision-MAP-for-recommender-systems.html#Precision-and-Recall-of-Recommender-Systems
- *Read birth of the demonic sword – eve of chaos*. Webnovel. (n.d.). Retrieved December 4, 2022, from https://www.webnovel.com/book/birth-of-the-demonic-sword_14187175405584205

- Zainurrohman, A. (2021, February 27). *Content-based recommender system using NLP*. Medium. Retrieved December 5, 2022, from https://medium.com/mlearning-ai/content-based-recommender-system-using-nlp-445ebb777c7a

- Chua, R. (2019, June 24). *How to exploit natural language processing (NLP), natural language understanding (NLU) and natural…* Medium. Retrieved December 5, 2022, from https://becominghuman.ai/how-to-exploit-natural-language-processing-nlp-natural-language-understanding-nlu-and-natural-c83fa256b190

- Bansal, S. (2022, July 12). *Components of artificial intelligence - how it works?* Blogs & Updates on Data Science, Business Analytics, AI Machine Learning. Retrieved December 5, 2022, from https://www.analytixlabs.co.in/blog/components-of-artificial-intelligence/
- Bansal, S. (2022, July 12). *Components of artificial intelligence - how it works?* Blogs & Updates on Data Science, Business Analytics, AI Machine Learning. Retrieved December 5, 2022, from https://www.analytixlabs.co.in/blog/components-of-artificial-intelligence/