Informatics Institute of Technology

In Collaboration With

University of Westminster, UK

University of Westminster, Coat of Arms

NFT-RecSys

A Trading Recommendations System for Non-fungible Tokens

A dissertation by

Mr. Dinuka Ravijaya Piyadigama

w1742104 / 2018373

Supervised by

Mr. Guhanathan Poravi

April 2022

Submitted in partial fulfilment of the requirements for the

BSc (Hons) Computer Science degree at the University of Westminster.

DECLARATION

ABSTRACT

Non-fungible Token (NFT)s allow people to trace the origin of digital items and with the help of Blockchain technology. Since the items are unique from each other, as expressed by the name itself, they are *not fungible*. One NFT is expected to be unique from another. Due to several restraints that are presented with the nature of NFTs & the overwhelming amount of data that needs to be analyzed, it is difficult to find NFTs of comparable value that is trending among the community, timely and relevant to each user's identified interests or the NFT that the user currently owns.

Recommendations Systems have been identified to be one of the integral elements of driving sales in e-commerce sites. The utilization of opinion mining data extracted from trends have been attempted to improve the recommendations that can be provided by baseline methods in this research, to address the restraints presented by NFTs.

NFT-RecSys is capable of acting as a decentralized Recommendations System to provide trending recommendations of NFT assets, while preserving user-anonymity. The data extraction methods explored for recommending NFTs, integration of social-trends into recommendations & the aggregation algorithm of recommendations from ensembled models are novel results yielded by this research.

Keywords: Recommendation Systems, Hybrid Recommendation Systems, Machine Learning, Non-fungible Tokens, Data Science, Opinion Mining

Subject Descriptors:

- Information systems → Information retrieval → Retrieval tasks and goals → Recommender systems
- Human-centered computing → Collaborative and social computing → Collaborative and social computing theory, concepts and paradigms → Social recommendation
- Information systems → Information systems applications → Data mining
- Applied computing → Electronic commerce → Online shopping
- Computing methodologies → Machine learning → Machine learning algorithms → Ensemble methods

PUBLICATIONS

ACKNOWLEDGEMENT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

AI	Artificial Intelligence.
API	Application Programming Interface.
DL	Deep learning.
ERC	Ethereum Request for Comments.
GUI	Graphical User Interface.
IDE	Integrated Development Environment.
LSTM	Long short-term memory.
MAE	Mean Absolute Error.
ML	Machine Learning.
MLP	Multilayer Perceptron.
MSE	Mean Squared Error.
NFT	Non-fungible Token.
NLP	Natural Language Processing.
P@K	Precision at K.
RMSE	Root Mean Square Error.
RNN	Recurrent Neural Network.

CHAPTER 1: INTRODUCTION

1.1 Chapter Overview

In this research project, the author tries to identify the required features to be considered for an NFT-trading Recommendations System and introduce a new Ensemble Architecture for Recommendations that can be applied in other related domains as well. The proposed architecture will try to automate several decision-making steps that a user would otherwise need to go through to find the best possible trade.

This chapter defines the problem, the research gap, the research challenge, and the research strategy that the author wishes to follow over the next few months. The necessary proofs of the problem, as well as previous research interests, are also reviewed.

1.2 Problem Domain

1.2.1 Non-fungible Tokens (NFTs)

In recent months, the NFT market has been growing exponentially as it appears to be the most widely accepted business application of Blockchain technology (Dowling, 2021b), since the introduction of crypto. NFTs are provably scarce unique digital assets that can be used to represent ownership (*ERC-721 Non-Fungible Token Standard* 2021). They can be one of a kind rare artworks, collectable trading cards, and other assets with the potential to increase in value due to scarcity (Conti, 2021; Fairfield, 2021). While being digital assets, they also can be used to represent physical assets. A digital certificate of land/ qualification can be identified as a couple of examples. The biggest winners in the NFT space over the last few months have been digital artists who were able to sell art worth over $2.5 Billion (*Off the chain* 2021).

NFTs were introduced by Ethereum (Wood, 2014) as an improvement proposal (*EIP-2309* 2021; *ERC* 2021) in the Ethereum Request for Comments (ERC)-721 standard (*ERC-721 Non-Fungible Token Standard* 2021). This allows anyone to implement a Smart Contract with the ERC-721 standard and let people mint NFTs as well as, keep track of the tokens produced by it. This allows the created tokens to be validated.

Each of these created tokens is unique from the other tokens created by the same Smart Contract, unlike fungible tokens which were introduced with cryptocurrencies and are denoted by the ERC-20 standard (*ERC-20 Token Standard* 2021) on the Ethereum network. One Bitcoin can be swapped to another Bitcoin, but each NFT will be unique. Then, the deployed Smart Contract will be responsible to keep track of the tokens created by it on the network. A Smart

Contract is a program that resides on the Ethereum network with a collection of code & data (*Introduction to smart contracts* 2021).

For each NFT, the contact address & unit256 tokenId are globally unique on any blockchain. This allows Decentralized Applications (DApps) (Frankenfield, 2021; *Decentralized applications (dapps)* 2021) to take the tokenId and present the image/ asset that is identified by the particular NFT.

"To put it in terms of physical art collecting: anyone can buy a Monet print. But only one person can own the original." (Clark, 2021)

While a digital file can be copied regardless of whether it's an NFT or not, what this technology provides is the ownership of the digital asset. If an NFT that contains your certificate/ domain is held under your wallet on the Blockchain, no one else can get it from you unless they have your digital wallet's private key. Similar to a deed. But, anyone can see, validate and admire what you own.

1.2.2 NFT Marketplaces

OpenSea, which was the first NFT marketplace is also considered to be the largest. In the attempt to become the "Amazon of NFTs", OepnSea raised $23 million in a Series A (Hackett, 2021), following a $100 million raise in a Series B round, ended the company in a valuation of $1.5 billion (dfinzer, 2021; Matney, 2021). Open Sea saw nearly $150 million in sales in the month of June. These marketplaces are set to increase access to the digital goods industry (Chevet, 2018).

An NFT purchased on an Ethereum marketplace can be traded on any other Ethereum marketplace for a completely different NFT. Creators don't necessarily need to sell their NFT on a market. They can do the transaction peer-to-peer, completely secured by Blockchain. No one is needed to intermediate and an owner isn't locked onto any platform (*ERC-721 Non-Fungible Token Standard* 2021).

1.2.3 Recommendation Systems

Recommendation Systems have been driving engagement and consumption of content as well as items on almost every corner of the internet over the last decade. These systems help users identify relevant items on an online platform. When users are recommended with relevant items, it enables businesses in growing their revenue. 35% of Amazon's revenue (Naumov et al., 2019) & 60% of watch time on YouTube (*Recommendations* 2021) comes from recommendations.

1.3 Problem Definition

Currently, there is no way of identifying possible tradable NFT assets, unless manually browsing through the internet. Marketplaces allow searching for NFTs by keywords, categories & pricing, but don't provide personalized recommendations of trending items. This applies to someone who wants to purchase an NFT that shows similar characteristics to another NFT that has already been purchased by a previous buyer or oneself. Since there can be only one owner for an NFT at a time, recommendations using standard collaborative filtering is also not entirely ideal. Content-based approaches won't help identify trending items.

To help with the exploration of these digital assets, it's identified that several steps that the user has to follow to identify trending items that are timely, popular among the community and may have an expected value can be automated.

1.3.1 Problem Statement

It is difficult to find NFTs of comparable value that is trending among the community, timely and relevant to the user's identified interest or the NFT that the user currently owns.

1.4 Research Motivation

The problem identified in this proposal applies to both people who have a lot of domain knowledge about NFTs and people who have no idea how valuable items are in relation to their interests. Whoever it is, no solution would mimic the exact thinking pattern of a person who is searching for a suitable NFT.

As mentioned in the work of Cheng and Lin (2020), Recommendation Systems play a significant role in the resolution of the problem of information overload. In order to provide ideal recommendations to a user, it is important to understand the user's thought process as well as other factors that affect a decision to trade.

Since the Recommendations domains are highly important for many business use-cases and the NFT domain is seeing a booming acceptance with a bright future ahead, this work is expected to add value to the progression of advancements & accessibility related to the domains of NFTs, Blockchain & Recommendation Systems.

1.5 Related Work

Table 1.1: Related work in Recommendations Systems

Citation	Technique Used	Improvements	Limitations
(Larry, 2019)	Autoencoder, trained on chronologically sorted movie-viewing data	Outperformed item-to-item collaborative filtering the bestseller list	*Critique: The timeline doesn't consider overlapping of movies at various points in time, which will be necessary for trends.* Tested only on movie recommendations.
(Cheng and Lin, 2020)	A framework that integrates collaborative filtering with opinion mining & sentiment analysis on users' reviews that is used to create preference profiles.	Effective in dealing with insufficient data and is more accurate and efficient than existing traditional methods. The quality of recommendations can be improved regardless of whether the dataset is rich or sparse.	The semantic strategy of opinion extraction is generic. This may not be ideal to identify different aspects in varied genres. Slang, irony or sarcasm isn't considered in the current framework. It's very dependent on text mining of user reviews. *Critique: A person has to have placed reviews on previous movies in order to create a preference profile.*

(Chen and Hendry, 2019)	A deep learning model to process user comments and to generate a possible user rating for user recommendations have been used.	Outperforms baseline models in training loss value, precision, and recall on the Yelp and Amazon data sets. In the Trip-Advisor data set, DBNSA (Deep Belief Network and Sentiment Analysis) has the best MSE training loss value and recall. DBNSA saves more time than the other baseline methods.	At present, the proposed method is not suitable for real-time testing. This method is required to be tested with a fast Deep Learning algorithm. Sarcastic comments have not been considered in user comments.
(Ayushi and Prasad, 2018)	A hybrid approach of combination of content-based recommendation, user-to-user collaborative filtering and personalized recommendation techniques.	Address the limitations of single domain analysis such as data sparsity and cold start problem. Integration of several domains is further capable of generating higher accuracy in suggestions. Twitter sentiment analysis over the recommended entities generated by the model to help the user in decision making by knowing the positive, negative and neutral polarity percentage based on tweets done by people.	*Critique: Sentiment analysis based on Twitter sentiment is calculated and shown after showing recommendations. It's ironic to recommend something and say if it's good/ bad by the system itself. Better if only positive sentiment-based items are recommended.*

(Ferdiansyah et al., 2019)	LSTM (Long short-term memory).	The proposed model with time series techniques can predict the price for the next days with split the data to train and test.	The result is not good enough regarding the RMSE (Root Mean Squared Error). Future work: modified LSTM layers, adding dropout and modified number of epochs, and using different instability data-sets to test how good the prediction results are or *try to use sentiment analysis combined with LSTM method* to see the impact of the uncertainty in value bitcoin.
(*What are you missing? Using basic machine learning to predict and recommend NFTs with OpenSea data - OpenSea Blog* 2020)	Multiple Regression	This considers past purchase patterns, NFTs saved in wallets to predict if another wallet containing a similar combination will be likely to own an NFT from a specific category (eg: Cryptokitties, ENS domains, etc) in the future.	Recommends NFT categories that a user may be interested in. Doesn't recommend specific NFTs. The user needs to either manually input preferences or provide his wallet key that contains all his owned assets. *Critique: This won't consider current trends. It won't consider the recognition of the creators (eg: NFT made by Beeple).*

1.6 Research Gap

Based on previous work done related to Recommendation Systems, the literature doesn't identify integrating all the factors that affect the desirability of owning relevant, timely & trending NFTs (items) to a recommendations model. This project focuses on an Empirical gap in the NFT domain as well as Theoretical and Performance gaps in Recommendations Systems.

Collaborative filtering, which has been a standard baseline technique for Recommendations for over a decade, can't be taken as the only recommendations model because, by the time one NFT is viewed many times by other users, it may already be too late for another user to purchase that item.

1.7 Research Contribution

The author's research contribution can be summarized as follows:

- **Recommendations Systems**: Data Engineering + Data Science [Machine Learning (ML) + Deep learning (DL)] + Ensemble models
- **NFT Trading**: Recommendations + Artificial Intelligence (AI) + Automation + Data Analysis

1.7.1 Technological Contribution

A Hybrid Recommendations technique that attempts to use public trends in a way that hasn't been attempted in previous research will be explored in order to facilitate the recommendation of relevant, trending and timely items. Automation of several decision-making steps that a user would otherwise need to go through to find the best possible trade will be integrated into the Recommendations Architecture. It is hypothesized that this novel recommendations architecture will be able to be applied to other items as well to give enhanced recommendations based on trends.

1.7.2 Domain Contribution

The information in an NFT that has an effect on a user's desire to be owned will be identified, when attempting to provide suitable recommendations. Looking at the success of Recommendation Systems across multiple systems for over a decade, it is understood that a Recommendation System would help users identify NFTs that they would be interested in trading. This will in return help in increasing sales on NFT Marketplaces and wider adoption of the technology.

NFTs are a result of the advancement of the application of techniques related to Blockchain, while Recommendation Systems are a result of Data Science advancements over the last few decades. Both the domains considered in this research can be identified to be originated from the field of Computer Science.

1.8 Research Challenge

NFTs is a new domain, which has very less research done related to preferences and factors considered when purchasing NFTs. Therefore, it is first important to identify the data points (features) & external factors that affect the value/ desirability of owning NFTs to suggest trading recommendations of NFTs to a user.

"Crypto has a founding tradition of emphasizing freedom and privacy. Maybe because of this prevailing cultural trend, the NFT space does not have many recommender systems." (*What are you missing? Using basic machine learning to predict and recommend NFTs with OpenSea data - OpenSea Blog* 2020)

NFTs are identified to be more challenging to be recommended to users using traditional recommendation methods due to the uniqueness of each item together with the traditions brought forward with the crypto community. Similar to cryptocurrencies, it has been identified that NFTs too have an impact on the general public opinion & trends (Dowling, 2021a).

Currently, available Recommendation Systems haven't had the necessity to consider trends as much as with related to the desirability of owning NFTs. Furthermore, scarcity of items opens another challenge of the inability to keep recommending items that are not available for sale or have already been purchased by an interested buyer. But that alone can't be considered due to the time-tested & proven baseline recommendation techniques being highly effective in multiple domains. Using the identified factors to be considered, a suitable recommendations architecture needs to be implemented.

1.9 Research Questions

RQ1: What are the features of NFTs & external factors that affect the desirability of owning NFTs?

RQ2: How can a system predict the most relevant, trending, timely & worthy NFTs for trading purposes?

RQ3: What are the recent advancement in recommendation models & architectures that can be taken into consideration when building a hybrid Recommendation Architecture, using ensemble techniques?

1.10 Research Aim

The aim of this research is to design, develop & evaluate a novel Recommendation Architecture that will provide relevant, trending, timely, and worthy NFTs for trading purposes by automating some of the decision making steps that the user would otherwise have to do manually.

To elaborate on the aim, this research project will produce a system & architecture that can be used to recommend trending items with respect to a chosen item in a specific data set. The focus will be laid on the recommendation of NFTs. In order to achieve this several public channels of trends will be required to be streamed into the recommendations architecture together with the automation of several decision-making steps that a user that is interested in purchasing NFTs would have to manually go through, in order to make the best possible trade. The use of Data Mining techniques, Natural Language Processing (NLP) techniques, Data Analysis, hybrid, content-based, collaborative filtering & Deep Learning methods will be researched to make the best possible recommendations.

The required knowledge will be studied and researched, components will be developed and the performance will be evaluated in order to validate or invalidate the chosen hypothesis. The system will be able to run in a local browser for personal use or in a hosted server for public use. The data science models & their code will be available for further research and use in a public repository that is easy to get up and running with ease. A review paper will be published with knowledge gathered from the survey of Literature. A research paper will be published on the outcome of the findings in the research project.

1.11 Research Objectives

The Aims and Research Questions mentioned above are expected to be achieved and answered with the completion of the following Research Objectives. These objectives are milestones that will be expected to be met in order for the research to be completed successfully.

Table 1.2: Research Objectives

Objective	Description	Learning Outcomes	RQ
Literature Survey	Read previous work to collate relevant information on related work and critically evaluate them. • **RO1:** Conduct a preliminary study on existing Recommendations Systems & Architectures. • **RO2:** Analyze the perception of Recommendation techniques. • **RO3:** Conduct a preliminary study on NFTs. • **RO4:** Analyze user desires and factors that affect the likability of owning NFTs.	LO4, LO2, LO5	RQ1 RQ3
Requirement Analysis	Specifying the requirements of the project using appropriate techniques and tools in order to meet the expected research gaps & challenges to be addressed based on previous related research and any domain-specific sources of knowledge. • **RO5:** Gather information about requirements related to desirability of owning NFTs & crypto-related assets. • **RO6:** Gather the requirements of a Recommendations System and understand end-user expectations. • **RO7:** Get insights & opinions from technology & domain experts to build a suitable system.	LO1, LO2, LO5, LO7	RQ1 RQ2 RQ3

Design	Designing architecture and a system that is capable of solving the identified problems with recommended techniques. • **RO8:** Design a price prediction system to identify the possible increase/ decrease in value of the NFTs. • **RO9:** Design an automated flow to match NFTs with global social trends data. • **RO10:** Design a data-preprocessing pipeline to add Smart Contract data related to NFTs in the system. • **RO11:** Design a DL or ML Recommendations model that is capable of appropriately utilizing feature-enhanced data to produce recommendations.	LO1	RQ2 RQ3
Development	Implementing a system that is capable of addressing the gaps that were aimed to be solved. • **RO12:** Develop a Recommendations System that can produce relevant, timely & trending NFTs (items). • **RO13:** Integrate automation steps in the prototype to enhance features of NFT records and use them to recommend suitable NFTs. • **RO14:** Develop an algorithm that can utilize factors that are considered to affect the desirability of owning an NFT by a person.	LO1, LO5, LO6	RQ1 RQ2 RQ3
Testing and Evaluation	Testing the created system & Data science models with appropriate data and evaluating them with baseline techniques identified in the literature. • **RO15:** Create a test plan and perform unit, integration and functional testing. • **RO16:** Evaluate the novel model by bench-marking with Precision at K (P@K) score, compared against baseline models.	LO4	RQ1 RQ2 RQ3

Documenting the progress of the research	Documenting and notifying the continuous progress of the research project and any faced obstacles.	LO8, LO6	RQ1 RQ2
Publish Findings	Produce well-structured documentation/ reports/ papers that critically evaluate the research. • **RO17:** Publishing a review paper on related work. • **RO18:** Publishing evaluation & testing results identified from the research. • **RO19:** Making the code or models created in the research process available for future advancements in research. • **RO20:** Making any modified data-sets or re-creation strategies available to the public, to train & test models related to similar use cases of utilized data.	LO4, LO8	RQ1 RQ2 RQ3

1.12 Project Scope

The scope is defined as follows based on the project objectives and a review of existing products with consideration to the granted time period for this research project.

1.12.1 In-scope

The following is a list of the project's scope:

- A system that is capable of recommending NFTs to users based on a specific chosen NFT.
- Creation of a Recommendations System that integrates public trends on social media.
- Creation of a Recommendations System that is capable of providing better rending recommendations compared to baseline techniques.
- Testing the requirement of integrating public trends into a Recommendations architecture with the use of Content-based filtering, collaborative filtering & Deep Learning techniques.
- Graphical User Interface (GUI) that allows a user to provide the tokenId of a chosen NFT by the user & to view the results given by the Recommendations System.
- Automation techniques with related to Smart Contracts will be directly applicable only to selected Blockchains.

1.12.2 Out-scope

The following are the parts that will not be covered by the project:

- Recommending items that haven't been seen previously by the system.
- Creating a Recommendations System that utilizes less computational power & resources compared to baseline techniques.
- GUI with options to tune the Recommendations System.
- All automation techniques to cover every available Blockchain.

1.12.3 Prototype Diagram

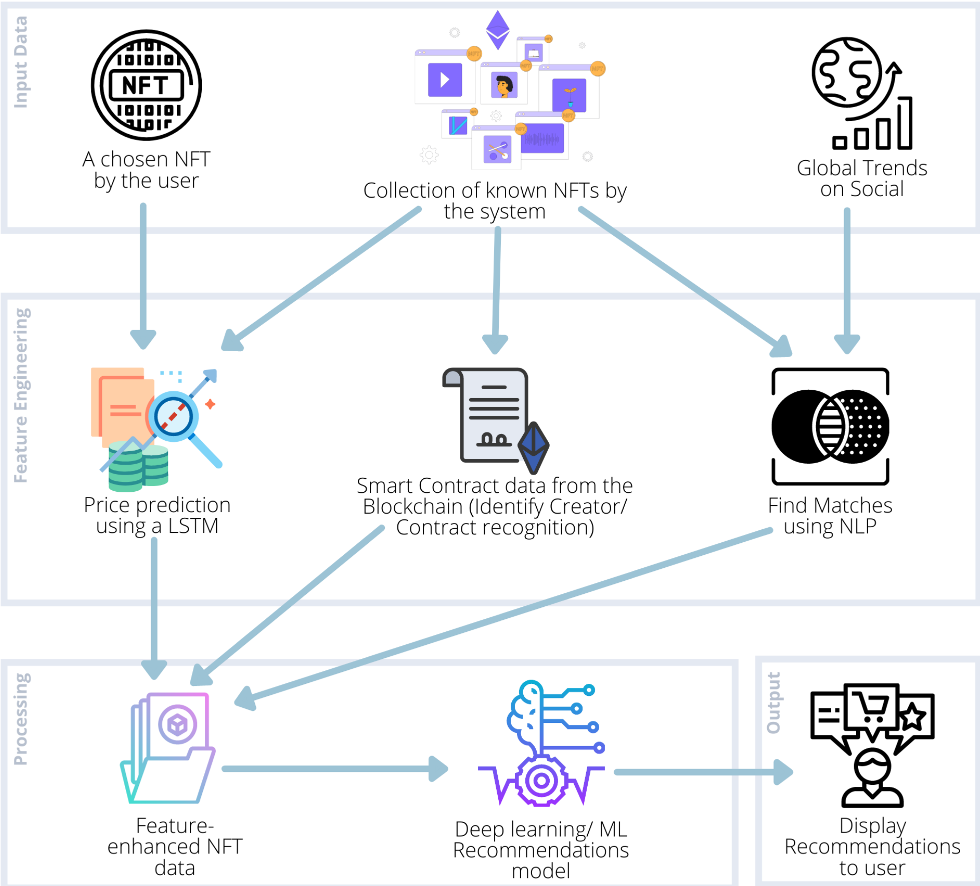


Figure 1.1: Prototype Feature Diagram *(self-composed)*

1.13 Chapter Summary

This chapter presented the problem with necessary proofs and domain description, the research gap, the research challenge, and the research strategy that is expected to be addressed by the author in the research project presented by this document. The research objectives were mapped to the learning outcomes of the project module in the BSc(Hons) Computer Science undergraduate program of the University of Westminster.

CHAPTER 2: LITERATURE REVIEW

2.1 Chapter Overview

As mentioned in the introduction chapter, NFTs have been a very popular application of Blockchain in the recent months. In this chapter, the author critiques on related work with respect to the application of Recommendation Systems while further exploring what, why & how NFTs have been making the headlines and pulling in investors from around the globe. Furthermore, the author has brought-forward possible improvements that may open up possibilities of providing expected recommendations in the NFT-space.

2.2 Concept Map

After conducting a literature survey across a wider-scope, the scope to be covered in this literature review was broken down in a concept graph. The concept graph was created to ensure that all required literuture to be covered would be identified under the areas of problem domain, existing work, technologies, evaluation approaches as well as limitations in each of these sections. The graph can be found in **Appendix A - Concept Map**.

2.3 Problem Domain

Blockchain has been one of the highest sought after fields in the current day and age. NFTs have made the biggest buzz after cryptocurrencies out of the applications of Blockchain technology. With more and more people expected to enter connected digital environments such as the metaverse (Casey Newton, 2021), it is clear that NFTs will play a huge role in tomorrow's internet (Peter Allen Clark, 2021) due to it's ability to make digital items have scarcity, uniqueness, and proof of ownership, similar to physical items (*Non-fungible tokens (NFT)* 2021).

2.3.1 ERC Standards

There're many ERC standards that have been brought forward by the Etheruem (Wood, 2014) development community that are meant to help maintaining standard in smart contracts that are created on the Blockchain with the desired functionalities.

The ERC-721 standard, which is the first standard that introduced NFTs; implements functionalities to transfer tokens from Blockchain accounts, to get the current token balance of an account, to get the owner of a specific token, the total supply of tokens available on the network, etc. Apart from the item itself, the creator can include metadata such as their signature in the NFT. What began on the Ethereum Blockchain with the ERC-721 standard has since been adopted by other Blockchains.

Some of the notable ERC standards that can be identified related to the domain of this research can be compared as below.

Table 2.1: Comparison of ERC standards

Standard	ERC-721	ERC-777	ERC-1155	ERC-20
Name	Non-fungible tokens	Non-fungible tokens (Dafflon, Jordi Baylina, and Thomas Shababi, 2017)	Semi-fungible tokens	Fungible tokens
Description	Each token is completely unique	A richer standard for fungible tokens, enabling new use cases and building on past learnings. Backwards compatible with ERC20.	Tokens begin trading as fungible tokens, then may end up being non-fungible in the long run	All coins of one kind are equivalent and hold the same value
Examples	CryptoKitties (CryptoKitties, 2021)		Concert tickets, gift vouchers, coupons	Cryptocurrencies - Bitcoin, ETH

This research focuses on the ERC-721 and ERC-1155 (Prathap, 2021) standards.

2.3.2 Benefits of NFTs for creators, collectors & buyers

NFTs have a feature to allow a creator to make a certain percentage as royalty whenever the NFT is transferred to a new buyer. Since the items can be verified on the Blockchain, it also ensures that the original creator of the NFT can be tracked down and given due credit, any date in the future, no matter how many wallets it gets passed through (Chevet, 2018). Apart from the fact that a buyer can claim the right of ownership of the original item, they also get to financially support the creator. Ultimately, NFTs may gain value over time due to their scarcity. This gives collectors an additional advantage of being able to sell it for a higher price later on.

Creators of NFTs can also create "shares" for their NFT. This allows investors and fans to own a portion of an NFT without having to purchase the entire thing (*ERC-721 Non-Fungible Token Standard* 2021).

2.3.3 Recent news trends & sales related to NFTs

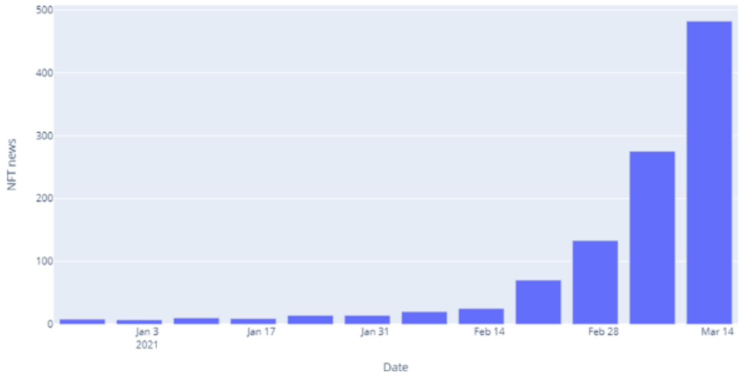


Figure 2.1: News trends in 2021 related to NFTs (Dowling, 2021a)

The above figure shows the increase in news trends related to NFTs since the start of 2021. It has been exponentially increasing and hitting headlines around the globe on a daily basis.

There is almost no brand in the world right now that hasn't either introduced NFTs into their marketing efforts or are working on doing so. *Nike's CryptoKicks* (Beedham, 2019) is one such example.

Two factors can be depicted by this. One; is that NFTs are gaining more and more public attraction and acceptance. The second is that since there's a huge buzz among the public on social media and numerous web-sites, it makes sense to consider the opinions that are shared online by them.

2.3.4 Value-driving factors in NFTs

When considering ownership desire of NFTs, it is understood that the increase in price of an NFT has the possibility of being a factor to be considered when making a purchase.

> *"The value of an NFT is entirely determined by what someone else is willing to pay for it."*
>
> (Conti, 2021)

The value of an NFT has been identified to be heavily reliant on the public's acceptance of the item. Demand is expected to drive price rather than technical, or economic indicators which are the usual factors that affect stock prices and investor demand.

> *"Ultimately owning the real thing is as valuable as the market makes it. The more a piece of content is screen-grabbed, shared, and generally used the more value it gains. Owning the verifiable real thing will always have more value than not."*
>
> (*ERC-721 Non-Fungible Token Standard* 2021)

In addition to gaining value, due to the "non-fungible" nature of the item, it cannot be replicated. Similar to a Mona Lisa painting, popularity helps improve the value of the original and only the original is identified as the truly original painting with immense value, even though anyone can Google and get a copy of the painting.

2.3.5 NFT Market places & what they offer

The money pumped into NFTs & the most popular NFT market, ***OpenSea*** has exponentially increased in 2021 (Matney, 2021). Similar to OpenSea, there're many other NFT market places such as *Foundation, Rarible, Nifty Gateway, Litemint etc*. Some of them built on the Ethereum Blockchain, while some others built on Blockchains such as *Solana (community, 2021; Staff, 2021), Stellar (Fred Rezeau et al., 2021), etc*.

2.3.6 Data mining NFTs

One recent study done on data mining and visualizing has made use of the OpenSea Assets & Events APIs using Python & Pandas to collect, visualize & analyse NFT data on Meebits (Larva Labs, 2021) NFT sales (Adil Moujahid, 2021). The author of this thesis expects to expand on analyzing features beyond those that have been extracted in the data mining and Analysis done on Meebits NFT sales.

2.3.7 Blockchain & AI

AI & Blockchain are bound to be extremely important technologies for businesses moving forward. There're already many applications that bring these two technologies together (Gwyneth Iredale, 2021).

The very first study done examining the pricing of NFTs suggests that *"prospects for future studies are potentially limitless, as at the beginning of any new market"* (Dowling, 2021a). As a future study, the author has suggested identifying if there's a fundamental model that drives the price determination in NFTs. Since NFTs are originating from Blockchain; which is a technology that comes from the field of Computer Science, it's important to understand the factors that affect the pricing and market created by them.

Why create a Recommendations System for NFTs?

In 2018 it was estimated that 35% of Amazon's revenue Naumov et al., 2019 is driven by Recommendation Systems. 75% of Netlfix viewer activity Vanderbilt, 2021 was also said to come from recommendations back in 2013. Therefore, it is clear that the use of a recommendation system that is catered toward the needs of potential NFT owners will help increase sales of NFTs, driving forward the adoption of this technology.

2.3.8 Proposed architecture of a Recommendations System for NFTs

By the requirements identified to purchase & own NFTs, the author has proposes the following architecture to be followed in order to achieve the aim stated to be achieved in this research.

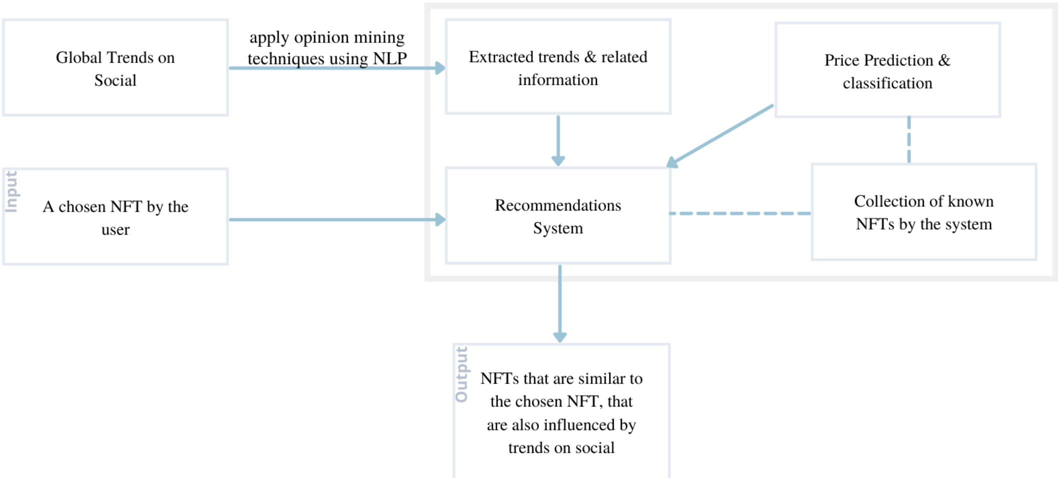


Figure 2.2: Proposed architecture of a Recommendations System for NFTs *(self-composed)*

As shown in figure 2.2, the proposed architecture is expected to make use of global trends extracted using social Application Programming Interface (API)s. These can be from Twitter, Reddit, Google Trends or any other source that the user wishes to use. Once extracting relevant information using NLP, the Recommendation System can then use this information to predict items that are relevant to the chosen item by the user and also those that have a possibility of getting influenced by trends on social.

2.4 Existing Work

2.4.1 NFT Recommendations Systems

There is only one study previously done with related to recommending NFTs and that study also comes in the form of a blog article on *OpenSea* (*What are you missing? Using basic machine* *learning to predict and recommend NFTs with OpenSea data - OpenSea Blog* 2020). The article considers the use of a basic ML technique called **Multiple Regression** with data gathered from OpenSea.

This takes into account previous purchase patterns and NFTs held in wallets to predict whether another wallet carrying a similar combination is likely to own an NFT from a certain category in the future. The categories considered here are mostly collections created by specific well-known creators. Cryptokitties and ENS domains are a couple of examples for collections that have been taken into consideration.

As a final recommendation, this system is capable of presenting NFT categories. Since users can't purchase an entire category, they will have to go back to the process of picking which NFT to purchase in the recommended collection.

This doesn't take into consideration of current global trends and it will not take into account the creators' recognition. An NFT minted by Beeple or a major league like NBA are bound to capture more attention of buyers compared to an NFT minted by a person who hasn't gained any reputation in this space. The major concern with regarding this system is that the user must either enter his preferences manually or provide his wallet key, which holds all of his owned assets, in order to get a recommendation from the system. Although, getting a users' public key can by no means cause any threat of loosing the NFTs, it can be lead to lack of privacy, which is a tradition that the people into crypto-related assets have a tendancy to be concerned about.

> *"Crypto has a founding tradition of emphasizing freedom and privacy. Maybe because of this prevailing cultural trend, the NFT space does not have many recommender systems."*
> (*What are you missing? Using basic machine learning to predict and recommend NFTs with OpenSea data - OpenSea Blog* 2020)

As mentioned in the same blog post, this tradition is also been identified as a reason to why we have not yet seen much development related to Recommendation Systems in this space. Another reason could be because of the very recent spark in interest this domain has seen in recent times, as mentioned in the Problem Domain.

2.4.2 Crypto recommendations

Since NFTs have a distant relationship with crypto assets, it is expected to be of help to understand how crypto assets are evaluated when opted for selection to comprehend how NFT assets could be evaluated. A study done related to a modelling framework that exposes this area of research (Bartolucci and Kirilenko, 2020) assumes that two main features, namely security and stability can be used to determine the user-desire to own a specific crypto asset.

Investor's attitudes towards assets' features, information about the adoption trends, and expected future economic benefits of adoption have been simulated in order to predict the features of the assets that will most likely be adopted. The preference of investors are collected from an app, which calculates the overall state of the 'market'. Then, the app recommends to the user which crypto assets proposed by the user would be a sensible investment. Information about the adoption choice of other investors is considered when making this recommendation.

The number of assets, investors and assets' features and investor preferences were fixed within the period of analysis. In a normal use-case scenario, it's highly likely that all these would fluctuate and evolve with the asset's adoption probabilities and expected returns. This revelation clarifies the fact that crypto related assets have a tendency to change with time, social acceptance and trends. Therefore, it is important to consider these factors when building a crypto-related Recommendations System.

2.4.3 Opinion mining & sentiment extraction based Recommendation Systems

"Catching opinions from social media could be a cheap, fast and effective way to collect feedbacks from users"

(Zhang, Xu, and Jiang, 2018)

When the above fact is looked at in a more generalized form, it is clear that exploiting user trends that build-up of opinions from social media can lead to better quality recommendations, while (Hu et al., 2020) expresses how sentiment analysis of user reviews can be used to point in the direction of personalized recommendations.

A **hybrid Recommendations System** (Cheng and Lin, 2020) which utilizes **opinion & sentiment extraction techniques from user reviews** to create preference profiles for movie recommendations, to enhance the quality of recommendations regardless of the rich or sparse nature of the dataset has been identified as one of the recent researches done towards pushing the limits of baseline recommendation models. The framework that has been designed here uses Collaborative Filtering as the base Recommendations model. The contribution of this research is applicable to the feature engineering stage of the system.

Sentiment analysis is applied on user-reviews to detect user-opinions about movies that were watched and reviewed by users. This data is used to create a user's preference profile, similar to what's created in Content-based filtering. The user's sentiment is identified as a step beyond traditional preference ratings.

Due to its capability of dealing with insufficient data, the framework is able to produce recommendations that are more accurate and efficient than existing baseline methods. This proves that using public opinion in the feature engineering stage can enhance the quality of recommendations.

Due to the fact that the semantic strategy of opinion extraction being generic, it is understood that it may not be ideal to identify different aspects in varied genres. Examples mentioned are, quality of sound may be of greater interest in action movies, while the story-line in dramas. Slang, irony & sarcasm haven't been taken into consideration when extracting user opinion. A major limitation identified in most systems that rely on similar opinion mining systems is that they are very dependant on the text mining technique used. Another identified drawback in this research by the author is that, to establish a preference profile, a person must have posted reviews on previous movies. If not, those users won't be able to get recommendations. This can be identified as a concern in systems that are dealing with user's who care about their privacy.

A **Deep Belief Network and Sentiment Analysis (DBNSA)** has been introduced to achieve data learning for recommendations (Chen and Hendry, 2019) to enhance recommendations produced by baseline-recommendation techniques. This deep learning model processes user comments to generate a possible user rating for user recommendations.

"Users usually transmit their decisions together with emotions."

(ibid.)

This research paper emphasizes the necessity of using user comments for recommendation systems since these comments contain a variety of emotional information that can influence the correctness and precision of recommendations.

Once applying sentiment analysis, a feature vector is created for the input nodes. A noise reduction procedure has been integrated into the system that deletes short comments, comments with no expression and false rating comments. This is used to improve the classification of user ratings. Finally, the DBNSA accomplishes data learning for the recommendations.

The paper published claims to outperform baseline models in training loss, precision and recall when tested on Yelp & Amazon datasets. When tested on the Trip-Advisor dataset, DBNSA had the best Mean Squared Error (MSE) training loss value & recall. The research also mentions that DBNSA saves more time, while producing results with better accuracy compared to other baseline models.

The main drawback that this paper points out is that the proposed system is not suitable & ready for real-time testing. The authors of the paper have also shown interest in testing the proposed method with a faster Deep Learning algorithm. Similar to the previously mentioned system, sarcastic user-comments have not been taken into consideration here as well. Out of the two recommendations models that were tested, *libSVM* was identified to have higher accuracy value, Mean Absolute Error (MAE) and F-score, while the Multilayer Perceptron (MLP) had the highest precision value.

Since user relationships and timeline comments also affect the user's decision making, these can be used to find information from relatable timelines to solve the cold start problem.

A **hybrid approach that combines techniques from content-based filtering, user-to-user collaborative filtering and personalize recommendations** (Ayushi and Prasad, 2018) has been introduced to address the limitation of single domain analysis. Data sparsity and cold start problem have been pointed out as the addressed limitations. Movie domain knowledge has been used to generate recommendations for books & music. After considering an array of supervised learning algorithms, the authors came to a conclusion that the Decision Tree classifier was found to give the highest accuracy.

The use of data from multiple domains allows the system to generate higher accuracy in suggestions. Twitter sentiment has been used to present the user with an analysis of the recommendations produced, to help users in their decision making process.

The drawback identified in the Recommendations System developed here is that Twitter sentiment is analysed, calculated and displayed only after showing the user recommendations.The author's suggestion is that only the items with positive sentiment could've been presented, at least results could've been bias towards positive sentiment.

2.4.4 Price prediction using social-media trends

As mentioned under the Problem Domain section of this literature review, it is understood that NFTs have very little spill-over with other Crypto assets. However, knowing Crypto price prediction models is important since Wavelet coherence analysis indicates a co-movement between these two markets (Dowling, 2021b). These models can be used separately on each NFT asset to anticipate the pricing with related to time, sales & bids. The author finds this research to be related to address the research gap in this thesis since an appropriate price prediction could be used to enhance NFT recommendations to users.

Past research suggests **a model which employs time series techniques, can predict the price for the next few days** by splitting the data into train and test runs (Ferdiansyah et al., 2019).

In terms of Root Mean Square Error (RMSE), the result is insufficient. The authors of this research have shown interest in testing out this method with modified Long short-term memory (LSTM) layers by adding dropout and modifying the number of epochs. Using different instability data-sets can also be tried out to test how good the prediction results could get. Furthermore, sentiment analysis is also proposed as future work to be combined with the LSTM method. This could be used to identify how public sentiment causes the value of crypto to adjust, with related to past price-fluctuations.

2.5 Technological Review

Recommendations Systems allow users to identify trending items among a community, while being timely and relevant to the user's expectations. When the purpose of various Recommendation Systems differ, the required type of recommendations also differ from each use case. While one Recommendation System may focus on recommending popular items, another may focus on recommending items that are comparable to the user's interests. Content based filtering, user-to-user & item-to-item Collaborative filtering and more recently; Deep Learning methods have been brought forward by the researches to achieve better quality recommendations.

Even though each of these methods have proven to perform well, there have been attempts to push the boundaries of their limitations. Following a wide range of methods, researches have tried to expand on the capabilities of standard recommendation systems in order to provide the most effective recommendations to users while being more profitable from a business's perspective. This has been achieved by taking a hybrid approach when building models and architectures for Recommendation Systems.

2.5.1 Machine Learning based recommendation techniques

There are several baseline techniques of Recommendations Systems that have been used by the biggest data-driven companies around the world. Among the many types of recommendation systems, **item-to-item Collaborative filtering** (G. Linden, B. Smith, and York, 2003) has been the most successful technique for an extended period of time (Brent Smith and Greg Linden, 2017), while user-to-user Collaborative filtering and Content based filtering have also had their own upsides. In order to take advantage of the relevant advantages of each method, Hybrid recommendation systems (Geetha et al., 2018) were introduced.

2.5.2 Deep Learning based recommendation techniques

In 2019, **Facebook** open-sourced a new categorical data-driven **Deep learning-based recommendation engine** (Naumov et al., 2019; *We are open-sourcing a state-of-the-art deep learning recommendation model to help AI researchers and the systems and hardware community develop new, more efficient ways to work with categorical data.* 2019). This recommendation model was developed from the two perspectives of recommendation systems and predictive analytics. It made use of embeddings, two MLPs, one sigmoid function (Freudenthaler, Schmidt-Thieme, and Rendle, 2011) and a parallelization scheme to support large-scales of data.

In recent research done by **Amazon** (Larry, 2019) it is understood that when a timeline is considered for recommendations, an ***Autoencoder*** **Deep Learning model** is capable of Recommending the best possible combination of movies to users.

2.5.3 Concerns about progress in Recommendation Systems

In several research & review papers, it has been brought to sight that Deep learning techniques in the area of recommendation systems have failed to live up to the expectations compared to the advancements in Computer Vision, Speech Recognition & Natural Language Processing domains (Choi et al., 2021). The results that have been published presenting advancements in the Recommendation Systems domain using Deep learning techniques have not been very convincing for the majority of use cases. Many standard Machine learning & regression techniques have been able to outperform systems created using Deep learning models in terms of recommendations. As highlighted in past reviews (Dacrema, Cremonesi, and Jannach, 2019) it is understood that Deep learning models have been used as baseline methods for evaluating new Deep learning models. Thus, when looking back at older Machine learning techniques, they haven't been making any improvement in many cases. As a result, many of the work related to Recommendation Systems using Deep learning techniques have been giving poorer recommendations, for higher computational power.

A study conducted in 2019 questioned if we are really making any progress with Deep Learning models in the domain of Recommendations (ibid.). In a more recent study researches tried to understand similarities and advantages of using **MLP** versus **dot product** (Rendle et al., 2020). Similar to many Deep learning approaches, it was understood that MLP weren't necessary unless the dataset was too large or the embedding dimension was very small. A dot product was identified as a better choice since it was efficient to a satisfactory extent.

2.5.4 How to choose the ideal algorithm for a Recommendations System?

A general application of a Recommendation System will come in a business use case, where companies focus on maximizing profits for minimum expenses. In a scenario like that, it would make more sense to choose a cheaper model that gets the job done to a satisfactory level. Dot products offer a significant advantage over MLPs in terms of inference cost due to the availability of efficient maximum inner product search algorithms. Since MLPs are too costly to use in production environments, the better default choice in most cases would be the dot product approach that uses Machine Learning techniques with Matrix Factorization.

$$< x, y >= \sum_{i=1}^{d} x_i y_i \tag{2.1}$$

$$y = \varphi(\sum_{i=1}^{n} w_i x_i + b) = \varphi(w^T x + b) \tag{2.2}$$

where w denotes the vector of weights, x is the vector of inputs, b is the bias and phi is the non-linear activation function.

A variation that combines the MLP with a weighted dot product model, named ***neural matrix factorization (NeuMF)*** is also explored in this research. But, that too is deemed to be outperformed by the dot product method.

One of the major limitations identified related to dot product in this study is that, learning a dot product with high accuracy for a large embedding dimension required a large model capacity. This may also require more computational resources. Therefore, it would be advisable for Data Science engineers to consider both approaches based on the requirements & data of the system that they're planning to work on.

2.5.5 Architectures of Recommendation Systems that integrate opinion mining techniques

There have been many attempts to expand the capabilities of Recommendations by making use of public opinion. Collaborative Filtering was one approach to achieve that. Another identified approach was to make use of user-data on social media. This has been integrated into Machine Learning-based Hybrid Recommendation Architectures in many ways. In the figure 2.3, the author tries to elaborate on the possible technical contribution brought forward in this research.

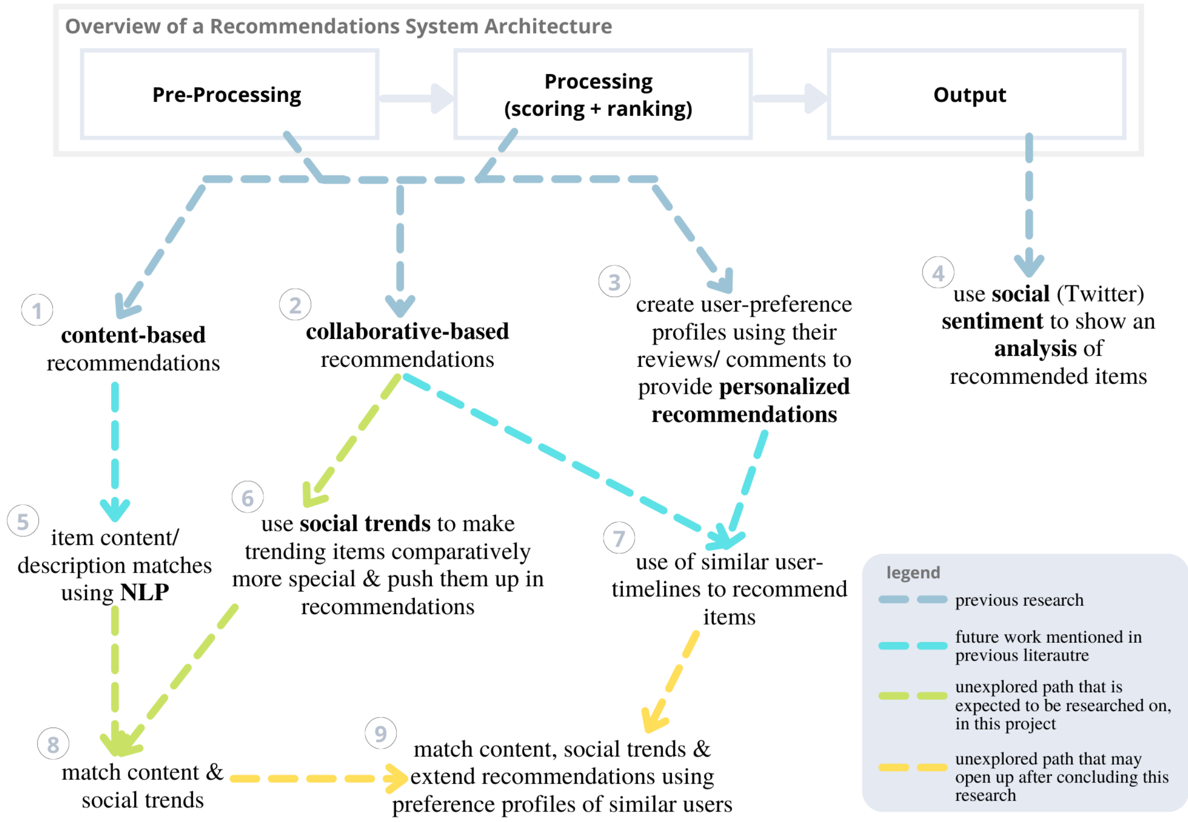


Figure 2.3: Enhancements done to Recommendation Systems using opinion mining techniques *(self-composed)*

The figure 2.3 shows the identified possible points of integration of opinion mining techniques to a Recommendations System. 1, 2 (G. Linden, B. Smith, and York, 2003; Larry, 2019), 3 (Cheng and Lin, 2020) & 4 (Ayushi and Prasad, 2018) techniques have been already applied as identified in past literature, while the 7th technique has been mentioned as a possible future work from the 3rd technique (Chen and Hendry, 2019). Method 5 hasn't been explicitly attempted in recent literature with respect to Recommendation Systems, but the data science models used aren't expected to require a lot of tweaking to achieve it, after the feature engineering step is being taken care of.

Method 6 has not been identified in previous literature and is expected to align better with the desires circulating the NFT market-space. This can be extended to method 8. Finally, if methods 7 & 8 turn out to give promising results, method 9 would be the next step to provide a completely new personalized recommendations architecture that integrates social media trends that are related to the content of the items.

2.5.6 NLP techniques that can be applied to support integration of opinion mining into Recommendation Systems

The main NLP techniques that were identified to be useful to be implemented in a system that requires data-mining & opinion mining techniques are were Sentiment Analysis, Named Entity-Recognition, Tokenization, Stemming & Lammetization; the latter 4 techniques being required for pre-processing scraped data from opinion-mining techniques.

In order to apply these techniques, many past literature (as mentioned in Existing Work), points in the direction of using industrial-grade libraries that utilize **Recurrent Neural Network (RNN) architectures** such as *SpaCy and NLTK*. The most state-of the-art models & techniques that make use of **Transformer architectures** can be found in the *Hugging Face* library (Wolf et al., 2020).

2.5.7 Practices to be followed to optimize the usage of gathered opinions

When considering multiple opinions related to a specific topic/ item, they can be combined into one document and processed rather than processing each opinion one by one (Zhang, Xu, and Jiang, 2018). When doing so, it would be good to have an impact score of each document to make sure that recommendations are biased appropriately towards the opinions of the majority with consideration of the users' opinions.

2.6 Review of Evaluation Approaches

When evaluating Recommendation Systems, we may examine the outcomes produced by the system in two ways. The first way would be identifying if the system is capable of recommending items that a user may use. The second method would be to identify if the system is capable of recommending items that a user will choose/ use.

The first way to evaluating the outcome can be done utilizing current data and pre-identified conditions. For the second approach, the evaluation algorithm would require feedback from the public. This can be done by having open beta testing. It would take more time & effort, but it will be capable of evaluating a system qualitatively on the final goal instead of a possibility.

If we look at evaluating this system from an expected-output performance point of view, $P@K$, also identified as *Top-N strategy* in several literature is the most common method of evaluating a Recommendations System. This measure and the metrics that have been mentioned below can be used to **quantitatively** evaluate Recommendation Systems.

Table 2.2: Evaluation Techniques for Recommendation Systems

Measure	Description	Objective Orientation
MAE	Measures the average absolute deviation between a predicted rating and the user's true rating, overall the known ratings.	Negatively oriented. Lower, the better.
RMSE	A variant of MAE emphasizes large errors by squaring them.	
Precision	The percentage of items in the recommended list that are assessed to be relevant to the user (i.e. it represents the probability that a selected item is relevant).	Positively oriented. Higher, the better.
Recall	The ratio of relevant items presented by the system to the total number of relevant items available in the items in the system.	

MAE & RMSE are used to measure the accuracy of predicted user-ratings (1-5 star ratings) per item, per user. Precision & recall are used to measure if the system successfully predicts which items the user will select or consume (Dayan et al., 2011).

Since the goal of the Recommendations System is to provide the user with multiple options, it is better if the system can produce options across a diverse range. To evaluate the diversity of items across the produce recommendations, *Aggregate diversity* can be measured.

Apart from these metrics, quality-of-service measures such as CPU & Memory usage can be considered for evaluation as well.

In the review questioning the advancements of Recommendation Systems, (Dacrema, Cremonesi, and Jannach, 2019) the author mentions that the lack of used datasets and code-bases hinder the ability to properly benchmark and evaluate new research related to Recommendation Systems. The importance of reproducibility of research related to Recommendations Systems have future been elaborated in reviews that follow (Dacrema, Boglio, et al., 2021; Ferrari Dacrema et al., 2020; Dacrema, Cremonesi, and Jannach, 2020).

2.6.1 Benchmarking

A common test dataset is required in order to consider the results produced by these methods to be valid. Since there's no previous NFT Recommendation System found in research, the author will not be able to conduct a comparative benchmark analysis on the proposed system. Therefore, a **Baseline-Benchmarking** strategy will be followed.

The evaluation benchmark results produced by this system will be made available public together with the used datasets in order to allow future researchers to evaluate new Recommendation Systems in this domain.

2.7 Chapter Summary

This chapter started off by breaking down the problem, technological domains, existing work & evaluation approaches in a concept map. Then these 4 sections were further broken down into sub-topics and reviewed based on work and concepts from past literature. A critical evaluation of all the literature has been done comparing similarities and differences in past work, possible future work mentioned in literature and novel methods that the author of this research suggests as possibilities that haven't been mentioned in previous research.

CHAPTER 3: METHODOLOGIES

3.1 Research Methodology

The quality of any project is governed by three key factors: cost, time, and scope, all of which must be managed efficiently throughout the project's lifetime. As a result, methodologies are required. Saunders Research Onion Model (Saunders, Lewis, and Thornhill, 2003) has been used to deduce the methodologies. The methodologies chosen as appropriate for the project are listed in the table below.

Table 3.1: Research Methodology

Research Philosophy	The philosophy of research influences data collection & data analysis since it is related to the nature of reality being investigated. Positivism, Interpretivism & Constructivism are philosophies that could be used to approach this research. Out of these, **Positivism** was chosen since the research is expected to be replicable with similar **quantifiable** results.
Research Approach	The approach that a researcher may use when conducting the research is the approach. A **Deductive** approach was chosen over an Inductive approach since this is expected to be a **quantitative** research that aims to **test & prove** the **hypothesis** at hand.
Research Strategy	The strategy focuses on the data collection methods that will be used to answer the research questions. **Survey, Archival Research** & **Ethnography** were the strategies chosen to address the research questions. These strategies were chosen as they would compliment each other while providing relevant data that is enough for the research. While Survey seems to be the primary strategy, Archival Research & Ethnography is expected to allow the qualitative aspect expected in the approach taken to the solution, which will finally affect the **quantitative results**, to be addressed.
Research Choice	Choice of the methodology identifies if the research is concerned with the qualitative and quantitative aspects of the research. **Multi-method** was chosen since although **quantitative results** are the primary perspective, it is identified that qualitativeness of the data used by the system to be developed will also be an important consideration that will affect the quantitative results.

Time Horizons	**Longitudinal** was chosen as the time horizon for the research since data will be gathered and used for evaluation and testing over a long period of time.
Techniques and procedures	Data collection and analysis techniques are considered here. Mediums such as online news, statistics & trends from social media, observations, conversations, reports, surveys, documents, secondary tabular data, organizational records will be used.

3.2 Development Methodology

3.2.1 Life cycle model

Agile Software Development Life-cycle was chosen as the research development method since iterative development is needed.

3.2.2 Design Methodology

Object-Oriented Analysis and Design were chosen as the Design Methodology by the author to support an incremental methodology that can be used to extend the system with the ability to reuse system components.

3.2.3 Evaluation Methodology

As identified in recent advancements in literature (Larry, 2019), P@K score has been identified as a suitable method of evaluating a Recommendations System. This is also identified as the Top-N strategy in several past literatures. Therefore, it will be used to compare the novel solution that is to be developed against baseline models.

Benchmarking

Precision, recall, MAE and RMSE will be used to Benchmark the Recommendation System (Dayan et al., 2011), to help evaluate future researches in this domain by conducting comparative benchmarking-analysis.

3.3 Project Management Methodology

Prince2 was chosen as the project management methodology. It allows the author to develop the product in controlled environments in logical compartmentalized units.

3.3.1 Schedule

Gantt Chart

Please refer Appendix B - Gantt Chart.

Deliverables

Table 3.2: Deliverables and dates

Deliverable	Date
Project Proposal Document The initial proposal of the project	4^{th} November 2021
Literature Review Document The Critical review of existing work and solutions	11^{th} December 2021
Software Requirement Specification The document specifying requirements to be satisfied and developed as the final prototype and means of collecting data	15^{th} December 2021
System Design Document The document specifying the design developed for the Recommendations System and overviews of the algorithms to be developed.	1^{st} December 2021
Prototype The prototype with main core features functional	1^{st} February 2022
Thesis The final report documenting the project and research process and decisions	15^{th} March 2022
Review Paper A review paper reviewing existing systems in the Recommendations domain published in a journal/ conference	1^{st} March 2022
Final Research Paper A research paper introducing the Recommendations System developed at the end of this project	1^{st} April 2021
Public project library A publicly accessible project library/ repository to set up, test and use the developed Recommendations System	1^{st} April 2021

3.3.2 Resource Requirements

The resources required to complete the project are identified based on the objectives, expected solutions, and deliverables. The following are the software, hardware, and data resource requirements.

Software Requirements

- **Operating System(Linux/ macOS/ Windows)** - Linux will be the default choice for development since of the ease of support for multiple development tools and performance benefits. macOS/ Windows will be used for research documentation & study purposes.
- **Python** - The language that will be used to create the Machine Learning & Deep Learning models. Python is an all-purpose language that has been used in many projects that integrate with data science.
- **Tensorflow/ Scikit learn Python packages** - Libraries that will be used to support model development, training & testing.
- **Golang/ NodeJS** - The API that will be used to communicate with the ML backend and the front-end. Golang will allow the application to support concurrency and multi-threaded communications while being extremely lightweight and fast. This will be used to avoid any bottlenecks that could occur at this point in the system. NodeJS will be kept as a secondary option in the case of requiring any pre-built features that are not directly supported by Golang & aren't directly relevant to the research.
- **JavaScript (React)** - The front-end of the application, where recommendations will be shown. This is also an important part of the project since it will be the users' point of interaction with the system.
- **PyCharm/ VSCode/ GoLand** - Development environments to support development of the project.
- **Google Colab** - Cloud development environment to build, train & test ML & Deep Learning models.
- **Zotero** - Research management tool to save and backup research artifacts & manage references.
- **Overleaf/ MS Office/ Google Docs/ Canva/ Figma** - Tools to create reports, figures & documentations.
- **Google Drive/ GitHub** - To backup files & code related to the project

Hardware Requirements

- **Core i7x Processor(8^{th} generation) or above** - To be able to perform high resource intensive tasks.
- **Nvidia 1050Ti GPU or above** - To manage training processes of data science models.
- **16GB RAM or above** - To manage data-sets & development environments.

- **Disk space of 40GB or above** - To store data & application code.

Data Requirements

- **Non-fungible Token data** - From OpenSea open-API.
- **Twitter data** - From Twitter developer API.
- **Google Trends data** - From Google Dataset Search & unofficial Google Trends Python API (Pytrends).
- **Ethereum Smart Contract data** - From Etherscan

Skill Requirements

- Creation of required Recommendation Systems.
- Ability to create optimized Machine Learning & Deep Learning models.
- Research writing skills.

3.3.3 Risk Management

The following are the risks identified prior to starting the project with possible mitigation steps.

Table 3.3: Risk Mitigation Plan

Risk Item	Severity	Frequency	Mitigation Plan
Loose access to on going development code	5	2	Keep all code backed up on GitHub & external backup
Corruption of documentation	4	4	Follow a cloud-first documentation approach and backup on a weekly basis
Inability to complete all expected deliverables within the allocated time	4	2	Work on deliverables on a priority basis.
Inability to explain the research work done due to illness	2	1	Have a recording of demonstration and detailed documentation with explanation

3.3.4 Chapter Summary

This chapter covered the research, development & project management methodologies with all the requirements, the reasoning for selection of each requirement and any foreseeable risks with a mitigation plan.

CHAPTER 4: SOFTWARE REQUIREMENTS SPECIFICATION

4.1 Chapter Overview

This chapter focuses on identifying possible stakeholders of the project by taking a look at all possible points of interaction with the system with the use of a rich picture diagram, gathering their perceptions to analyse and come up with possible expected use cases, functional and non-functional requirements of the prototype.

4.2 Rich Picture

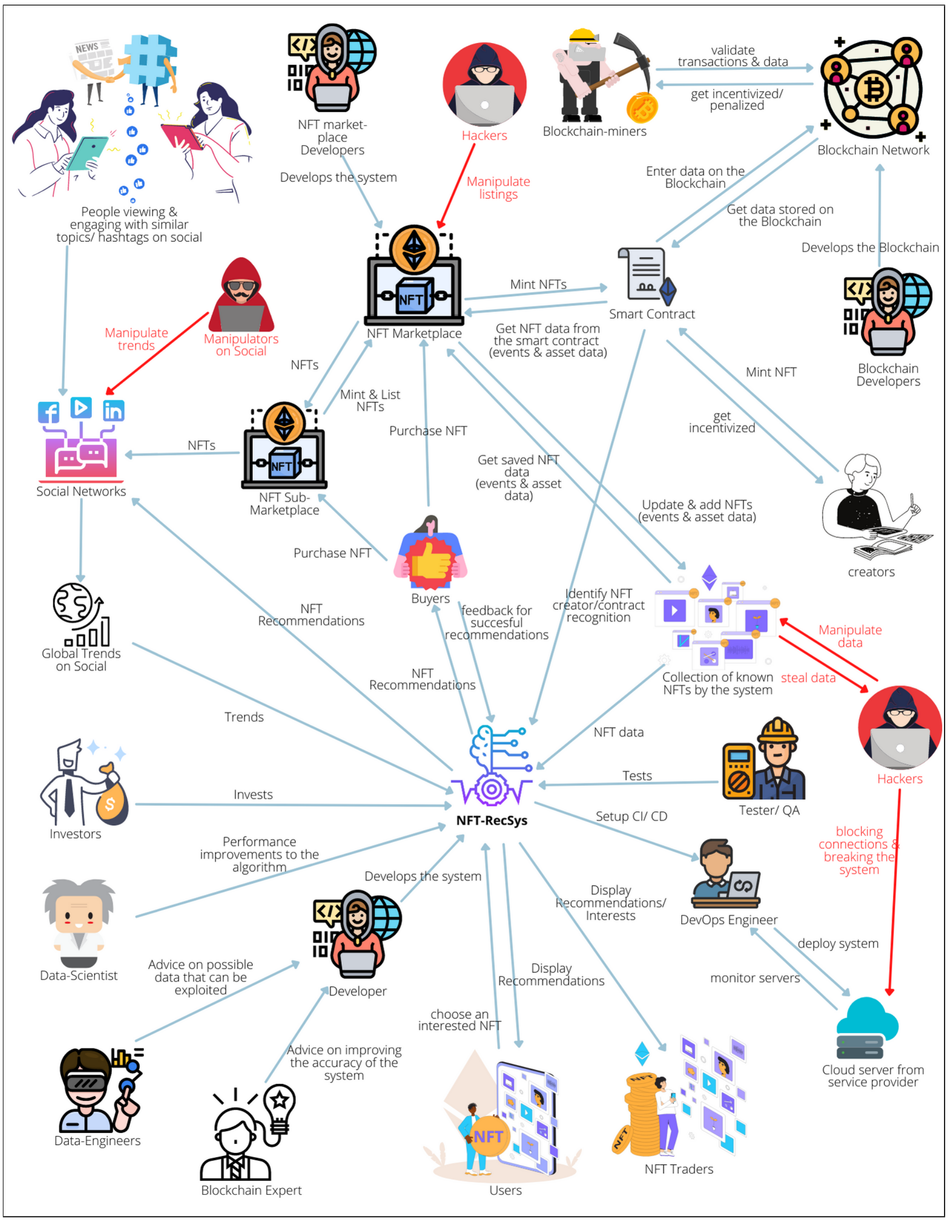


Figure 4.1: Rich Picture Diagram *(self-composed)*

The above Rich Picture diagram shows a helicopter view of how related parties in the rest of the world interacts with the system. It is used to understand the possible interactions that are expected to happen when the system is functional.

4.3 Stakeholder Analysis

The Stakeholder Onion Model illustrates recognized stakeholders who are associated with the system, along with an explanation of each stakeholder's involvement in the system, in Stakeholder Viewpoints.

4.3.1 Stakeholder Onion Model

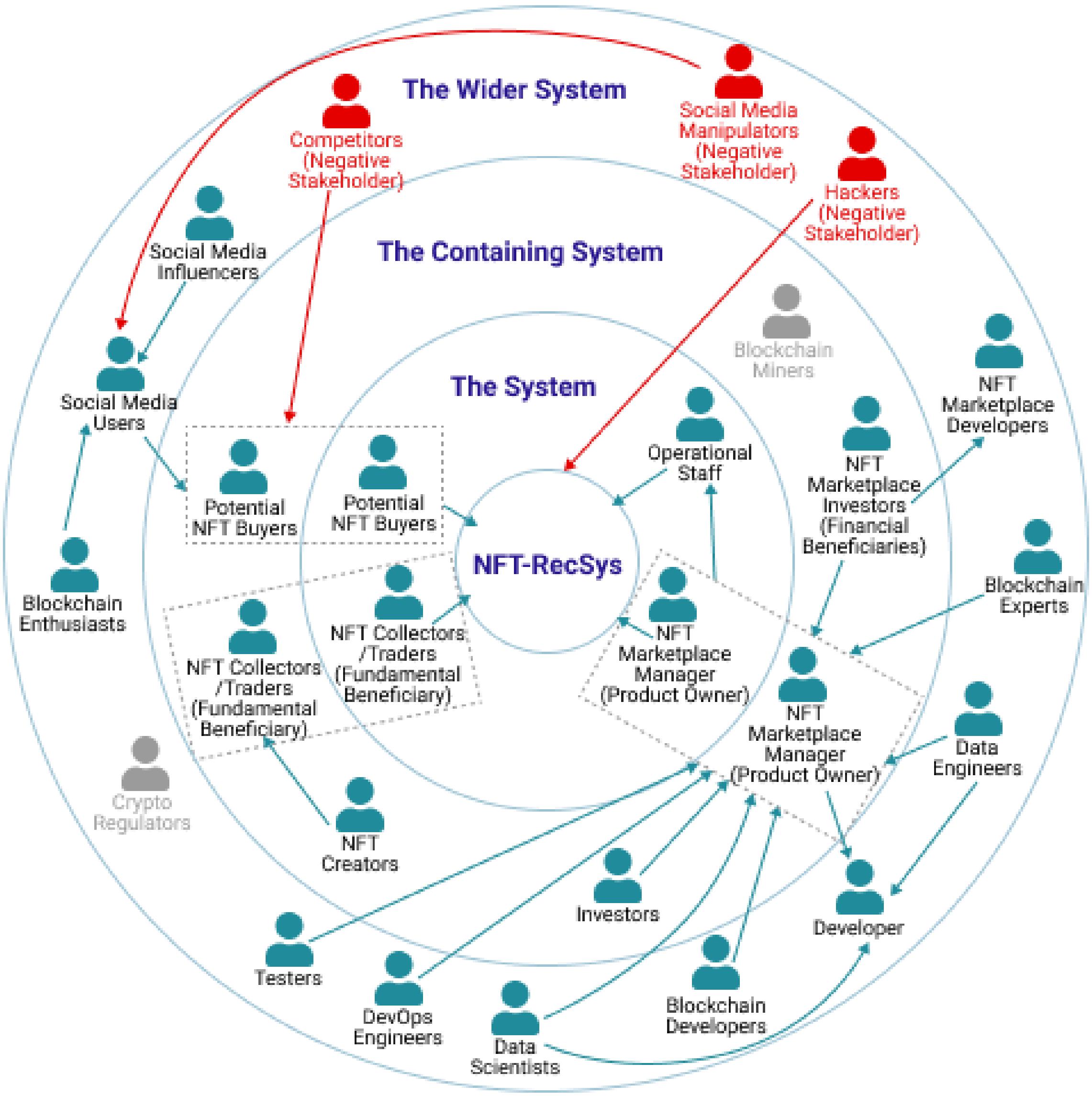


Figure 4.2: Stakeholder Onion Model *(self-composed)*

4.3.2 Stakeholder Viewpoints

Table 4.1: Roles and benefits of identified stakeholders

Stakeholder	Role	Benefits/ Role Description
Developer	Financial Beneficiary	Develops the system
Investors		Makes a profit out of the investments put into marketing, deployments and development of the system
NFT Marketplace Developers	Operational - Maintenance	Integrates the system into NFT Marketplaces.
Blockchain Experts	Expert, Quality Regulator	Provides expert advice & insights into domain knowledge, to improve the system's performance.
Data Scientists		Provides performance improvements for the performance of the Data scienc models/ algorithms used.
Data Engineers		Provides advice on possible data that can be exploited, to make the best possible recommendations.
NFT Creators	Financial Beneficiary	Gets a better opportunity to get their creations in the eye of potential buyers. Makes a profit by selling creations to people who are interested in the creations.
NFT Traders/ Collectors	Fundamental Beneficiary	It becomes easier for traders to sell NFTs as well as explore more NFTs to purchase. It also allows them to explore NFTs that may be worth collecting for a future trade.
Potential NFT Buyers		It becomes more convenient for these parties to explore NFTs that they're interested in.
NFT Marketplace Manager	System Owner, Operational - Administration	Inputs data sources for opinion mining, sets default biases. Makes sure that the system is up & running, while managing the operational staff.

Operational Staff	Operational - Support	Makes sure that the system is up & running, while attending to users' requests & issues.
DevOps Engineers	Product Deployment & Maintenance	Deploys the system to the cloud and make sure that it's up & serving users, without throttling.
Social Media Influencers	Operational - Secondary	Influences users on social media and drives trends.
Social Media Users	Operational - Secondary & Fundamental Beneficiary	Get influenced to search for items of interest and possibly turn into potential NFT buyers.
Hackers	Negative Stakeholder	May manipulate listings in NFT market places.
Competitors		May build competing products that outperform/ undercut pricing.
Social Media Manipulators		May manipulate users on social media & drive trends that a majority of users aren't interested in.
Blockchain Enthusiasts	Operational	Helps drive awareness and keep the public up to date with the latest releases & feature updates.
Blockchain Miners	Operational - Secondary	Helps keep Blockchains up & running by validating the data on the network.
Crypto Regulators	Quality Regulator	May have an impact as a regulator, if the system is used by mainstream networks.
Testers	Quality Inspector	Tests the system & ensures that it's suitable to run in production.

4.4 Requirement Elicitation Methodologies

In order to gather requirements for the development of the research project, there were multiple requirement elicitation methodologies that were followed. literature review, interviews, survey & prototyping were the methodologies chosen for this purpose. The reasons to choosing the specified requirement elicitation methodologies have been discussed below.

Table 4.2: Requirement Elicitation Methodologies

Method 1: Literature Review
At the inception of the project, the author has done a thorough literature review to identify research gaps that are open in the desired field of study and a chosen domain of interest. In order to understand research gaps available in technologies that can be applied, existing systems were studied together with relatable technologies that are possible to be applied to the existing systems that were mentioned in literature.
Method 2: Interviews
Interviews were conducted as a means of gathering expert-insights into domain-specific requirements and also to identify the best possible way to solve the problem at hand while contributing to the body of knowledge through research. Due to the domain being new and the required technical knowledge being specific, interviews were identified to be the best-possible source of knowledge to gather requirements that align with the research gap. This method also allowed to get qualitative feedback on the proposed system making it possible to identify any drawbacks/ challengers that may have to be addressed while prototyping.
Method 3: Survey
As a means of conducting a survey, questionnaire was used as a tool to gather requirements and insights from potential users of the proposed system. This form of survey will aid the author in comprehending people's cognitive processes and the expectations they have for the prototype. It will also allow the author to clarify if the proposed solution would be helpful to intended users.
Method 4: Prototyping
Since the project was chosen to follow the *Agile* Software Development Life-cycle, prototyping would allow the author to recursively try out various alternative implementations to identify any areas of improvement while testing and evaluating the prototype.

4.5 Analysis of Data & Presentation of the Outcome through Elicitation Methodologies

The analysis of data gathered through the chosen means of requirement elicitation have been presented below.

4.5.1 Literature Review

Table 4.3: Findings through Literature Review

Finding	Citation
In completion of the review of literature, it was identified that a Recommendations System for NFTs would benefit the majority of users to make purchase decisions as well as allow them to explore relevant items, that would in return benefit the market places, creators & traders who are selling them as Recommendations Systems have proven to improve sales of e-commerce sites in the past.	(Naumov et al., 2019; Vanderbilt, 2021)
When exploring technologies that can be applied to achieve the required outcome, it was understood that the use of Deep learning hasn't been able to improve the output of recommendations compared to other fields of applications, in most cases.	(Choi et al., 2021)
It was identified that implementing a custom hybrid ensembled model with the injection of social media trends has not been explored in literature.	(Ayushi and Prasad, 2018; Cheng and Lin, 2020)
The use of data from similar users' timelines for recommendations has been mentioned as possible future work.	(Chen and Hendry, 2019)
Pricing of NFTs & contract recognition data have not been considered for any previous implementations of Recommender Systems	(*What are you missing? Using basic machine learning to predict and recommend NFTs with OpenSea data - OpenSea Blog* 2020)
The only study related to recommending NFTs only recommends NFT collections that a user may be interested in, but not actual NFTs themselves.	(ibid.)

4.5.2 Interviews

In order to get opinions of technical as well as domain expertise, interviews were conducted with experts from the respective fields. Experts & researchers in ML, Recommendation Systems and Blockchain were chosen to be interviewed in order to establish project requirements. 3 Blockchain experts, 1 NFT Creator, 1 Senior Data Engineer, 2 PhD students in ML and a Data science engineer were interviewed. The outcome of interviews were processed to a **thematic analysis** based on the following themes.

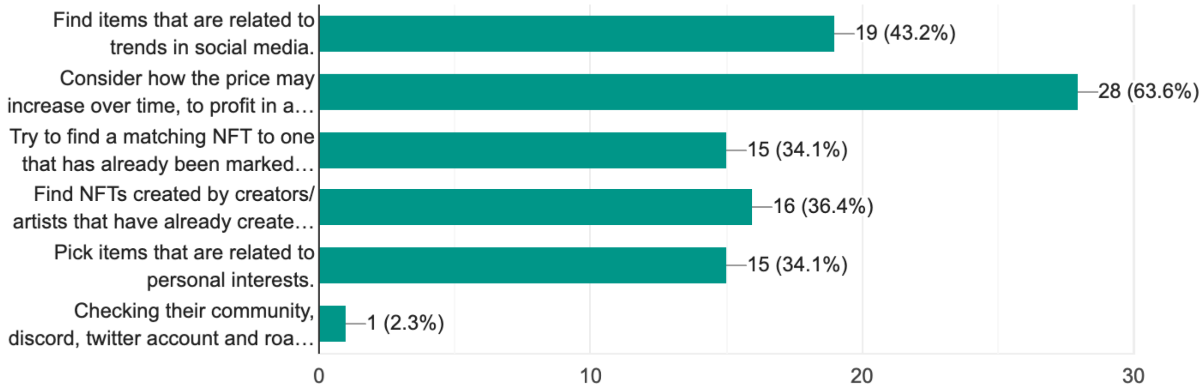
Table 4.4: Thematic analysis of interview findings

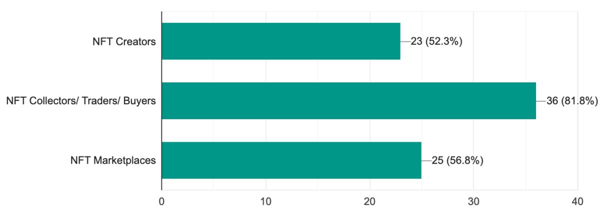
Theme	**Analysis**
Collection & pre-processing of available data.	As this is expected to be a Data science project, the main concern that all participants had was the availability of data. Clustering of available data was suggested to identify possible patterns by ML experts, while Blockchain experts suggested the use of publicly available data on the Blockchain such as details from Smart-Contracts to be used to improve the quality of recommendations.
Applicable Recommendation Techniques	The opinion of majority of the interviewees was that this project would benefit more by the use of rule-based algorithmic recommendation models instead of DL models due to the constraint of . According to technical experts, having a specialized recommendation model built using algorithms is very highly accepted in industrial applications. They seem to perform better in most new domains according to PhD researches. Even some of the biggest e-commerce organizations in the world seem to benefit a lot by custom-built recommendations algorithms tailored to specified use-cases according to research & development experts in Recommendation Systems.
Integration of Opinion Mining into Recommendation Systems	Domain experts thought that integrating trends and other social opinion will add value to the recommendations. They were also interested in identifying a possibility of checking for the sentiment represented by the opinions as well. When considering social sentiment, Tweets/ opinions of well-known influencers may play a bigger effect into the value of curtain NFTs.

Research gap & scope	The technological experts thought that the method that the author proposed was very innovative and that according to their knowledge, they haven't seen a similar integration to the suggested architecture in previous applications.
Creating the bias for a Hybrid Recommendations Model	While some of the interviewees suggested the use of a fixed weighted bias, others suggested a variable bias. The method applicable for variable bias or the best-possible fixed bias can be tested via continuous prototyping & evaluation. The use of user-input was also suggested to identify a possible expected bias.
Prototype features & suggestions	The Data science experts were very interested in seeing a Recommendations System built purely using custom algorithms with the help of vectorization functions that many ML libraries support. The use of transfer learning or pre-trained models were suggested for NLP parts of the implementation.
Understanding a buyer's decision making for automation	The value proposition was identified to be created by an external entity based on contract & token Ids stored on the blockchain. Due to the difference in real world trust and blockchain trust, this may have to be inferred from the available data such as past contract data and social sentiment from trends.
The necessity of NFT-RecSys & contributions	As the first research study related to a Recommendations System for NFTs, the interviewees thought that the contribution to the domain will be of great value and also, since the hybrid architecture of the proposed system is novel, the contribution to the technological domain would help the advancement of the quality of recommendations in future implementations. It was also understood that it's difficult to find specific NFTs based on tags/ characteristics. Furthermore, it was revealed that Sri Lanka does not have Machine Intelligence/ Data science driven Recommendation Systems in all local e-commerce stores.

4.5.3 Survey

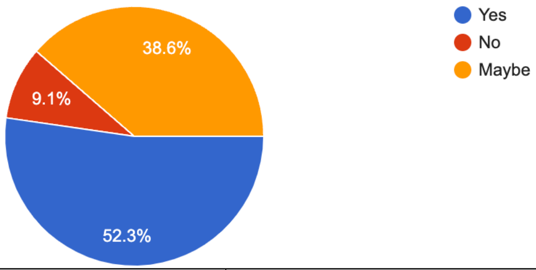
Table 4.5: Analysis of replies to questionnaire

<table>
<tr><td>Question</td><td>How will you decide which NFT to purchase?</td></tr>
<tr><td>Aim of question</td><td>To understand how a potential buyer would proceed to purchase an NFT.</td></tr>
<tr><td colspan="2">Findings & Conclusion


A majority of the participants thought that considering the price increase over time would be the primary factor of consideration when purchasing an NFT, while the second most impact to be considered was trends in social media. Finding NFTs that have been created by creators/ artists who have created valuable NFTs in the past, an NFT that is similar to what is already highly valuable and picking items related to personal interests saw similar weightings when making purchase decisions.</td></tr>
<tr><td>Question</td><td>Who do you think will be benefited from using this system?</td></tr>
<tr><td>Aim of question</td><td>To identify the beneficiaries of the proposed system.</td></tr>
<tr><td colspan="2">Findings & Conclusions


While more than 50% of participants aggreed that the proposed system would benefit the suggested beneficiaries, 81.8% thought that NFT collectors/ traders/ buyers would benefit. Since, they are the ultimate target users, it's satisfying to see such positive responses.</td></tr>
<tr><td>Question</td><td>Do you think that this system would benefit people who have no expertise in Blockchain/ NFTs as well as people who have a decent amount of expertise in Blockchain/ NFTs?</td></tr>
<tr><td>Aim of question</td><td>To identify how valuable the system would be to people of all levels of expertise in Blockchain/ NFTs</td></tr>
</table>

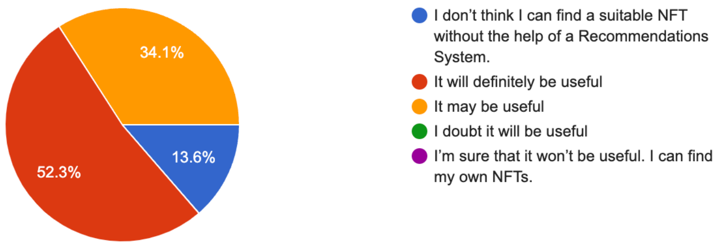
Findings & Conclusion



With majority of the responses suggesting that people of all levels of expertise in Blockchain/ NFTs would benefit from the system depicts that the proposed system would be beneficial for above-average users as well.

Question	**How much do you think that a Recommendations System would benefit you, if you ever plan on purchasing an NFT?**
Aim of question	To identify if the respondents think that the system would benefit them.

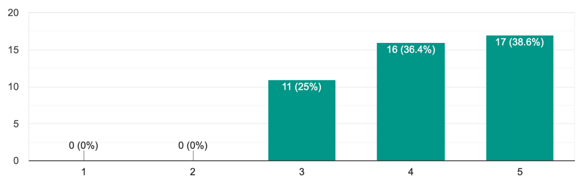
Findings & Conclusion



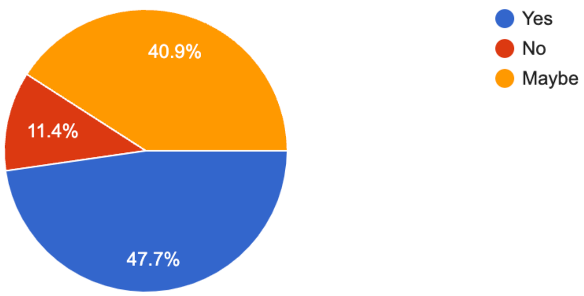
52.3% of users thought that a Recommendations System would definitely be useful to them if they plan on purchasing an NFT, while 34.1% thought that it may be useful. Meanwhile, 13.6% of users thought that they don't think that they could find a suitable NFT without the help of a Recommendations System. 100% of the results were aligned towards seeing a possible benefit of the proposed system.

Question	**How much would you expect a Recommendations System that considers social media trends to be beneficial for businesses to integrate into their online platforms?**
Aim of question	To identify the importance of the technological contribution in the project

Findings & Conclusion



The results from this question suggests that the technological contribution that has been highlighted in this project, which addresses an advancement of development of Recommendation Systems is expected to be extremely beneficial for business applications.

<table>
<tr><td>Question</td><td>Do you think that a user would benefit more if one platform provides recommendations that differ from another platform with the same dataset?</td></tr>
<tr><td>Aim of question</td><td>To identify if the proposed Recommendations System will benefit from implementing a Reinforcement Learning technique or a variable bias to adapt and suite different platforms.</td></tr>
<tr><td colspan="2">Findings & Conclusion


A majority of participants thought that having varied recommendations in different platforms, using the same recommendations algorithm. This leads to the requirement of implementing a variable bias towards the factors considered for recommendations or implementing a reinforcement learning technique, for the model to adjust based on user-inputs. Having a pre-configurable bias will also allow to achieve this, but the results from recommendations may not be optimum.</td></tr>
<tr><td>Question</td><td>What functionalities would you like to have in a Trading Recommendations System for Non-fungible Tokens?</td></tr>
<tr><td>Aim of question</td><td>To identify the non-function requirements of the system, that would make the system as user-friendly as possible</td></tr>
<tr><td colspan="2">Findings & Conclusion
Most responses form the participants revolved around considering price-predictions when making recommendations. There were also suggestions to integrate trending crypto news to the system. Suggesting potential NFTs that suit a person's personal interests were also suggested to be integrated.</td></tr>
</table>

4.5.4 Prototyping

Through iterative prototyping, there were many requirements & challengers that emerged. Firstly, there was no dataset. The data had to be pulled from an open API and filtered. The main challenge that was met here was the overwhelming amount of data that was received related to each NFT and rate limits of the API. The data received had to be filtered quite a lot and the most usable data points possible to be used for recommendations had to be identified & extracted. Not all NFTs contained usable content-information. This had to be addressed with normalizing

several fields and finding alternatives to map items using other available data.
The integration of social trends data brought in a new valid perspective that could be used for recommendations.

4.6 Summary of Findings

Table 4.6: Summary of Findings

Id	Finding	Literature Review	Interviews	Survey	Prototyping
1	The proposed system would benefit experienced & inexperienced users searching for NFTs as well as NFT creators, traders & market places	✓	✓	✓	
2	The limits of Recommendation Systems can be pushed without the use of Deep learning, by the application of various hybrid ensemble models	✓	✓		
3	The integration of social media trends would be beneficial to improve recommendations produced by a Recommendations System	✓	✓	✓	✓
4	The identified research gap would contribute to both the Blockchain-NFT domain as well as the advancement of Recommendations Systems & ML	✓	✓	✓	
5	Building custom use-case specific algorithms for the Recommendations System is prefered over the use of pre-built models from a business application perspective		✓		
6	Having a method of price-prediction & using the prediction data to make decisions on recommendations would benefit users		✓	✓	
7	Using data-clustering techniques to identify contract-recognition & data tags are expected by advanced-users		✓		
8	Personalized recommendations could be achieved by the use of information extracted from the Blockchain with related to a user's public key. Past purchases of NFTs made by users can be considered.	✓	✓		
9	It would be good to have a user-interface that allows the user to choose the bias/ his primary concerns when expecting a recommendation, to provide the perfect recommendation for each user.		✓		

9	Having a adaptable, variable Recommendations Model that allows different platforms to have varied recommendations is preferred.		✓	✓	
10	Having a sufficient set of well-cleaned & pre-processed data would be vital for the performance of the system	✓	✓		✓
11	Opinions of well-known influencers could have a bigger impact on the decision-making process of a majority of users.		✓		

4.7 Context Diagram

Prior to development, the system's boundaries and interactions should be determined. The system's context is depicted in the diagram below.

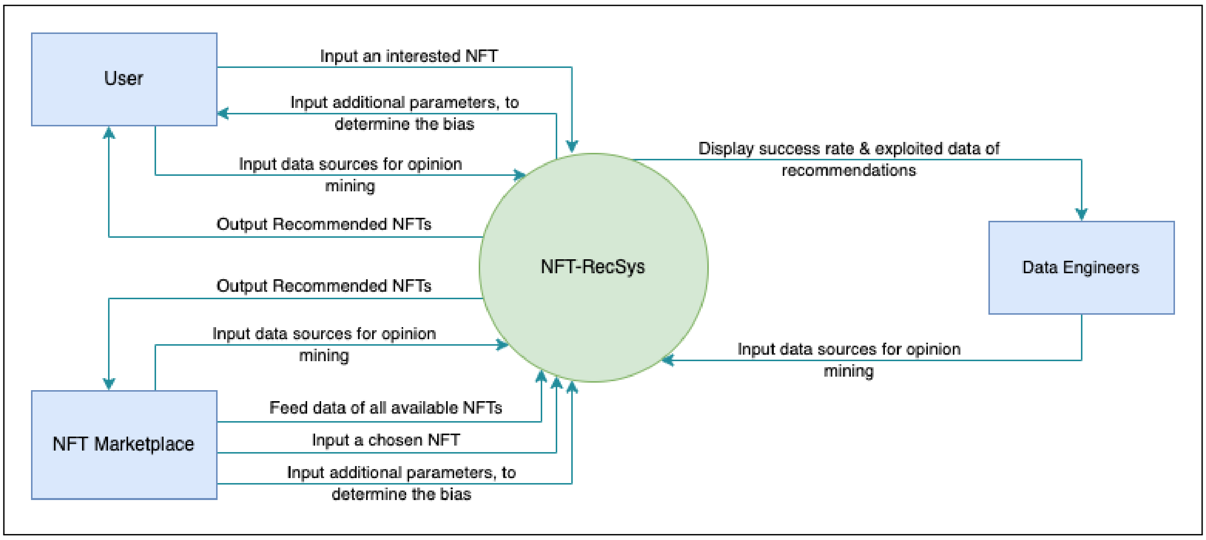


Figure 4.3: Context Diagram *(self-composed)*

4.8 Use Case Diagram

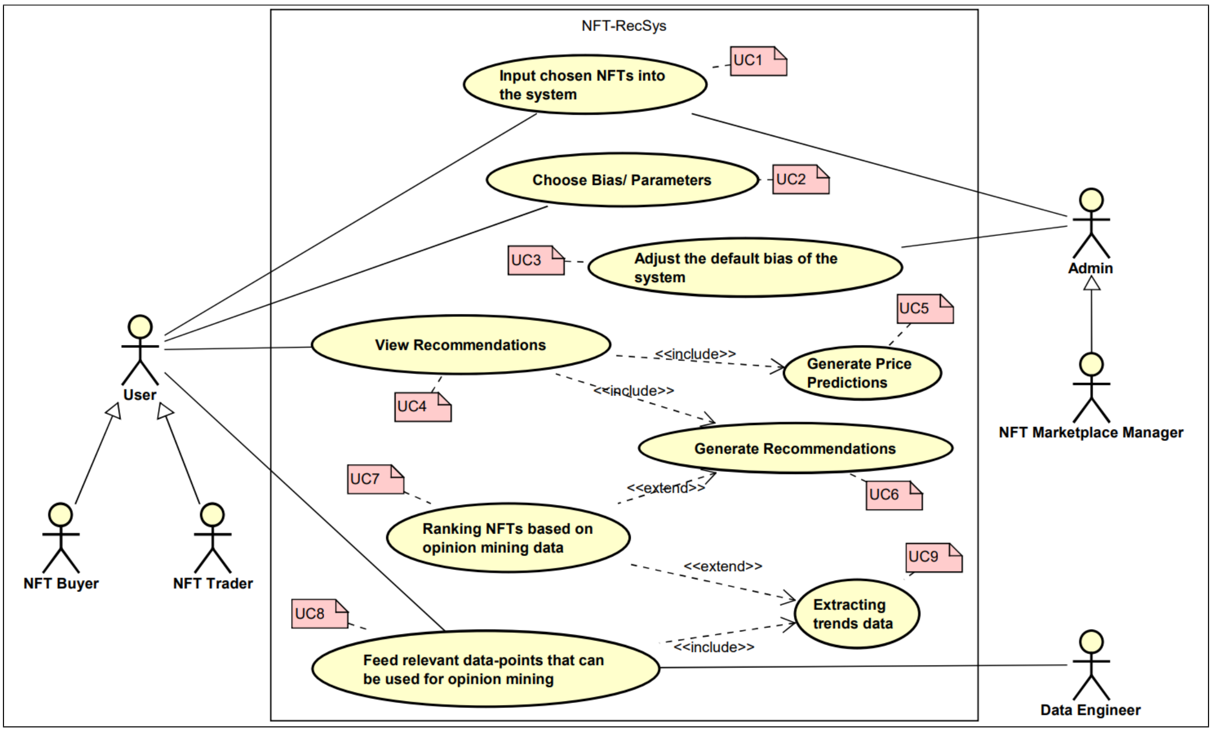


Figure 4.4: Use Case Diagram *(self-composed)*

4.9 Use Case Descriptions

Table 4.7: Use case description UC:04

Use Case	View Recommendations
Id	UC:04
Description	Display the most relevant NFT Recommendations based on the user's selection & available data in the system.
Primary Actor	User
Supporting Actors (if any)	none
Stakeholders and Interests (if any)	Admins, NFT Traders, NFT creator
Pre–Conditions	The NFT data and trends data have to have been pre-processed. The recommendations have to have been generated.
Post Conditions	Success end condition: The user is presented with recommended NFTs.
Trigger	A user wishes to find similar NFTTs to those that are currently being viewed or to explore possible interests based on past views.

Main Success Scenario	• User chooses the option to view recommendations. • System recognizes the user's preferred bias for recommendations. • System filters out and diversifies recommendations based on the user-bias and general bias that has been set in the system. • System displays the recommended NFTs.
Variations	A user can be presented with recommended NFTs based on past interests shown and views in a feed similar to a social network/ e-commerce store.

Table 4.8: Use case description UC:07

Use Case	Ranking NFTs based on Opinion mining data
Id	UC:07
Description	Rank NFTs for recommendations based on gathered social media trends data, opinion mining data & content in NFTs.
Primary Actor	none
Supporting Actors (if any)	Admins, Users
Stakeholders and Interests (if any)	NFT Collectors, NFT Traders, NFT creator
Pre–Conditions	New data-points have been added by an admin or a user and the trends have been extracted.
Post Conditions	Success end condition: Rank NFTs
Trigger	An admin or a user wishes to find NFTs that have content related to what's trending on the internet at the current moment in time.
Main Success Scenario	• System matches data of each NFT in the current data-set with extracted trends data. • System calculates a score for each NFT based on the matches & impact of the identified trends. • System re-ranks NFTs based on the calculated scores.

Variations	When recommendations are produced using other methods apart from trends, the data ranking scores generated here can be used to re-rank the recommendations when presenting to a user.

4.10 Requirements

4.10.1 Functional Requirements

The MoSCoW technique was used to determine the priority levels of system needs based on their importance.

Table 4.9: Levels of priority according to the "MoSCoW" technique.

Priority Level	**Description**
Must have (M)	This level's requirement is a prototype's core functional requirement, and it must be implemented.
Should have (S)	Important requirements aren't absolutely necessary for the expected prototype to work, but they do add a lot of value.
Could have (C)	Desirable requirements that are optional and aren't deemed essential critical to the project's scope.
Will not have (W)	The requirements that the system may not have and that are not considered a top priority at this time.

Table 4.10: Functional requirements

FR ID	**Requirement**	**Priority Level**	**Use Case**
FR1	Users must be able to add a chosen NFT to be considered as the reference point to generating recommendations.	M	UC1
FR2	Admins should be able to add a collection of NFT to be used as recommendations.	S	UC1
FR3	The system could be able to fetch relevant data of the NFT using an entered token Id.	C	UC1
FR4	Users must be able to set/ adjust the bias and parameters to be used by the Recommendations System using parametric selections prior to generating recommendations.	M	UC2

FR5	Admins should be able to adjust the default bias of the Recommendations System.	S	UC3
FR6	Users must be able to view recommendations with the click of a button.	M	UC4
FR7	The prototype could have an option to receive user feedback regarding the satisfaction level of the generated recommendations by the system.	C	UC4
FR8	The system could show the reasons for recommending each item to users.	C	UC4
FR9	The system should generate price predictions and consider the results for recommendations.	S	UC5
FR10	Opinion mining trends data must be used to generate NFT recommendations.	M	UC7
FR11	A user could be allowed to feed data-points such as interested public figures, websites to use as opinion mining data for recommendations.	C	UC8
FR12	Admins should be able to feed data-points such as interested public figures, websites to use as opinion mining data for recommendations.	S	UC8
FR13	User-input could be aggregated and used as a reinforcement learning bias for the Recommendations Model.	C	
FR14	The system will not act as a decentralized system.	W	

4.10.2 Non-functional Requirements

Table 4.11: Non-functional requirements

NFR ID	Requirement	Description	Priority Level
1	Performance	Although recommendations should be provided upon user-input; the recommendations matrix & opinion-mining data can be pre-processed and stored in-memory to be used. Real-time processing isn't essential.	Desirable
2	Quality of Output	The quality of the output should be of the highest possible level, utilizing all the available data.	Important

3	Security	The application should prevent any attackers from manipulating results and extracting user-inputs. Security could be assured by means of testing.	Desirable
4	Usability	Since the purpose of the system is to automate and make it easy for the user to explore NFTs, the usability of the system must be easy for users of all levels of expertise.	Important
5	Scalability	The prototype may open up for testing for many users. Considering the hype around NFTs and the interest in the project, the system may have to support many concurrent user-requests.	Desirable

4.11 Chapter Summary

In this chapter, a Rich Picture Diagram was drawn to illustrate how the system connects with the society to understand the stakeholders of the system. Saunder's Onion model was used to represent the stakeholders with the flow of influence of each stakeholder. Requirement gathering techniques were utilized to gather all the required data and opinions of possible stakeholders of the system. Lastly, the system's use cases, functional, and non-functional requirements were specified based on the insights derived from the requirement elicitation techniques.

CHAPTER 5: SOCIAL, LEGAL, ETHICAL AND PROFESSIONAL ISSUES

5.1 Chapter Overview

The purpose of this chapter is to define the social, legal, ethical & professional difficulties that may arise during the project, with steps taken to mitigate those issues.

5.2 Breakdown of Social, Legal, Ethical and Professional Issues

Table 5.1: Breakdown of SLEP Issues

Social	Legal
• Questionnaire responses were not added to the thesis in a manner that would expose personal opinions of the respondents. Only the summary of responses were recorded. • Interviewers were notified that the responses will be recorded in the thesis & their consent was taken to have their name & designation added to the thesis.	• All programming languages, tools & frameworks that were used were under open source license. • All the source code of the research including data collection & preprocessing will be licensed under GPL3 license. • The system was developed in a manner that no personal data would be required to produce the expected output.
Ethical	**Professional**
• Participants who completed the questionnaires were informed about the project and how they were contributing to it. • There is no fabrication, falsification, or plagiarism in the thesis. All of the data and information given are authentic, and the knowledge and facts that were retrieved were appropriately cited and referenced.	• None of the software or tools utilized to create the prototype were illegal or pirated. Only open source or student licenses were used throughout the process. • The project's outcomes were true to nature and were documented exactly as they were without any modifications. • A high level of research standards were followed throughout the research process.

5.3 Chapter Summary

This chapter identified possible social, legal, ethical & professional issues under each section with mitigation strategies that were followed for the research.

CHAPTER 6: DESIGN

6.1 Chapter Overview

This chapter consists of the design decisions made to come up with a suitable architecture for implementation, based on the gathered requirements. High-level design, low-level design, design diagrams, UI wireframes have been used to convey how the design goals are expected to be achieved while discussing the reasoning for chosen design decisions.

6.2 Design Goals

Table 6.1: Design Goals of the proposed system

Design Goal	**Description**
Performance	The recommendations matrix & opinion-mining data can be pre-processed and stored in memory to be used for recommendations. Since ensembled models are expected to be utilized, concurrency would be ideal to get the output from multiple models at the same time. This could cut down the processing time by 4-5 times (based on the number of models that are required to provide recommendations for the given input).
Correctness	The correctness & quality of the output should be of the highest possible level, utilizing all the available data. Explaining why a user is getting the proposed recommendation will ensure that the user isn't misled into wrong purchase decisions.
Usability	Since the purpose of the system is to automate and make it easy for the user to explore NFTs, the usability of the system must be easy for users of all levels of expertise.
Scalability	The system may have to support many concurrent user requests in a production environment. The backend should be able to handle this. New data should be able to be added to the system with minimum effort.
Adaptability	Since the utilized Recommendation models may have to be altered based on the available data and user requirements in the future, these models should be able to be easily swapped out for new models while ensuring that the system won't break in the process of upgrading, with minimum changes.

6.3 High-Level Design

6.3.1 Tiered Architecture

The system's architecture is depicted in the diagram below. The data, logic and presentation layers are organized in a three-tier architecture.

The research contribution in this system lies in data preprocessing of the *data tier*, recommendations models, and the recommendations diversifier of the *logic tier*.

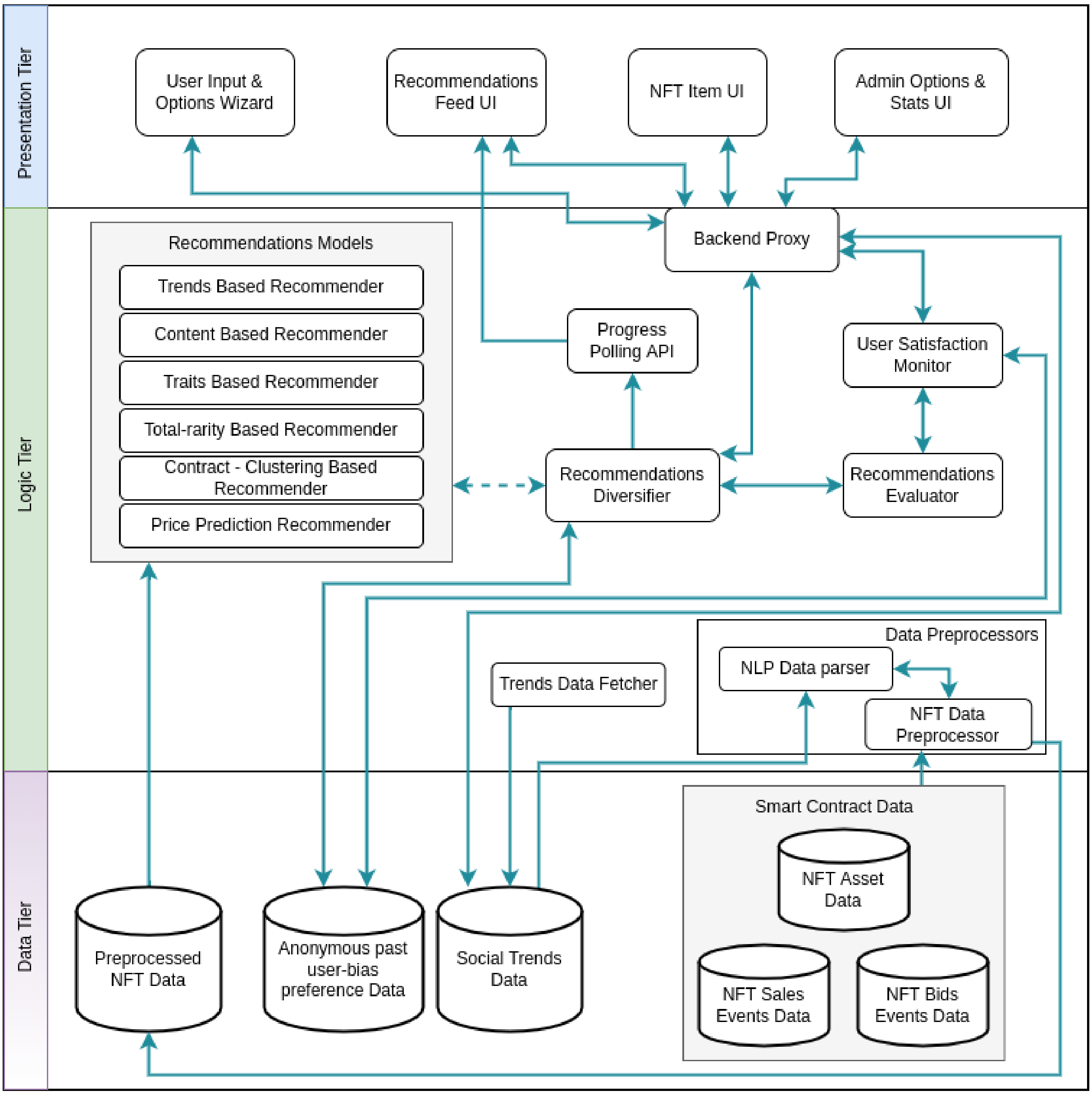


Figure 6.1: Three Tiered Architecture *(self-composed)*

While the entire architecture is represented in a modular approach for ease of understanding, several backend services are expected to work together in the fashion of a distributed microservices architecture when it comes to implementing the proposed architecture.

The reason for following a microservices architecture is to allow the system to scale while ensuring that points of failure can be easily recognized and taken care of separately. The distributed nature of the system is expected to be seen in the connection between the numerous Recommendations Models and the Recommendations Diversifier. These combined through output pipelines will act as an Ensebled Recommendations System. Although the system will be capable of distributing the load at this point, the expectation with the prototype is to run this in a single machine.

The purpose of each module that is represented in the above architecture is described below.

Data Tier

1. Smart Contract Data - Data that is retrieved from Blockchain Smart Contracts. For convenience purposes, the data is fetched from the OpenSea API. Contains all the available data of each NFT.
   (a) NFT Asset Data - All the content of each NFT.
   (b) NFT Sales Events Data - Past sales data from NFT trading.
   (c) NFT Bids Events Data - All the current bids of each NFT.
2. Social Trends Data - Data gathered from social trends sites (Twitter, news sites, etc.)
3. Anonymous past user-bias preference Data - Each user's preferred bias is stored anonymously. This can be identified by a user's selection based on their requirement or based on the feedback received for each recommendation. This can be a temporary data store that can be cleared once the user session has ended.

Logic Tier

1. Data Preprocessors - The preprocessing code required to modify/ extract required data that is usable for recommendations from all the available data.
   (a) NLP Data parser - Responsible for extracting all the required data from what was collected through data mining techniques.
   (b) NFT Data Preprocessor - Used to modify and separate data that can be utilized from smart contracts and processed trends data.
2. Recommendations Models - The various models that are used to provide recommendations based on identified diverse data points.
3. Recommendations Diversifier - The module that combines the recommendations produced by all the Recommendations Models, considering the bias.
4. User Satisfaction Monitor - The feedback received by users will be filtered and updated through this module, to update the moving bias while preserving user anonymity,

5. Recommendations Evaluator - The module that evaluates the user's satisfaction with the recommendations produced, to separately identify under-performing & high-performing models.
6. Progress Polling API - The web-polling API that will be used to update the progress of recommendations generation in the frontend.
7. Backend Proxy - The interface that exposes the backend services to the frontend.
8. Trends Data Fetcher - Fetch global trends data from social APIs or by scanning through news websites.

Presentation Tier (Client Tier)

1. User Input & Options Wizard - The UI that is presented to the user to enter the desired NFT(s) to be considered to recommendations as well as desired parameters and data-points (for advanced users).
2. Recommendations Feed UI - The UI that will show all the recommendations generated for a user. This will be similar to a home page on Youtube/ any other social network.
3. NFT Item UI - The UI that will show a chosen NFT with its data and recommendations.
4. Admin Options & Stats UI - The UI that will be exposed to a system Admin, allowing him to view the stats such as the general bias of the system. This will have options to define the data sources to be used for trends based recommendations and to adjust the bias.

6.4 System Design

6.4.1 Choice of the Design Paradigm

Although the author was very tempted to use OOAD (Object-Oriented Analysis and Design) to build the prototype due to the ease of extendability and further development of the system, the decision was made to use **SSADM (Structured Systems Analysis and Design Method)** based on the following factors.

- The project's core research component is inclined towards Data Science. Therefore, it doesn't gain a noticeable benefit by using Object Oriented approaches.
- The programming languages that are expected to be used for implementation don't support OOP by nature.
- Ease of implementation of an MVP (Minimum Viable Product) for demonstrating the research application using the prototype.
- The time constraint of having to implement & document research within the time span of 10 months.

6.4.2 Data Flow Diagram

The Level 1 Data Flow Diagram presented below provides a more extensive breakdown of the components of the Context Diagram that was presented in the SRS.

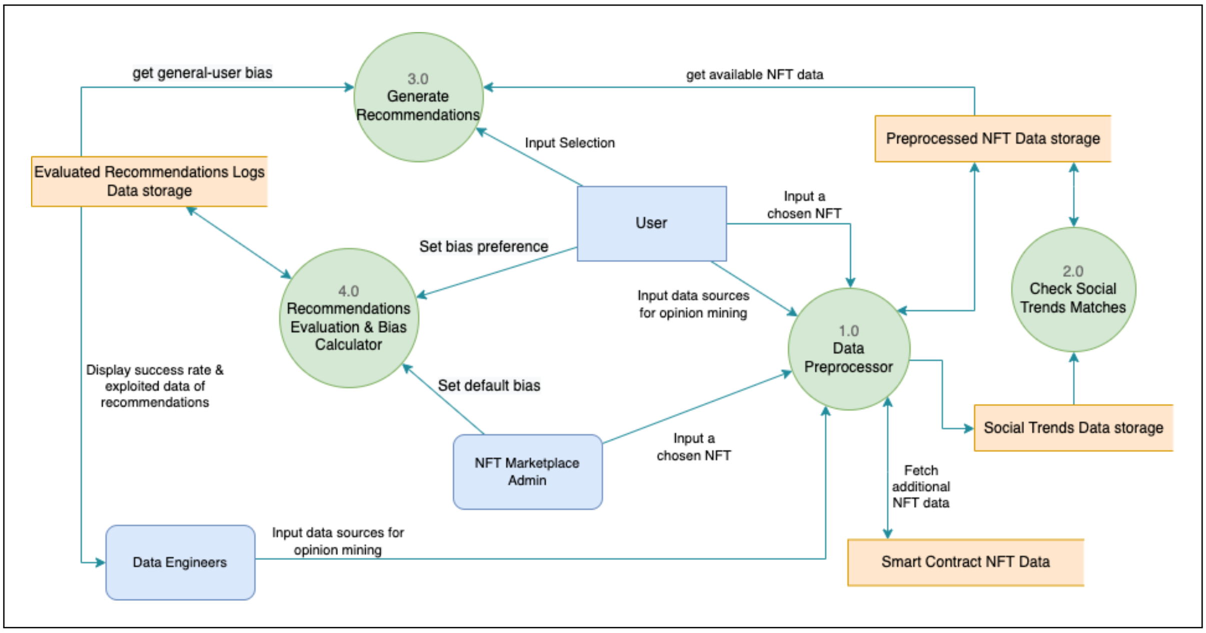


Figure 6.2: Data Flow Diagram - Level 1 *(self-composed)*

The Level 2 Data Flow Diagram presented below provides a more extensive breakdown of the components of the above Level 1 Data Flow Diagram.

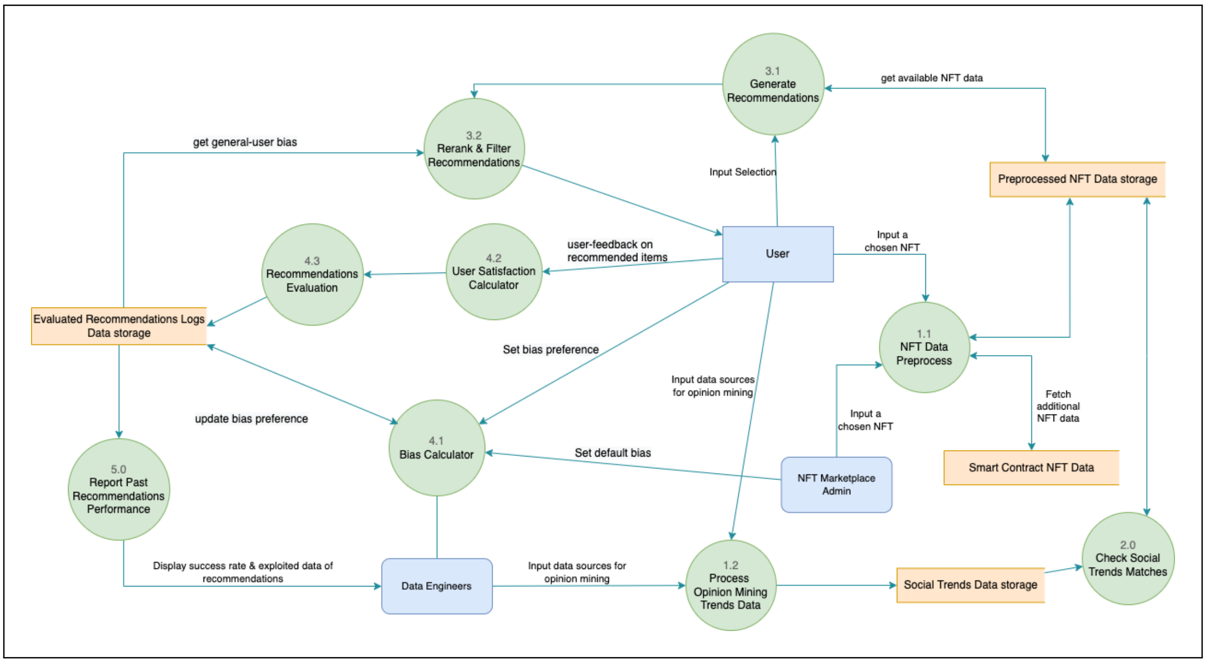


Figure 6.3: Data Flow Diagram - Level 2 *(self-composed)*

6.4.3 Algorithm Design

When studying available data in the system, it was identified that cross-collection NFTs cannot be recommended using the same concepts & data points followed for inter-collection matches. Therefore, multiple algorithms were considered to get a diverse set of recommendations.

Infusing trends matches into Recommendations

The equation composed below is designed to be used to calculate the total trends score for an item. The methods of utilization of this score for recommendations have been discussed following the breakdown of the equation.

$$T_{t_s,i} = \frac{\sum_{i_s=1}^{N_{i_s}} \left[\sum_{k_w=1}^{k_w} s_c \left(\frac{t_{vt,c}}{Med(T_{vt})} \right) \frac{mu}{(\mu + n_m)} \right]}{N_{i_s}} \tag{6.1}$$

Equation for social trend-match score for recommendations *(self-composed)*

$T_{t_s,i}$ - Total trends score for one item

N_{i_s} - Total number of information sources

i_s - Source of information

k_w - Number of keywords in the current item

s_c - Sentiment score surrounding chosen trend content

m - Match value, a Boolean used to check if the current evaluated content contains the chosen trend to be matched against.

u - User priority, used to check the current user's interest in the chosen trend. This is 1 by default

$t_{vt,c}$ - Tweet volume at this moment in time of the chosen content

$Med(T_{vt})$ - Median Tweet volume at this moment in time

μ - Constant, set to 0.1 to avoid division by 0 error for today's trends

n_m - Number of days between the current day & the day of the trend.

The following equation extracts the calculation of the impact score of the chosen trend (i_t), as described above. Twitter data has been taken as the example source here. The data source can be even an internet forum.

$$i_t = \frac{t_{vt,c}}{Med(T_{vt})} \tag{6.2}$$

Equation for the calculation of the impact score of a chosen trend *(self-composed)*

For trends that don't have a measurable volume, $t_{vt,c}$ can be taken as $(T_{vt}min - 1)$ to give it the lowest possible value, or as $Med(T_{vt})$ to omit the impact score all-together.

The algorithm, $T_{t_s,i}$ can be applied to inter-collection recommendations as well, if each NFT in the collection has unique names and descriptions. Using unique traits didn't seem to make sense for comparison with this algorithm, but it may be valid if it can be proved that the traits can be matched with trends data.

The Total trends score for one item calculated above can either be taken for recommendations as to the top N items or as an absolute similarity match with other chosen items' trends scores.

The beauty of this equation is that it isn't necessarily required to be applied for only NFT recommendations. It can be used to enhance any content-based recommendations model. It can be seen as another way of infusing collaborative filtering, without the collection of user-specific data by the platform that integrates the presented Recommendations Architecture.

Recommendations based on Rarity

$$T_{r,t} = \sum_{t=1}^{Nt} \frac{1}{\left(\frac{c_t}{T_N}\right)} \tag{6.3}$$

Equation for the calculation of the total trait rarity score of an NFT *((rarity.tools, 2021b), (rarity.tools, 2021a))*

$T_{r,t}$ - Total rarity of a trait

Nt - Total number of traits in the NFT

c_t - Trait count of the chosen trait (number of occurrences in the collection) T_N - Total supply of NFTs in the collection

The absolute difference between the total rarities is calculated when an NFT from a collection is chosen. The lowest scoring items are recommended to the user. This gives the NFTs that may be as closely valuable as the initially chosen NFT. This allows recommending NFTs that don't have unique content descriptions.

Furthermore, the traits are fed into a Content-based Recommendations Model to get NFTs with the most similar traits to be recommended.

Varying Bias for Recommendations Diversifier

Finally, all these recommendations produced by algorithmic models had to be presented to the user suitably. Instead of going with a weighted bias which was recommended by the experts that were interviewed, it was decided to make this bias variable with time.

The reason for opting for this in contrast to having pre-trained weights & biases using a Neural Network architecture that Amazon successfully attempted with its recent Autoencoder (Larry, 2019) DL model was to allow a more optimized output, without having to retrain the model. Another reason to opt for this method was due to the lack of user data to identify the most optimum weights or to train a DL model.

The calculation of this bias draws concepts from Reinforcement learning techniques.

$$B_{w,p} = \frac{\left[\sum_{i=0}^{n_g} \frac{b_{p,s}}{(\alpha + n_m)}\right]}{N_{n_g}} \tag{6.4}$$

Equation for the calculation of the recommendations bias in combining outputs in ensembled models *(self-composed)*

$B_{w,p}$ - Default Bias weighting for a chosen pipeline that recommendations are given from

$b_{p,s}$ - Successful bias selection for a chosen pipeline for the last n days

α - Constant, set to 0.001 to avoid division by 0 error for today's bias selections

n_m - Number of days away from the current day.

n_g - Grouped days (Eg: 1 day, 7 days, 1 month, 3 months, 6 months, 1 year)

N_{n_g} - Total number of grouped days considered

Applying Bias Push

When presenting recommendations, the author decided to allow a system admin to be capable of suggesting a push towards a preferred direction to allow the bias to be altered.

$$B_{c,p} = b_{l,p} + \left(B_{w,p} - b_{a,p}\right) \tag{6.5}$$

Equation for the calculation of the recommendations bias in combining outputs in ensembled models *(self-composed)*

$B_{c,p}$ - Current bias of a chosen recommendations pipeline

$b_{l,p}$ - Last applied user bias for the chosen recommendations pipeline. This can be 0 or null

$B_{w,p}$ - Default bias of a chosen recommendations pipeline

$b_{a,p}$ - Admin suggested bias of a chosen recommendations pipeline

The above bias will be applied only to users who haven't chosen a preferred bias. It can be applied to users who have chosen the bias as well, but it is suggested to be applied after initially showing recommendations to the user using their requested bias.

6.4.4 UI Design

UI wireframes will be designed and added before implementing the UI of the MVP (Minimum Viable Product) that will be created over the following weeks. Since the core research component didn't require a UI, this design was not necessary for this submission.

6.4.5 System Process Flow Chart

The algorithm's flow and decision structures are depicted in the flowchart below. It explains a significant proportion of the system since the expected implementation is primarily procedural.

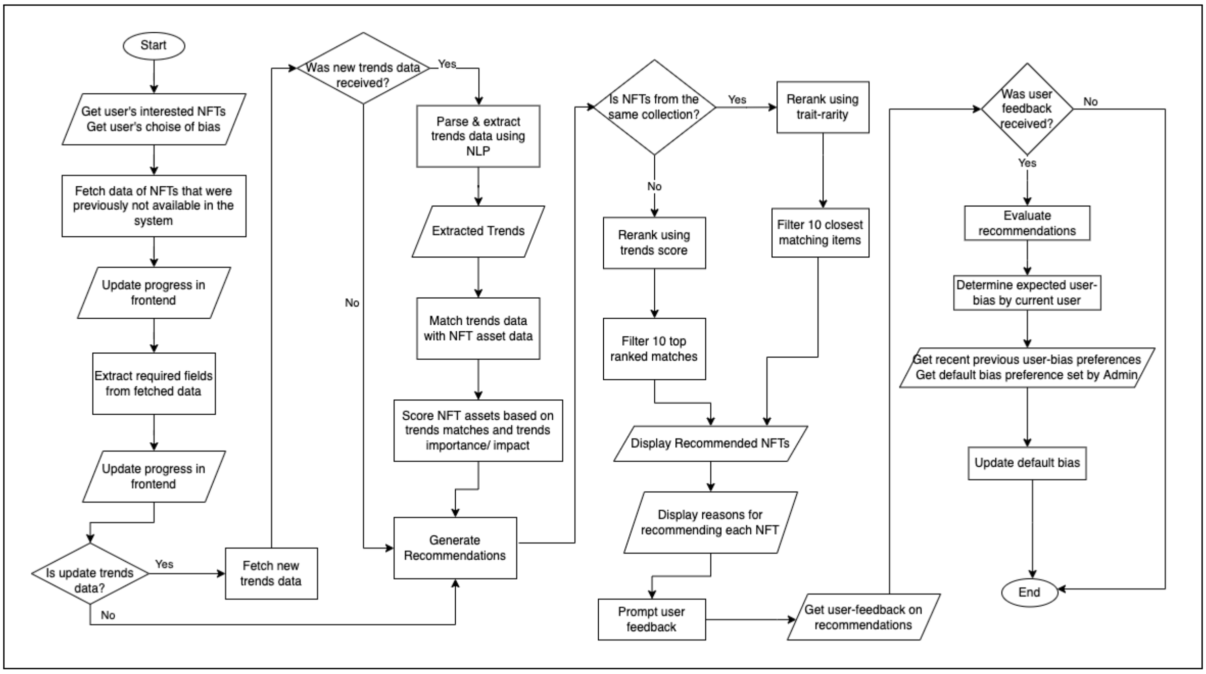


Figure 6.4: System Process Flow Chart*(self-composed)*

6.5 Chapter Summary

The design, architectural aspects, and the flow of the project and novel author-designed algorithms were documented in this chapter followed by the expected UI wireframes to be implemented for the end-users interaction with the system.

CHAPTER 7: IMPLEMENTATION

7.1 Chapter Overview

This chapter explains the core implementation of the research prototype together with the technologies, languages & supporting tools used for development of the prototype, with reasoning to the choice of each selection.

7.2 Technology Selection

7.2.1 Technology Stack

The technologies that were used to implement the prototype at each layer are shown below.

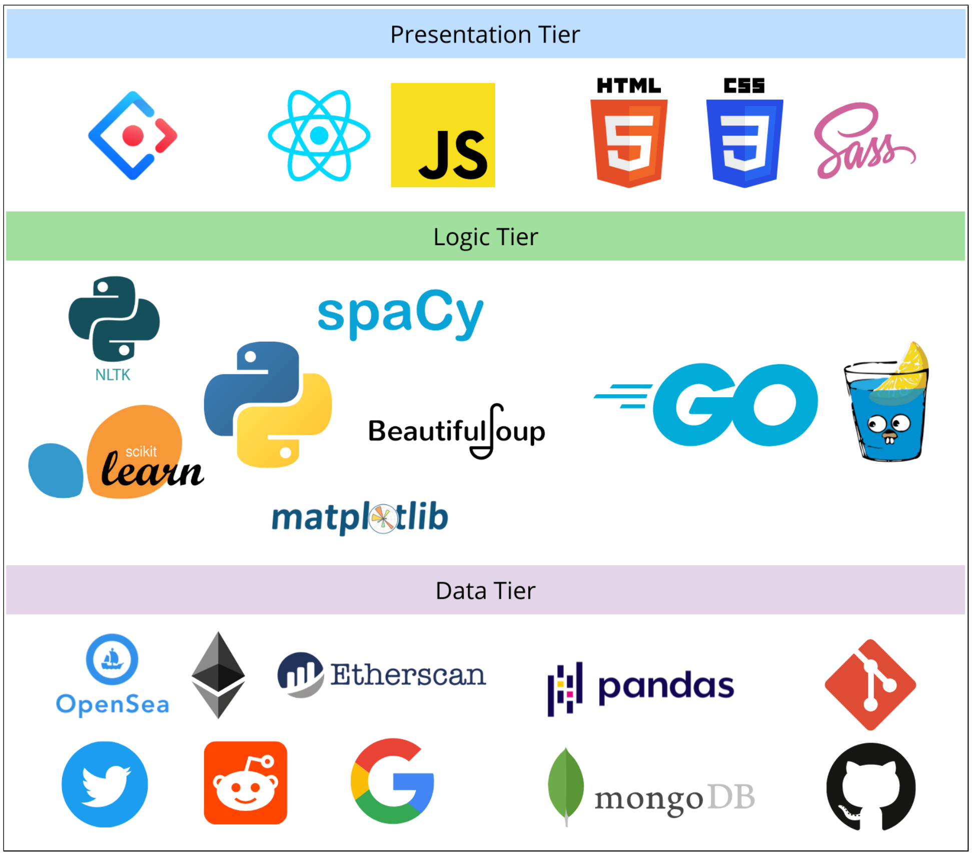


Figure 7.1: Technology Stack *(self-composed)*

Linux will be the default choice for development since of the ease of support for multiple development tools and performance benefits. MacOS/ Windows will be used for research documentation & study purposes.

The rest of the choices in the above tech-stack have been explained in the following sections.

7.2.2 Data Selection

Being a data science project at the core, it was important to choose the best possible sources of data to gather sufficient data for analysis & produce the best possible recommendations.

The data requirements identified were,

1. NFT asset data
2. Global trends data
3. NFT Smart Contract data
4. NFT events (sales) data
5. NFT bids data

Since the main technological research gap to be addressed was with the integration of global trends into content based recommendations, this was given a higher priority at first. These data requirements were sourced from the following sources and heavily pre-processed there after to create a usable dataset for data analysis.

- NFT asset, events, bids data - From the **OpenSea API**.
- Global trends data
  - Twitter data - From **Twitter developer API**.
  - Google Trends data - From Google Dataset Search & unofficial **Google Trends Python API (Pytrends)**.
- Ethereum Smart Contract data - From Etherscan & OpenSea

All the data-points that could be used for recommendations and explored with iterative development, as a research. This iterative process took a long time since the APIs were rate limited. The gathered pre-processed datasets will be made available for public use for future researches.

7.2.3 Selection of development framework

Table 7.1: Selection of development framework

Framework	**Justification for selection**
Gin Gonic	It's extremely convenient to build APIs using Gin with Golang. It also has an easily debuggable log output & claims smashing performance (up to 40 times faster!)

Ant Design	The world's second most popular React UI framework. Used in many industrial applications and has a wide range of components to match most UI requirements. Since it's tree-shaking compatible, it will build only the components that are used. This reduces build time of the frontend. The CSS is easily customizable as well.

Although this is a data science project, all data science models utilized were built from scratch without the use of libraries, since doing so allowed the author to tweak the models at will.

7.2.4 Programming language

Python is the language that will be used to create the ML models. Python is an all-purpose language that has been used in many projects involving data science. It has a vast collection of supporting libraries that eases many data science related tasks.

For the API proxy it was decided to use **Golang**, which is statically typed language that attempts to resemble the performance of C. Golang will allow the application to support concurrency and multi-threaded communications while being extremely lightweight and fast. This will be used to avoid any bottlenecks that could occur at this point in the system, while potentially bolstering performance.

For the frontend, **JavaScript** was decided to be used to show dynamic content and allow a highly interactible & inviting user experience.

7.2.5 Libraries Utilized

Table 7.2: Libraries Utilized with justification for choices

Library	**Justification for selection**
Pandas	Pandas dataframes allow a vast range of functionalities required for data analysis such as cleaning, transforming, filtering, sorting & manipulating of data
Scikit-learn	Used for vectorizing text and generate similarity matrices between items, for recommendations.
NLTK	Convenient to use for NLP data parsing, using the RAKE vectorizer.
SpaCy	Allows production-ready advanced NLP.
Beautiful Soup	Convenient to scrape data from the internet.
Matplotlib	Has almost any type of visualization method for data analysis.

React	A UI library that makes it easy to build interactive websites. Used as an alternative to using a framework since the vast array of capabilities and other integratable frameworks and libraries. It was important to develop an easily interactible frontend, since it will be the users' point of interaction with the system.

7.2.6 IDE's Utilized

Table 7.3: IDEs Utilized with justification for choices

IDE	Justification for selection
Google Colab	Convenience of trial & error of fetching data, building, testing ML models and ability to work across multiple devices with the cloud development environment.
VSCode	Extremely dynamic while being simple to use, yet powerful for front-end development with it's extensions & code snippets.
Golang	Convenient syntax highlighting & auto-completion for Golang development.
PyCharm	Well-equipped Python Integrated Development Environment (IDE) with a lot of capabilities.

7.2.7 Summary of Technology selection

Table 7.4: Summary of Technology selection

Component	Tools
Programming Languages	Python, Golang, JavaScript
Development Framework	Gin Gonic
UI Framework	Ant Design of React
Libraries	Pandas, Scikit-learn, NLTK, SpaCy, Beautiful Soup, Matplotlib, React
IDE – Research	Google Colab
IDE – Product	VSCode, Golang, Pycharm
Version Control	Git, GitHub
Application hosting	Netlify, AWS

7.3 Implementation of Core Functionalities

Since a Recommendations System's ultimate goal is to reduce the amount of information overload and provide the user with the best possible options, it was essential to build a dataset to suit the expected requirements. Just throwing in all the data fetched from APIs into a DL wouldn't give an expected successful recommendation. Therefore, the fetched data was heavily preprocessed.

NFT Data Mining

Continuously being able to add new NFTs or even adding an initial set of NFTs should be possible in the system for users' convenience. When doing so, we need to make sure that relevant information is extracted.

Figure 7.2: Implementation code segment: NFT data mining & preprocessing *(self-composed)*

The data extraction is done to extract information required for recommendations, to view details of items & to save information for recommendation algorithms/ predictions that are potentially possible in the future.

NLP Preprocessing, Vectorizing & Recommendations

```
[ ] # instantiating and generating the count matrix
    count = CountVectorizer()   # used to transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text
    count_matrix = count.fit_transform(df['All_key_words_str'])
```

Figure 7.3: Implementation code segment: Content Vectorizer *(self-composed)*

A Count Vectorizer was used from the *scikit learn* library to vectorize all words, to be used for similarity matching. The reason for choosing the Count Vectorizer over a Tf-Idf Vectorizer was because Tf-Idf will give lower scores to more common words found in the dataset. Since our intent is to identify all the possible matches and primarily rank the content based results using global trends, it made more sense to go with a Count Vectorizer.

A Cosine Similarity Matrix is then generated from the *scikit learn* library to identify all the matching words contained across all NFTs content. This generates the recommendation ahead of time.

```python
# generating the cosine similarity matrix
cosine_sim = cosine_similarity(count_matrix, count_matrix)
```

Figure 7.4: Implementation code segment: Generating the Cosine Similarity Matrix *(self-composed)*

Figure 7.5: Implementation code segment: Produce Trait Rarity Based Recommendations *(self-composed)*

The above recommendation generation algorithms were created to cater towards matching NFTs within a collection, since most of the major NFT-collections have comparatively more unique data in traits compared to descriptions. Trait rarity similarity was identified to be the best way to identify total uniqueness which represents the value of each NFT. Although the calculation of total rarity was explored by *rarity tools* during the course of the research (rarity.tools, 2021a; rarity.tools, 2021b), recommending similar total rarities is a novel implementation in the application domain.

Trends Extraction, Preprocessing & Recommendations

The above code segment preprcoesses trends that are fetched from the live Twitter API.

The above code segment assigns a tweet volume for trends with no volume & calculates the median Tweet volume which used to calculate the impact score of each trend.

The below code segment is used to calculate the trends score for each NFT and finally make trends-based recommendations.

7.4 User Interface

The UI wireframes depicting the planned UI for the MVP (Minimum Viable Product) have been place in **Appendix C - UI Wireframes**.

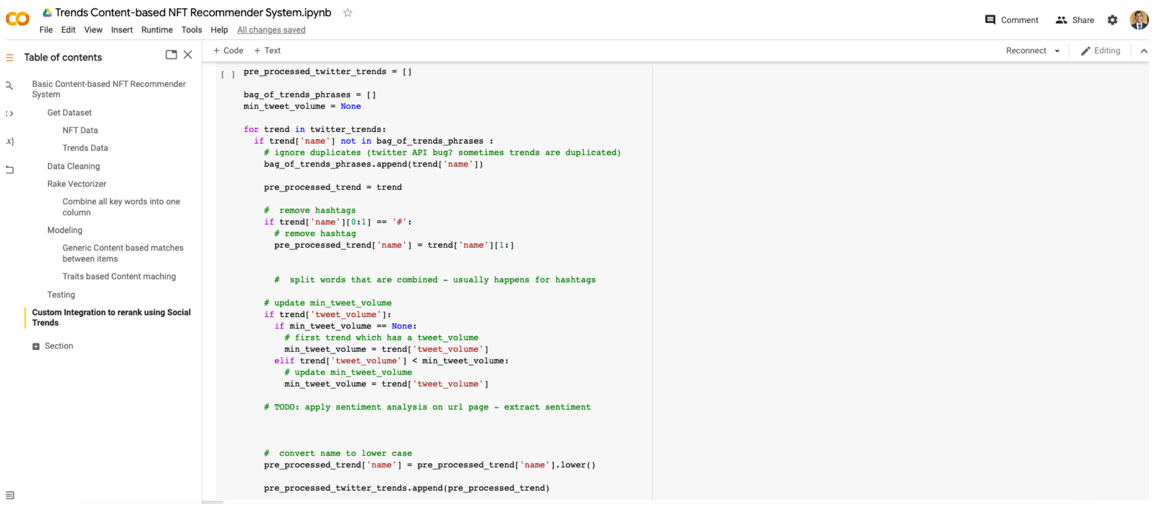


Figure 7.6: Implementation code segment: Preprocess Trends Data *(self-composed)*

Figure 7.7: Implementation code segment: Calculating Trends Score *(self-composed)*

Figure 7.8: Implementation code segment: Calculating Trends Score *(self-composed)*

7.5 Chapter Summary

The chapter comprised of the technologies, languages & supporting tools utilized to implement the prototype developed as part of the research. Discussions accompany the code snippets and algorithms produced as part of core functionality.

CHAPTER 8: TESTING

8.1 Chapter Overview

This chapter discusses how testing was carried out to ensure that functions flowed as expected. It will cover testing objectives and procedures such as model testing, benchmarking, functional testing, non-functional testing, module, and integration testing.

8.2 Objectives and Goals of Testing

The primary goal of software testing is to ensure that the system is performing as expected based on the requirements acquired.

These expectations can be broken down as follows

- Ensure that all models of the system are working as intended and that they have been tested in order to achieve the desired optimum results.
- Ensure that the system meets the MoSCoW technique's mandatory "Must have" and "Should have" functional requirements.
- Apply possible benchmarking techniques that can be used to benchmark the developed system for future work.
- Identify if the required & important non-functional requirements have been satisfied.
- Identify possible points of improvements/ bug fixes that can be applied to the system.

8.3 Testing Criteria

With the goal of narrowing the gap between the intended and implemented systems, a criterion to test the system in two ways was defined. The following are the two methods for testing:

1. Functional Quality - This focuses on the system's development characteristics and technical requirements in order to see how well it meets the specified design based on functional requirements.
2. Structural Quality - This tests the system's non-functional requirements while ensuring that it meets the functional requirements' performance criteria.

8.4 Model Testing & Evaluation

8.4.1 Model Testing

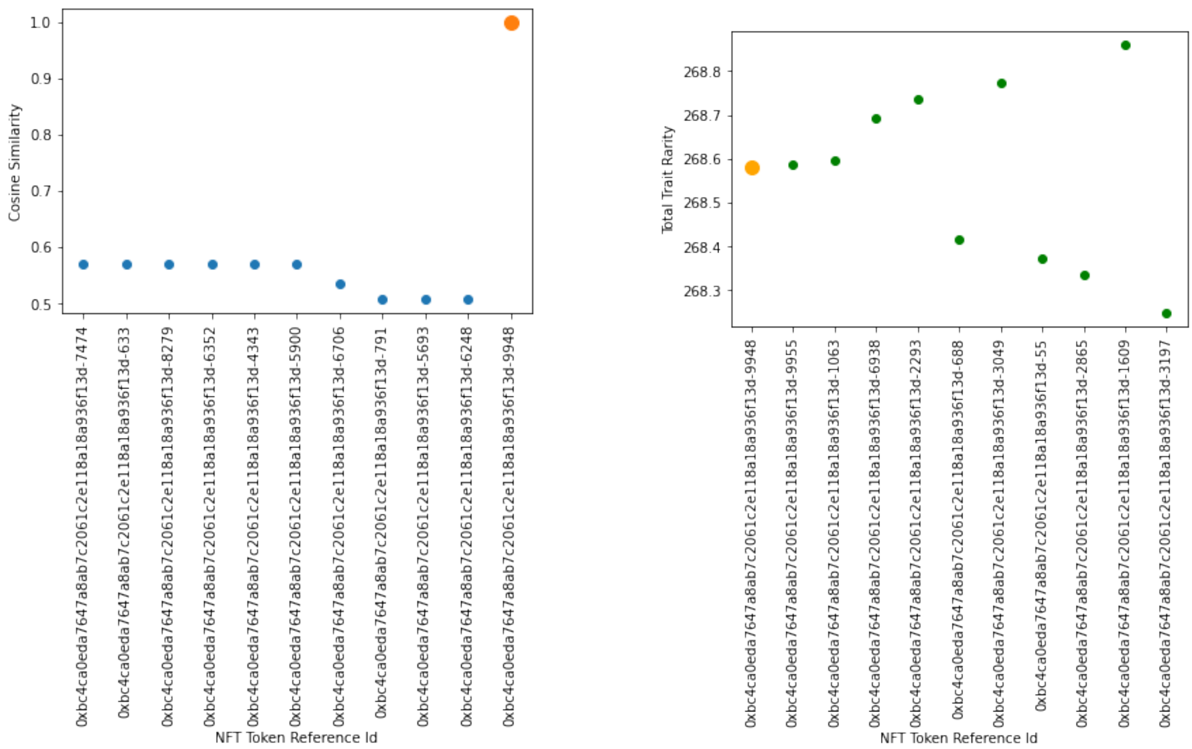
The multiple ensemble models developed in the project were tested based on the following conditions.

Table 8.1: Testing Trait Content & Trait Rarity based recommendations

Model	**Testing Method**
Trait Rarity based RecSys	The total rarity being as close as possible to the reference item's rarity
Trait Content based RecSys	Cosine Similarity being the closest to the reference item's traits
Trends based RecSys	Check if the items that get the highest trend score will be recommended and identify if the most trending, accepted & timely items are obtained using the available data.

Trait Rarity & Content based Recommendations Systems

Following graphical analysis of the test output shows the validity of the outputs produced by the trait rarity & content based models.



(a) Trait Content based recommendations (b) Total Rarity based recommendations

Figure 8.1: Outputs produced by Trait Content & Rarity Recommendation Systems *(self-composed)*

Trends based Recommendations System

The following heatmap shows how the trend-score for items decrease with time, from the date of matching with the trend. Additional heatmaps with detailed outputs have been placed under *Appendix D - Testing*.

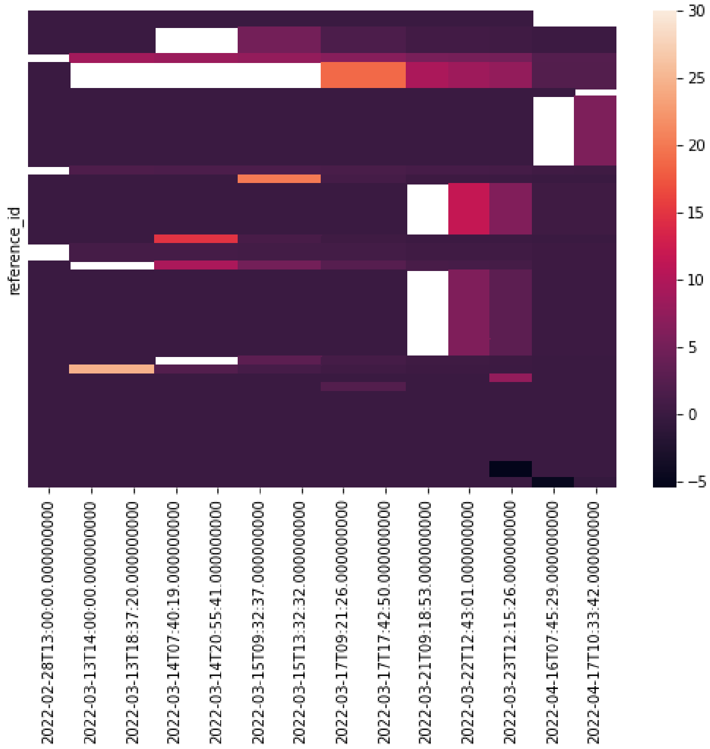


Figure 8.2: Trends based Recommender - Trend Score Heatmap *(self-composed)*

8.4.2 Model Evaluation

Trait Rarity & Content based Recommendations Systems

The NFT trait rarity and trait content based Recommendation Systems were matched against each other to demonstrate the difference in recommendations produced by each other even though they are both generated based on the traits and overall repetition of each of these traits across a collection.

Table 8.2: Evaluating Trait Content & Trait Rarity based recommendations

Testing Method	**precision@k**	**recall@k**	**f1_score@k**
self-scored	1.0	1.0	1.0
combined-scored	1.0	0.5	0.67

The above precision & recall @k are customized precision & recalls created for the purpose of testing & evaluating Recommendations Systems.

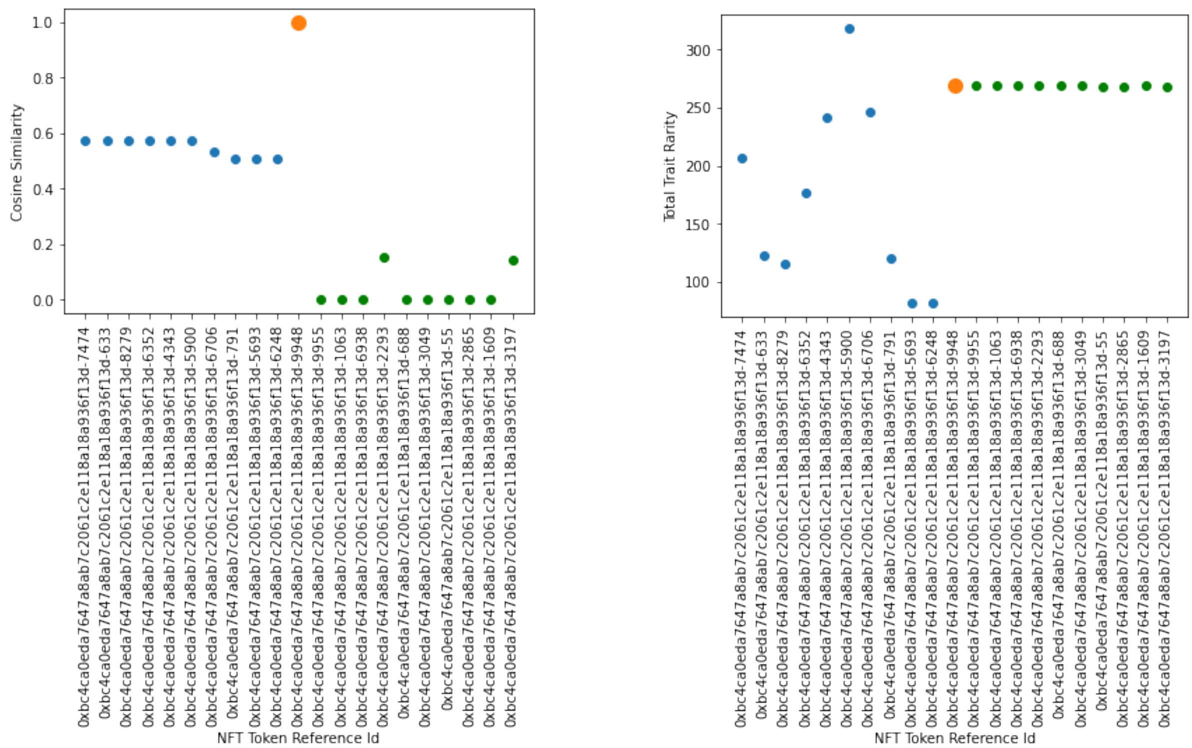
These were calculated using the below formulae.

$$\text{Recommender System Precision} = \frac{\text{no. of recommendations that are relevant}}{\text{no. of items that we recommended}}$$

$$\text{Recommender System Recall} = \frac{\text{no. of recommendations that are relevant}}{\text{no. of all the possible relevant items}}$$

The formula for f1 score is the same, except that the above altered precision & recall were used.

The reason that both the models were self-scored & combined-scored was to demonstrate that although they produce the best possible results by themselves, using only one of the models won't give all the possible results. This can be further explained with the aggregate diversity graphs displayed in the **Benchmarking** section.



(a) Cosine similarities of content (b) Total rarities

Figure 8.3: Comparison Recommendations generated by both models *(self-composed)*

The above graphs establishes the necessity of generating recommendations from both models.

The items marked in blue were recommended by the Trait content based RecSys, the green ones by the Trait rarity based RecSys and the orange item was the reference item used to generate recommendations. It is clear that although both the trait content and rarity based Recommendation Systems are generating recommendations using traits of items, they produce very different outputs, especially in the case of rarity based recommendations.

Trends based Recommendations System

The trends-based recommender that is expected to enhance content-based Recommendation Systems with Collaborative-filtering-like capabilities without collecting user click-data was tested to identify if the trends based recommendations were possible to be generated.

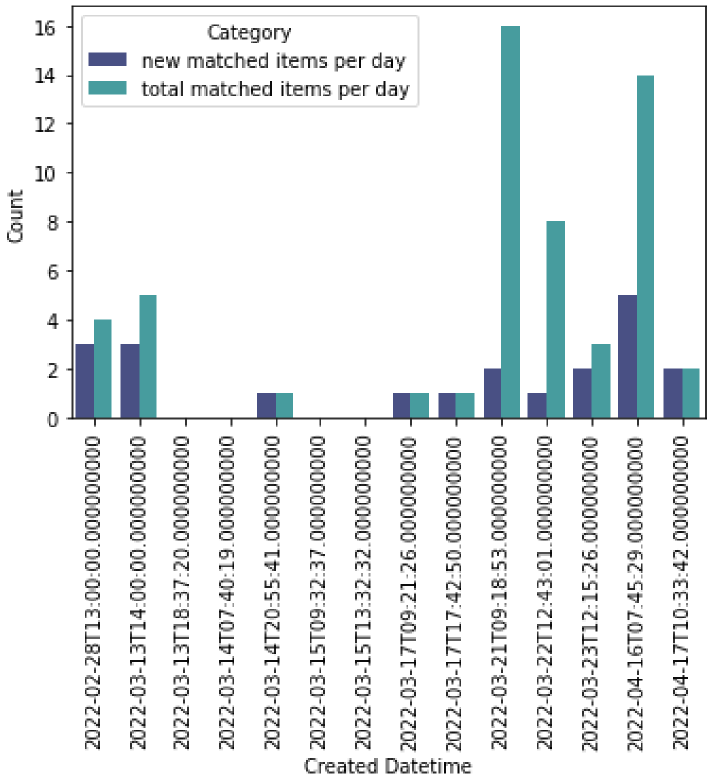


Figure 8.4: Evaluation of Trends based Recommender *(self-composed)*

8.5 Benchmarking

The following graph shows the aggregate diversity of items generated by both the models by using the entire *Bored Ape Yatch Club* NFT collection of 10,000 NFTs.

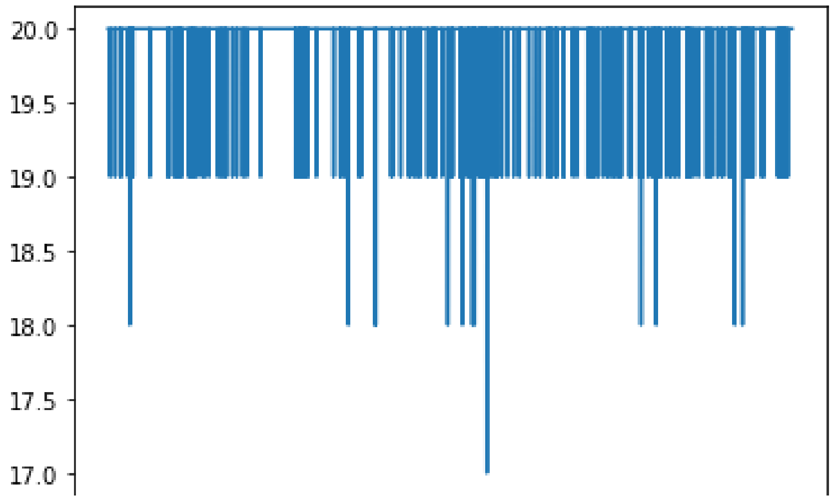


Figure 8.5: Aggregate diversity of generated Recommendations *(self-composed)*

As displayed, the maximum overlapping of items was 3, while most recommendations had 1 or less than one similarities.

8.6 Functional Testing

The application was functionally tested against the Functional Requirements (FR) specified during the requirements gathering phase.

Testing results of Functional Requirements in **Appendix D - Testing** shows the breakdown of functional tests that were carried out.

8.7 Module & Integration Testing

Module	**Input**	**Expected Output**	**Actual Output**	**Status**
Trends Data-fetcher	New raw Trends data	Extract required data and save	Extract required data and save	Passed
NFT Data-fetcher	NFT asset data	Filters out required information and calculates rarity score	Filtered out required information and calculates rarity score	Passed
Trends scorer	Trends with sentiment score, volume and trend datetime	Trend score	Trend score	Passed
Sentiment Analyzer	Top tweets of each trend	Sentiment score sentiment polarity	Sentiment score sentiment polarity	Passed

8.8 Non-functional Testing

8.8.1 Important Non-functional Requirement Completion Percentage

8.9 Limitations of Testing Process

As an initial study of recommending NFTs, it was difficult to pin-point on the ground truth of what exactly should've been produced by the models as recommendations.

With the trends based recommender, the unavailability of past trends data restricted extensive testing. The unavailability of an open e-commerce dataset restricted from benchmarking the models. The lack of data was the biggest constraint in testing, evaluating & benchmarking this project.

8.10 Chapter Summary

This chapter covered extended testing, evaluation & benchmarking of the core-research component. Furthermore functional, integration and non-functional tests were carried out & the results were recorded. Any limitations of the process was explained at the end.

CHAPTER 9: EVALUATION

9.1 Chapter Overview

After the designed prototype had been successfully implemented and was optimized to achieve the best performance through a large number of testing combinations, the system was evaluated with respective to the requirements gathered in the SRS chapter. This chapter is dedicated to the project's evaluation, which will involve self-evaluation as well as assessments from technical, domain and industry experts.

9.2 Evaluation Methodology & Approach

Since the research project consists of models that can be quantitatively evaluated and another model that presents a more qualitative output, both qualitative & quantitative evaluation approaches were taken. Based on the tests carried out in the testing chapter, the research outcome given by the prototype was evaluated using evaluation techniques of Recommendation Systems extracted from literature. In this chapter, a thematic analysis will be used to present the feedback received from experts.

The link to the demonstration video of the research, that was used for evaluations can be found here: `https://youtu.be/fjRzZXUOrRo`

9.3 Evaluation Criteria

The following criteria were used for the thematic analysis that surfaced in interviews with experts & other aspects of research that needed to be assessed to determine the value of the research that was conducted.

Table 9.1: Evaluation Criteria

Criterion	**Evaluation Purpose**
Choice of research undertaken	To validate the significance of the choices of topic, domain, research gap, depth undertaken in this research.
Research contribution	To determine the value of the contributions produced to the technical field of Recommendation Systems, the domain of NFTs or Blockchain and any other additional research-oriented contributions made.
Quality of research documentation	To confirm that an adequate amount of literature has been reviewed and the entire research process has been documented & presented in an acceptable quality.

Development approach	To confirm that an appropriate development approach had been taken to solve the problem at hand to the best possible extent, with the implementation of the prototype.
Quantitative analysis of results	To validate the metrics used to evaluate & analyze the results produced by the research.
Usability, UI/ UX of MVP	To verify that the product developed for demonstration is convenient for end-users.

9.4 Self-Evaluation

The following self-evaluation was done by the author of the research according to the above mentioned evaluation criteria.

Table 9.2: Self-evaluation of the author according to the Evaluation Criteria

Criterion	**Author's Self-evaluation**
Choice of research undertaken	The research area chosen revolved around a highly useful technical application as well as a very new & popular domain that is expected to be used in many applications in the future.
Research contribution	The contributions of this research lie across a broad spectrum. Firstly, the technical contributions made in Recommendation Systems can be identified by the novel recommendations method introduced with the use of a custom algorithm to recommend trending items based on social media trends. Secondly, the contribution towards the domain is novel & has opened new pathways to possible future work.
Quality of research documentation	The quality of the documentation is of the highest possible standard. The use of Latex for all the research documentation including the thesis signifies this, together with the quality of the diagrams & content as well as the research papers written.
Development approach	A significant effort has been put into data collection & pre-processing to give the best possible results with a very meagre amount of data. Cutting edge-languages & tools have also been used in the process.

Quantitative analysis of results	Even though the quantitative analysis & evaluation of the results produced by the system is difficult to be measured, jupyter notebooks have been used to demonstrate these using graphical outputs in a comprehensible format.
Usability, UI/ UX of MVP	The UI/UX of the final product has been developed in a very usable & attractive manner.

9.5 Selection of Evaluators

The selection categories of evaluators for the project can be broken down into the following 3 categories.

Table 9.3: Categorization of selected evaluators

CAT ID	**Category**
1	Experts with research experience in the fields of Recommendation Systems, Data Science, Data Engineering & Machine Learning.
2	Experts with domain expertise in the fields of Blockchain, NFTs & economics.
3	Possible end-users of the applications such as NFT creators, collectors & enthusiasts.

9.6 Evaluation Results & Expert Opinions

9.6.1 Qualitative Analysis

The expert opinions that were received have been analysed according to emerged themes below.

Table 9.4: Thematic analysis of expert evaluation feedback

Criterion	CAT ID	Theme	Summary of Opinions
Choice of research undertaken	1, 2	Recommendation Systems choice gap	The study of Recommendation Systems is valuable impactful, especially in an unexplored e-commerce domain that has had high trading volumes.
	2	Domain research gap	The domain is new. There's a clear research gap identified to be fulfilled.
	3	Domain research applicability for use	The domain application is new required since it's difficult to explore items.
Research contribution	1	Technical Contribution towards Recommendation Systems	Innovative methods of solving the research gap have been identified.
	2	Domain Contribution	The contribution is good because there's no other system like this and gathering data is difficult.
	3		
Quality of research documentation	1, 2	Content presentation of content	The use of latex was immediately noticed and commended.
Development approach	1	Data preprocessing	A great amount of data pre-processing has been done, as it should be for a Recommendations System to produce optimum results.
	2	Requirement of a separate application	Since the prototype produces clear results, a separate application is not required.
Quantitative analysis of results	1	Analysis of the social trends based RecSys	Could try to scrape the internet/ OpenSea to validate generated recommendations from the social trends RecSys.
	2		Could try to synthesize demonstrate results from the social trends RecSys to show why it's needed.

	1, 2	Analysis of trait-based RecSys	The graphical analysis of the trait-based recommendations models is very clear.
Usability, UI/ UX of MVP	3		

9.7 Limitations of Evaluation

As discussed in the literature review, it is very difficult to evaluate a Recommendations System, especially one that is specific to a particular use-case. Therefore, the testing & evaluation equations had to be adjusted to suit these. The social trends based Recommendations System was the most difficult to evaluate due to the lack of data that was available to the author. Since the domain is very new, there were very few people who could understand the impact of the domain contribution & why some choices had to be made.

Long hours of power-cuts throughout the evaluation phase of the project made it very difficult to setup meetings with evaluators and work on evaluation aspects of the research.

9.8 Evaluation of Functional Requirements

The breakdown of the evaluation of functional requirements can be found in the table Evaluation of the implementation of Functional Requirements of **Appendix E - Evaluations**

9.9 Evaluation of Non-functional Requirements

The breakdown of the evaluation of non-functional requirements can be found in the table Evaluation of the implementation of Non-functional requirements of **Appendix E - Evaluations**

9.10 Chapter Summary

This chapter covered the evaluation aspects of the research that was conducted. The approaches taken for evaluation were discussed with reasoning of choosing each method. The criteria for evaluation was defined prior to the author's self-evalution & feedback from evaluators. The opinions received form evaluators were broken down into themes and presented based on the pre-defined criterion. Finally the functional & non-functional requirements were evaluated.

CHAPTER 10: CONCLUSION

REFERENCES

Adil Moujahid (June 28, 2021). *Data mining Meebits NFTs using Python and OpenSea API // Adil Moujahid // Data Analytics and more*. URL: http://adilmoujahid.com/posts/2021/06/data-mining-meebits-nfts-python-opensea/ (visited on 10/09/2021).

Ayushi, Smriti and Badri Prasad (Nov. 8, 2018). "Cross-Domain Recommendation Model based on Hybrid Approach". In: *International Journal of Modern Education and Computer Science* 10.11, pp. 36–42. ISSN: 20750161, 2075017X. DOI: 10.5815/ijmecs.2018.11.05. URL: http://www.mecs-press.org/ijmecs/ijmecs-v10-n11/v10n11-5.html (visited on 07/12/2021).

Bartolucci, Silvia and Andrei Kirilenko (Aug. 12, 2020). "A model of the optimal selection of crypto assets". In: *Royal Society Open Science* 7.8. Publisher: Royal Society, p. 191863. DOI: 10.1098/rsos.191863. URL: https://royalsocietypublishing.org/doi/full/10.1098/rsos.191863 (visited on 07/07/2021).

Beedham, Matthew (Dec. 10, 2019). *Nike now holds patent for blockchain-based sneakers called 'CryptoKicks'*. TNW | Hardfork. Section: hardfork. URL: https://thenextweb.com/news/nike-blockchain-sneakers-cryptokick-patent (visited on 07/19/2021).

Casey Newton (July 22, 2021). *Mark Zuckerberg is betting Facebook's future on the metaverse - The Verge*. The Verge. URL: https://www.theverge.com/22588022/mark-zuckerberg-facebook-ceo-metaverse-interview (visited on 11/20/2021).

Chen, Rung-Ching and Hendry (June 2019). "User Rating Classification via Deep Belief Network Learning and Sentiment Analysis". In: *IEEE Transactions on Computational Social Systems* 6.3. Conference Name: IEEE Transactions on Computational Social Systems, pp. 535–546. ISSN: 2329-924X. DOI: 10.1109/TCSS.2019.2915543. (Visited on 05/25/2021).

Cheng, Li Chen and Ming-Chan Lin (Oct. 2020). "A hybrid recommender system for the mining of consumer preferences from their reviews". In: *Journal of Information Science* 46.5, pp. 664–682. ISSN: 0165-5515, 1741-6485. DOI: 10.1177/0165551519849510. URL:

http://journals.sagepub.com/doi/10.1177/0165551519849510 (visited on 07/16/2021).

Chevet, Sylve (2018). "Blockchain Technology and Non-Fungible Tokens: Reshaping Value Chains in Creative Industries". In: *SSRN Electronic Journal*. ISSN: 1556-5068. DOI: 10.2139/ssrn.3212662. URL: https://www.ssrn.com/abstract=3212662 (visited on 04/18/2021).

Choi, Minjin et al. (Mar. 8, 2021). "Local Collaborative Autoencoders". In: *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. WSDM '21. New York, NY, USA: Association for Computing Machinery, pp. 734–742. ISBN: 978-1-4503-8297-7. DOI: 10.1145/3437963.3441808. URL: https://doi.org/10.1145/3437963.3441808 (visited on 05/18/2021).

Clark, Mitchell (Mar. 3, 2021). *People are spending millions on NFTs. What? Why?* The Verge. URL: https://www.theverge.com/22310188/nft-explainer-what-is-blockchain-crypto-art-faq (visited on 07/19/2021).

community (Nov. 18, 2021). *Solana Ecosystem*. Solana. URL: https://solana.com/ecosystem/# (visited on 11/20/2021).

Conti, Robyn (Apr. 29, 2021). *What You Need To Know About Non-Fungible Tokens (NFTs)*. Forbes Advisor. Section: Investing. URL: https://www.forbes.com/advisor/investing/nft-non-fungible-token/ (visited on 07/19/2021).

CryptoKitties (2021). *CryptoKitties | Collect and breed digital cats!* CryptoKitties. URL: https://www.cryptokitties.co (visited on 11/20/2021).

Dacrema, Maurizio Ferrari, Simone Boglio, et al. (Jan. 6, 2021). "A Troubling Analysis of Reproducibility and Progress in Recommender Systems Research". In: *ACM Transactions on Information Systems* 39.2, 20:1–20:49. ISSN: 1046-8188. DOI: 10.1145/3434185. URL: https://doi.org/10.1145/3434185 (visited on 11/14/2021).

Dacrema, Maurizio Ferrari, Paolo Cremonesi, and Dietmar Jannach (Sept. 10, 2019). "Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches". In: *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 101–109. DOI: 10.1145/3298689.3347058. arXiv: 1907.06902. URL: http://arxiv.org/abs/1907.06902 (visited on 05/30/2021).

Dacrema, Maurizio Ferrari, Paolo Cremonesi, and Dietmar Jannach (July 9, 2020). “Methodological Issues in Recommender Systems Research (Extended Abstract)”. In: Twenty-Ninth International Joint Conference on Artificial Intelligence. Vol. 5. ISSN: 1045-0823, pp. 4706–4710. DOI: 10.24963/ijcai.2020/650. URL: https://www.ijcai.org/proceedings/2020/650 (visited on 11/14/2021).

Dafflon, Jacques, Jordi Baylina, and Thomas Shababi (Nov. 20, 2017). *EIP-777: Token Standard*. Ethereum Improvement Proposals. URL: https://eips.ethereum.org/EIPS/eip-777 (visited on 11/19/2021).

Dayan, Aviram et al. (2011). “Recommenders benchmark framework”. In: *Proceedings of the fifth ACM conference on Recommender systems - RecSys ’11*. the fifth ACM conference. Chicago, Illinois, USA: ACM Press, p. 353. ISBN: 978-1-4503-0683-6. DOI: 10.1145/2043932.2044003. URL: http://dl.acm.org/citation.cfm?doid=2043932.2044003 (visited on 10/16/2021).

Decentralized applications (dapps) (Sept. 1, 2021). ethereum.org. URL: https://ethereum.org (visited on 09/01/2021).

dfinzer (July 20, 2021). *Announcing our $100M raise, led by a16z*. OpenSea Blog. Section: Company Announcements. URL: https://opensea.io/blog/announcements/announcing-our-100m-raise-led-by-a16z/ (visited on 07/23/2021).

Dowling, Michael (Apr. 29, 2021a). “Fertile LAND: Pricing non-fungible tokens”. In: *Finance Research Letters*, p. 102096. ISSN: 1544-6123. DOI: 10.1016/j.frl.2021.102096. URL: https://www.sciencedirect.com/science/article/pii/S154461232100177X (visited on 07/17/2021).

— (Apr. 29, 2021b). *Is non-fungible token pricing driven by cryptocurrencies? | Elsevier Enhanced Reader*. DOI: 10.1016/j.frl.2021.102097. URL: https://www.sciencedirect.com/science/article/pii/S1544612321001781?via%3Dihub (visited on 06/23/2021).

EIP-2309 (2021). *EIP-2309: ERC-721 Consecutive Transfer Extension*. Ethereum Improvement Proposals. URL: https://eips.ethereum.org/EIPS/eip-2309 (visited on 07/20/2021).

ERC (2021). Ethereum Improvement Proposals. URL: https://eips.ethereum.org/erc (visited on 08/24/2021).

ERC-20 Token Standard (2021). ethereum.org. URL: https://ethereum.org (visited on 08/31/2021).

ERC-721 Non-Fungible Token Standard (2021). ethereum.org. URL: https://ethereum.org (visited on 07/19/2021).

Fairfield, Joshua (Apr. 6, 2021). *Tokenized: The Law of Non-Fungible Tokens and Unique Digital Property*. SSRN Scholarly Paper ID 3821102. Rochester, NY: Social Science Research Network. URL: https://papers.ssrn.com/abstract=3821102 (visited on 04/18/2021).

Ferdiansyah, Ferdiansyah et al. (Oct. 2019). "A LSTM-Method for Bitcoin Price Prediction: A Case Study Yahoo Finance Stock Market". In: *2019 International Conference on Electrical Engineering and Computer Science (ICECOS)*. 2019 International Conference on Electrical Engineering and Computer Science (ICECOS), pp. 206–210. DOI: 10.1109/ICECOS47637.2019.8984499. (Visited on 07/15/2021).

Ferrari Dacrema, Maurizio et al. (Oct. 19, 2020). "Critically Examining the Claimed Value of Convolutions over User-Item Embedding Maps for Recommender Systems". In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. CIKM '20. New York, NY, USA: Association for Computing Machinery, pp. 355–363. ISBN: 978-1-4503-6859-9. DOI: 10.1145/3340531.3411901. URL: https://doi.org/10.1145/3340531.3411901 (visited on 11/14/2021).

Frankenfield, Jake (2021). *Decentralized Applications – dApps*. Investopedia. URL: https://www.investopedia.com/terms/d/decentralized-applications-dapps.asp (visited on 09/01/2021).

Fred Rezeau et al. (Apr. 15, 2021). *NFTs on Stellar - Stellar Development Foundation*. Stellar Events. URL: https://stellar.org/events/nfts-on-stellar?locale=en,

%20https://stellar.org/events/nfts-on-stellar?locale=en (visited on 11/20/2021).

Freudenthaler, C., L. Schmidt-Thieme, and Steffen Rendle (2011). *Factorization Machines Factorized Polynomial Regression Models*. URL: https://www.semanticscholar.org/paper/Factorization-Machines-Factorized-Polynomial-Models-Freudenthaler-Schmidt-Thieme/10e8617b6c58d599b8d17b3a60047a599bfe5e72 (visited on 08/27/2021).

Geetha, G. et al. (Apr. 2018). "A Hybrid Approach using Collaborative filtering and Content based Filtering for Recommender System". In: *Journal of Physics: Conference Series* 1000. Publisher: IOP Publishing, p. 012101. ISSN: 1742-6596. DOI: 10.1088/1742-6596/1000/1/012101. URL: https://doi.org/10.1088/1742-6596/1000/1/012101 (visited on 04/29/2021).

Gwyneth Iredale (June 29, 2021). *Are Blockchain and AI a Perfect Combination?* 101 Blockchains. URL: https://101blockchains.com/blockchain-and-ai/ (visited on 07/23/2021).

Hackett, Robert (Mar. 18, 2021). *This crypto marketplace just raised $23 million to be the 'Amazon of NFTs'*. Fortune. URL: https://fortune.com/2021/03/18/nft-art-crypto-marketplace-opensea-amazon/ (visited on 07/15/2021).

Hu, Shigang et al. (2020). "Reviewer Credibility and Sentiment Analysis Based User Profile Modelling for Online Product Recommendation". In: *IEEE Access* 8. Conference Name: IEEE Access, pp. 26172–26189. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2971087.

Introduction to smart contracts (2021). ethereum.org. URL: https://ethereum.org (visited on 08/31/2021).

Larry, Hardesty (Nov. 22, 2019). *The history of Amazon's recommendation algorithm*. Amazon Science. Section: Latest news. URL: https://www.amazon.science/the-history-of-amazons-recommendation-algorithm (visited on 05/25/2021).

Larva Labs (2021). *The Meebits*. URL: https://meebits.larvalabs.com/ (visited on 11/20/2021).

Linden, G., B. Smith, and J. York (Jan. 2003). "Amazon.com recommendations: item-to-item collaborative filtering". In: *IEEE Internet Computing* 7.1, pp. 76–80. ISSN: 1089-7801. DOI:

10.1109/MIC.2003.1167344. URL: http://ieeexplore.ieee.org/document/1167344/ (visited on 05/18/2021).

Matney, Lucas (July 20, 2021). *NFT market OpenSea hits $1.5 billion valuation*. TechCrunch. URL: https://social.techcrunch.com/2021/07/20/nft-market-opensea-hits-1-5-billion-valuation/ (visited on 07/23/2021).

Naumov, Maxim et al. (May 31, 2019). "Deep Learning Recommendation Model for Personalization and Recommendation Systems". In: *arXiv:1906.00091 [cs]*. arXiv: 1906.00091. URL: http://arxiv.org/abs/1906.00091 (visited on 04/29/2021).

Non-fungible tokens (NFT) (2021). ethereum.org. URL: https://ethereum.org (visited on 11/20/2021).

Off the chain (July 5, 2021). *Off the chain: NFT market surges to $2.5B so far this year*. Aljazeera. URL: https://www.aljazeera.com/economy/2021/7/5/off-the-chain-nft-market-surges-to-2-5b-so-far-in-2021 (visited on 07/15/2021).

Peter Allen Clark (Nov. 15, 2021). *What is the Metaverse? Here's Why It Matters | Time*. Time. URL: https://time.com/6116826/what-is-the-metaverse/ (visited on 11/20/2021).

Prathap, Madana (Oct. 22, 2021). *Semi-fungible tokens — coins that travel between the worlds of cryptocurrency and NFTs*. Business Insider. URL: https://www.businessinsider.in/investment/news/what-is-a-semi-fungible-token-sft/articleshow/87205941.cms (visited on 11/19/2021).

rarity.tools (Apr. 28, 2021a). *Introducing rarity.tools. rarity.tools is a website dedicated to... | by rarity.tools | Medium*. URL: https://raritytools.medium.com/introducing-rarity-tools-9b0138e992b3 (visited on 04/28/2022).

— (Oct. 31, 2021b). *Ranking Rarity: Understanding Rarity Calculation Methods*. Medium. URL: https://raritytools.medium.com/ranking-rarity-understanding-rarity-calculation-methods-86ceaeb9b98c (visited on 04/28/2022).

Recommendations (2021). *Recommendations: What and Why? | Recommendation Systems*. URL: https://developers.google.com/machine-learning/recommendation/overview (visited on 08/24/2021).

Rendle, Steffen et al. (Sept. 22, 2020). "Neural Collaborative Filtering vs. Matrix Factorization Revisited". In: *Fourteenth ACM Conference on Recommender Systems*. RecSys '20. New York, NY, USA: Association for Computing Machinery, pp. 240–248. ISBN: 978-1-4503-7583-2. DOI: 10.1145/3383313.3412488. URL: https://doi.org/10.1145/3383313.3412488 (visited on 07/20/2021).

Saunders, Mark, Philip Lewis, and Adrian Thornhill (2003). "Research methods for business students". In: *Essex: Prentice Hall: Financial Times*.

Smith, Brent and Greg Linden (May 2017). "Two Decades of Recommender Systems at Amazon.com". In: *IEEE Internet Computing* 21.3. Conference Name: IEEE Internet Computing, pp. 12–18. ISSN: 1941-0131. DOI: 10.1109/MIC.2017.72.

Staff, Daily Hodl (Nov. 4, 2021). *Solana Becomes Fourth-Biggest NFT Blockchain As SOL Overtakes Cardano*. The Daily Hodl. URL: https://dailyhodl.com/2021/11/04/solana-becomes-fourth-biggest-nft-blockchain-as-sol-overtakes-cardano/ (visited on 11/20/2021).

Vanderbilt, Tom (2021). "The Science Behind the Netflix Algorithms That Decide What You'll Watch Next". In: *Wired* (). Section: tags. ISSN: 1059-1028. URL: https://www.wired.com/2013/08/qq-netflix-algorithm/ (visited on 08/27/2021).

We are open-sourcing a state-of-the-art deep learning recommendation model to help AI researchers and the systems and hardware community develop new, more efficient ways to work with categorical data. (Feb. 7, 2019). URL: https://ai.facebook.com/blog/dlrm-an-advanced-open-source-deep-learning-recommendation-model/ (visited on 05/15/2021).

What are you missing? Using basic machine learning to predict and recommend NFTs with OpenSea data - OpenSea Blog (Jan. 30, 2020). URL: https://opensea.io/blog/analysis/predict-and-recommend-nfts/ (visited on 08/27/2021).

Wolf, Thomas et al. (Oct. 2020). "Transformers: State-of-the-Art Natural Language Processing". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6. URL: https://aclanthology.org/2020.emnlp-demos.6 (visited on 11/19/2021).

Wood, Dr Gavin (2014). "ETHEREUM: A SECURE DECENTRALISED GENERALISED TRANSACTION LEDGER". In: p. 39. (Visited on 07/15/2021).

Zhang, Wenping, Mengna Xu, and Qiqi Jiang (2018). "Opinion Mining and Sentiment Analysis in Social Media: Challenges and Applications". In: *HCI in Business, Government, and Organizations*. Ed. by Fiona Fui-Hoon Nah and Bo Sophia Xiao. Vol. 10923. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 536–548. ISBN: 978-3-319-91716-0. DOI: 10.1007/978-3-319-91716-0_43. URL: http://link.springer.com/10.1007/978-3-319-91716-0_43 (visited on 07/12/2021).

APPENDIX A - CONCEPT MAP

Figure 1: Concept Map *(self-composed)*

APPENDIX B - GANTT CHART

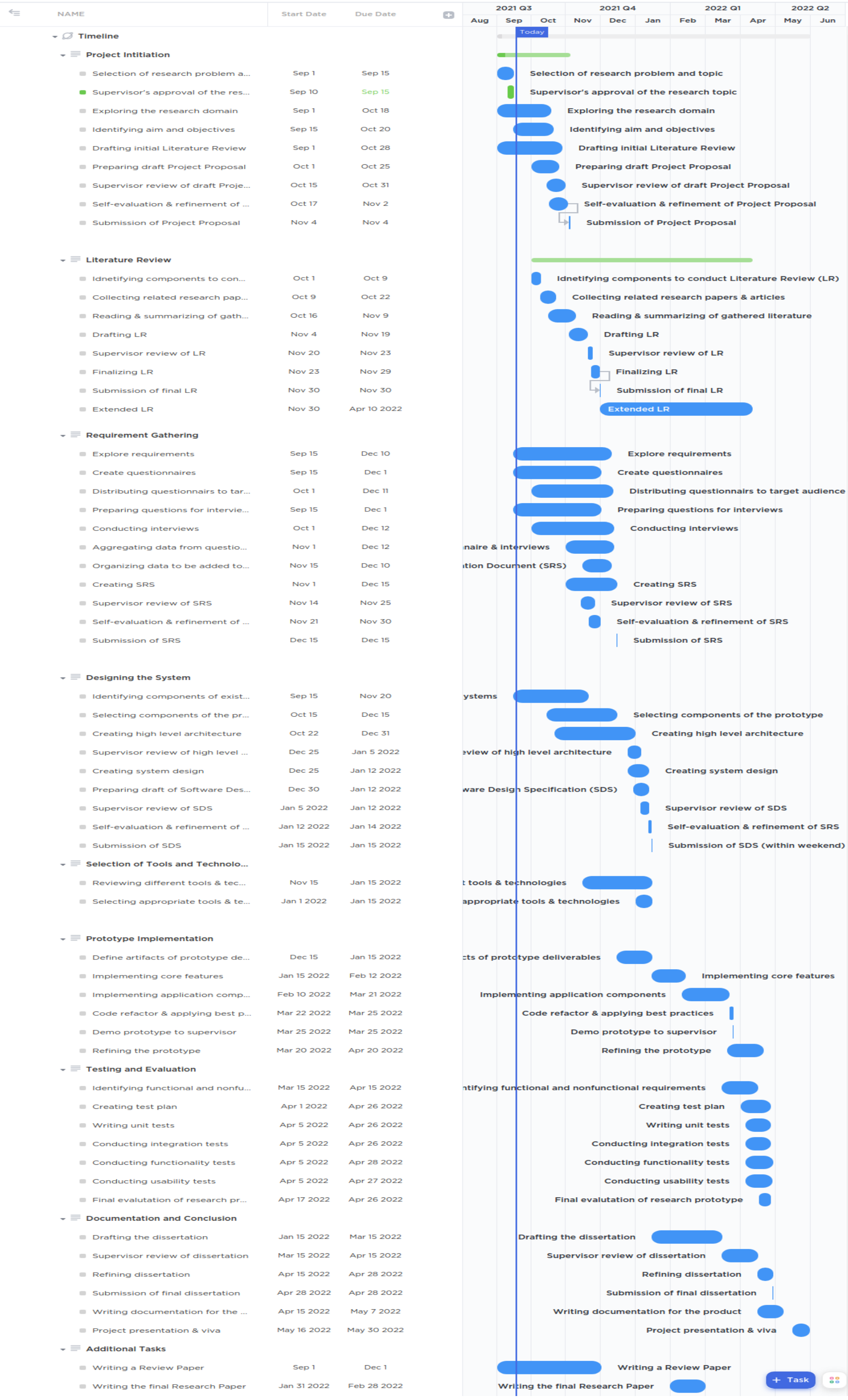


Figure 2: Gantt Chart

APPENDIX C - UI WIREFRAMES

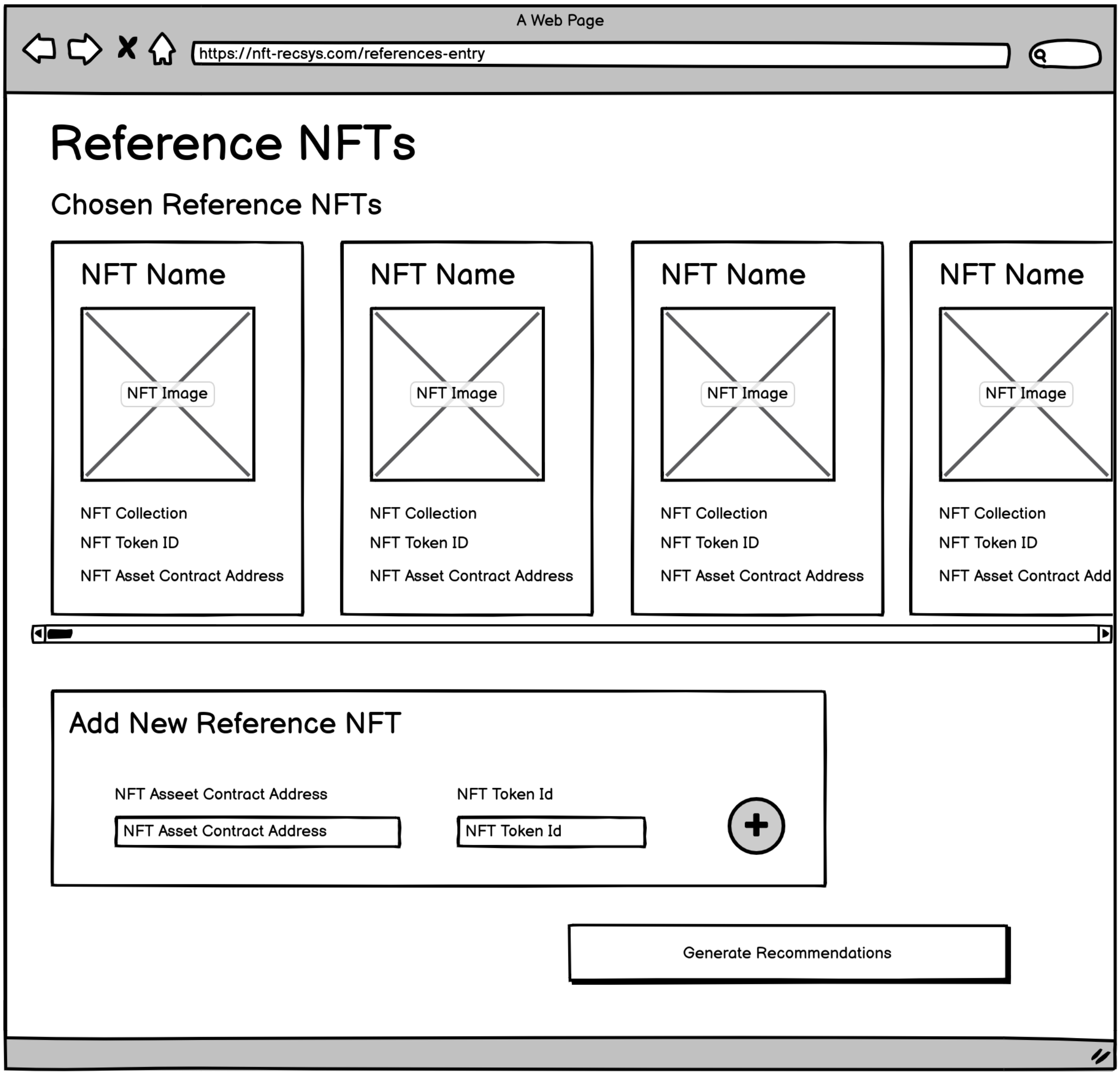


Figure 3: UI Wireframe - Reference NFT entry *(self-composed)*

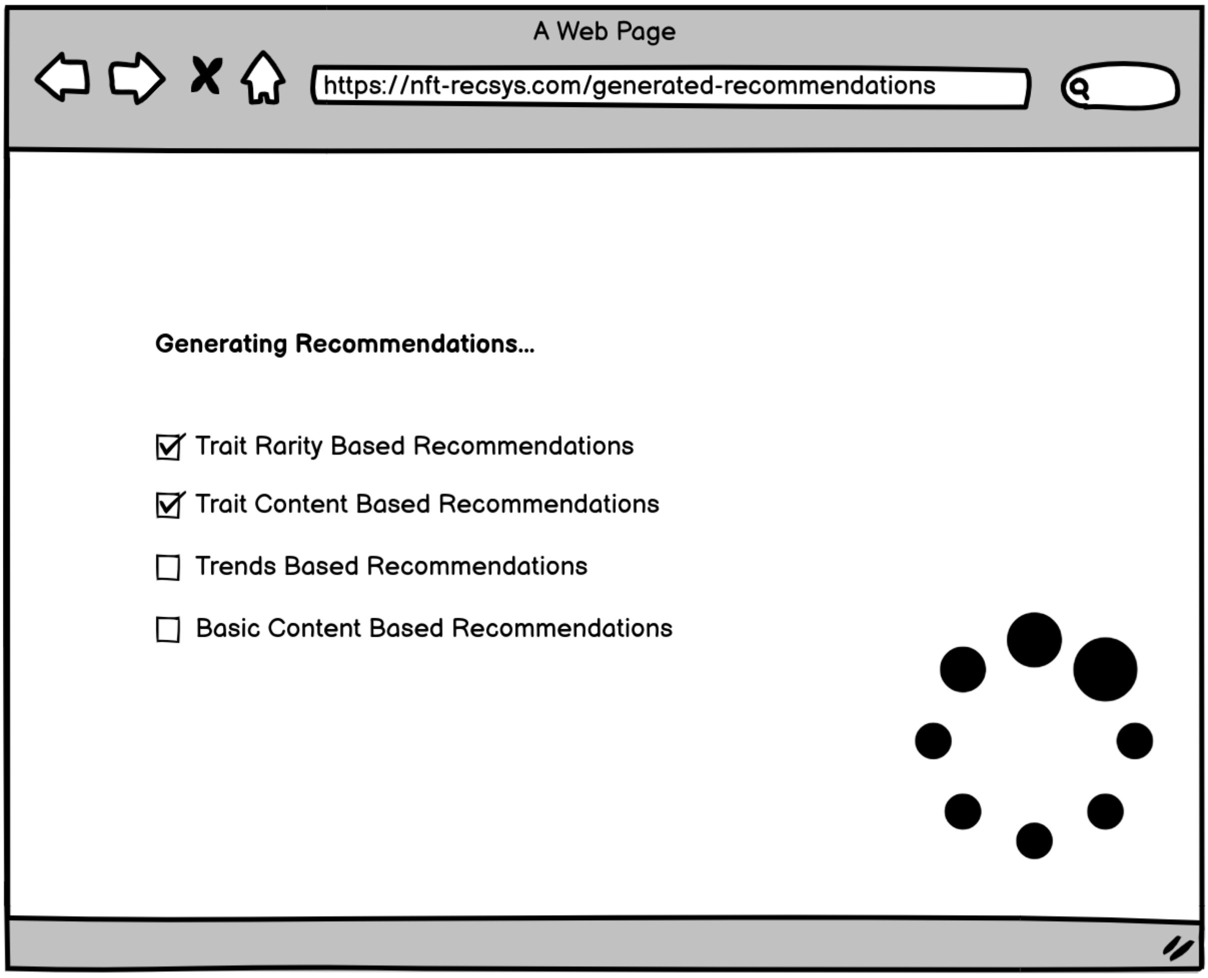


Figure 4: UI Wireframe - Loading-Generating Recommendations *(self-composed)*

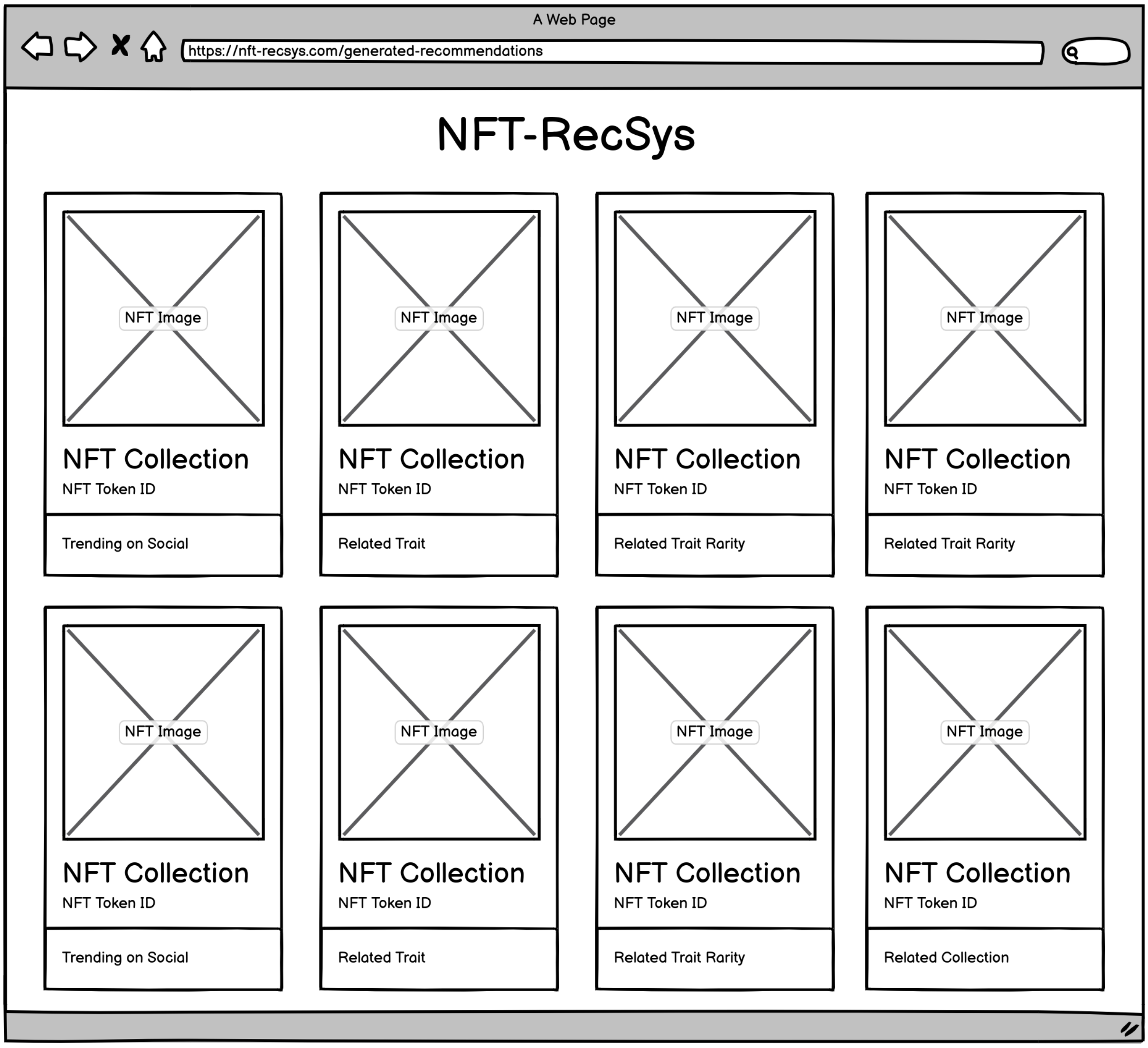


Figure 5: UI Wireframe - Generated Recommendations *(self-composed)*

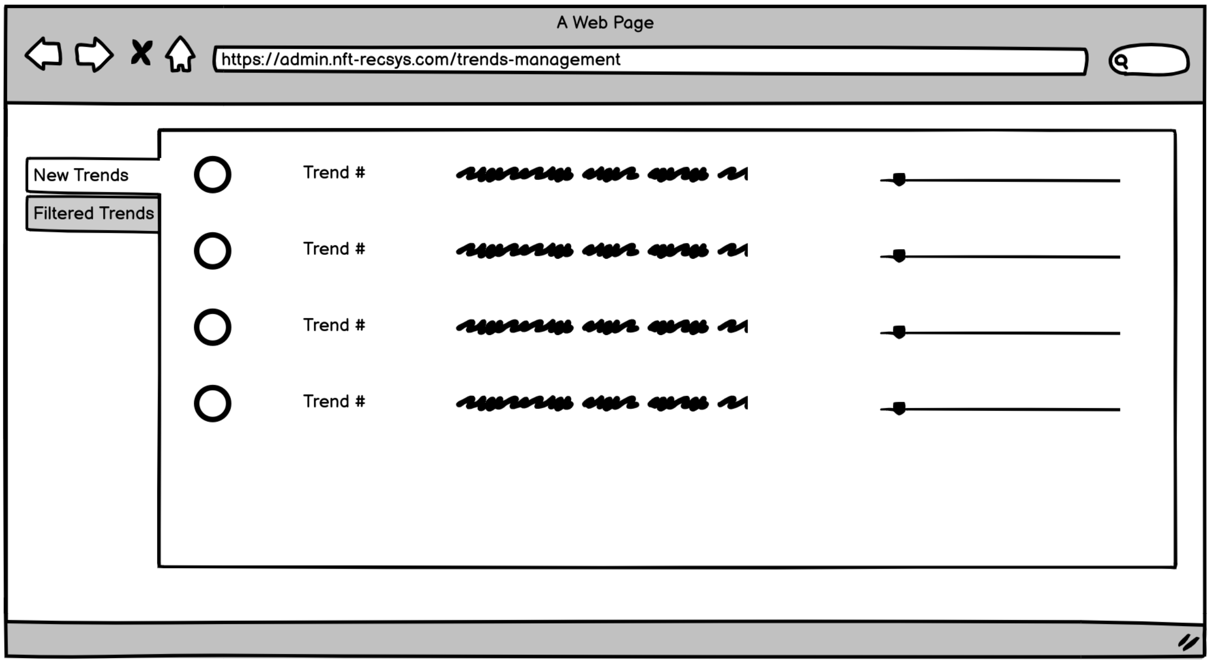


Figure 6: UI Wireframe - Admin Trends *(self-composed)*

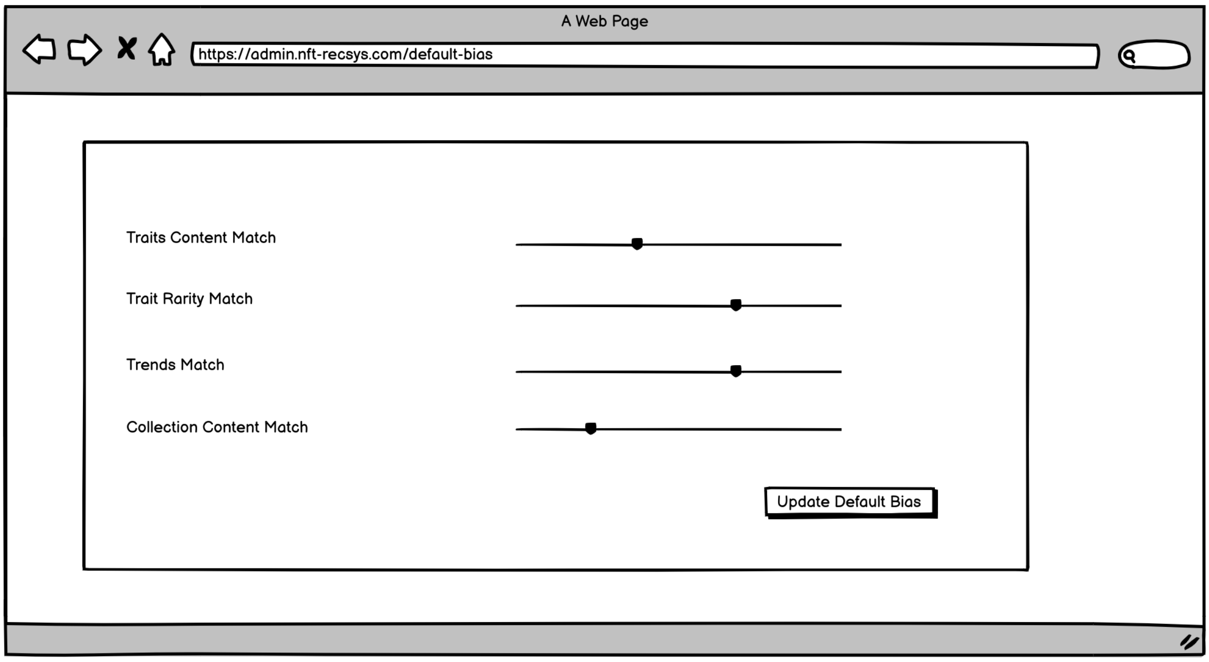


Figure 7: UI Wireframe - Admin Default Bias selection *(self-composed)*

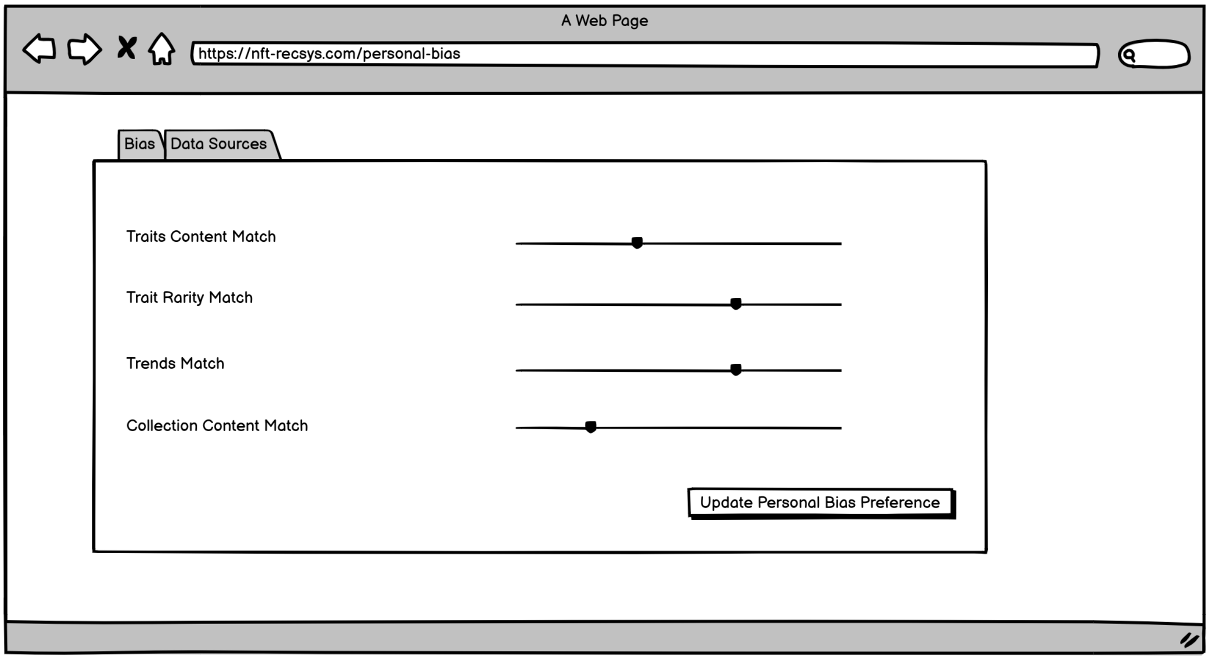


Figure 8: UI Wireframe - User Bias selection *(self-composed)*

APPENDIX D - TESTING

Appendix D1 - Model Testing

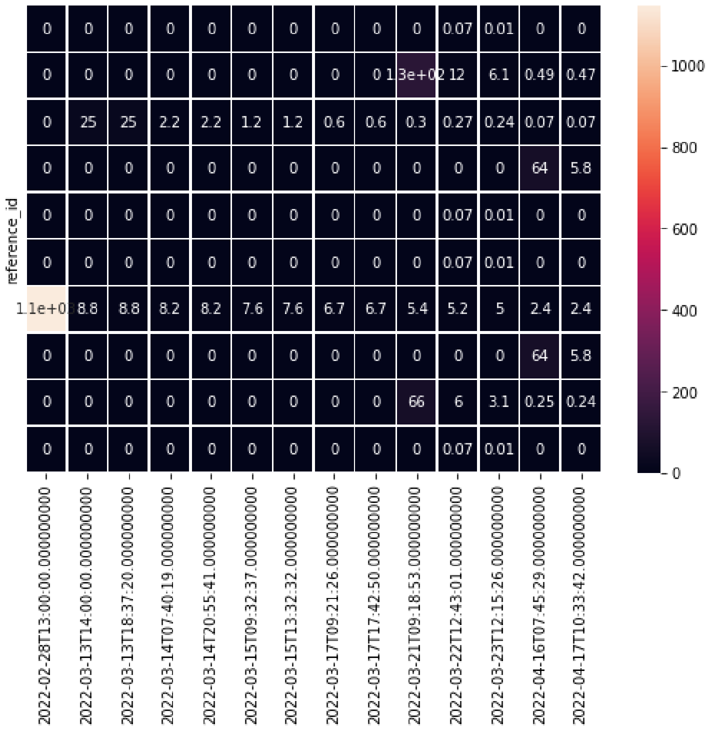


Figure 9: Trends based Recommender Testing Annotated Heatmap - 10 random items *(self-composed)*

Figure 10: Trends based Recommender Testing Heatmap - max score 30 *(self-composed)*

Figure 11: Trends based Recommender Testing Heatmap - max score 100 *(self-composed)*

Figure 12: Trends based Recommender Testing Heatmap - All items *(self-composed)*

reference_id	2022-02-28 13:00:00	2022-03-13 14:00:00	2022-03-13 18:37:20	2022-03-14 07:40:19	2022-03-14 20:55:41	2022-03-15 09:32:37	2022-03-15 13:32:32	2022-03-17 09:21:26	2022-03-17 17:42:50	2022-03-21 09:18:53	2022-03-22 12:43:01	2022-03-23 12:15:26	2022-04-16 07:45:29	2022-04-17 10:33:42
0x495f947276749ce646f68ac8c248420045cb7b5e-30041571019036570279696425714306755064417936087066583674697121488131734896641	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	893.14	81.19
0x495f947276749ce646f68ac8c248420045cb7b5e-94916783746142258917689322291875902080657954662840061252830161925292293095425	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	593.52	53.96
0x495f947276749ce646f68ac8c248420045cb7b5e-11004477123036844155157199207004096353062349592953033517586526381715485412556 9	0.00	0.00	0.00	54.92	54.92	4.99	4.99	1.77	1.77	0.77	0.68	0.60	0.17	0.16
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263433004574310401	0.00	0.00	0.00	54.92	54.92	4.99	4.99	1.77	1.77	0.77	0.68	0.60	0.17	0.16
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263434104085938177	0.00	0.00	0.00	54.92	54.92	4.99	4.99	1.77	1.77	0.77	0.68	0.60	0.17	0.16
0x495f947276749ce646f68ac8c248420045cb7b5e-54913028823869484484969640998172385576008206379346437203989969339057359028225	1148.85	8.77	8.77	8.15	8.15	7.61	7.61	6.72	6.72	5.44	5.20	4.97	2.44	2.39
0x495f947276749ce646f68ac8c248420045cb7b5e-94916783746142258917689322291875902080657954662840061252830161918695223328769	0.00	774.92	774.92	70.45	70.45	36.90	36.90	18.90	18.90	9.57	8.52	7.67	2.27	2.21
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263437402620821505	0.00	774.92	774.92	70.45	70.45	36.90	36.90	18.90	18.90	9.57	8.52	7.67	2.27	2.21
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263436303109193729	0.00	774.92	774.92	70.45	70.45	36.90	36.90	18.90	18.90	9.57	8.52	7.67	2.27	2.21
0x495f947276749ce646f68ac8c248420045cb7b5e-25415852838154585030182810174953724094201545459452816665085826161511803912193	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	61.35
0xd85ebb24bc0c2fcd6901cc9ae7409e41d4a9e0a3-345	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	63.77	5.80
0x495f947276749ce646f68ac8c248420045cb7b5e-33667053789437555920986692599124277764649475007205440601468718220039543586817	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	63.77	5.80
0x495f947276749ce646f68ac8c248420045cb7b5e-33667053789437555920986692599124277764649475007205440601468718221139055214593	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	63.77	5.80
0xd85ebb24bc0c2fcd6901cc9ae7409e41d4a9e0a3-346	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	63.77	5.80
0x495f947276749ce646f68ac8c248420045cb7b5e-74915313939230038050448181057740401670280602835803512556494204368644347002 89	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	63.77	5.80
0xd85ebb24bc0c2fcd6901cc9ae7409e41d4a9e0a3-341	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	63.77	5.80
0xd85ebb24bc0c2fcd6901cc9ae7409e41d4a9e0a3-342	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	63.77	5.80
0x495f947276749ce646f68ac8c248420045cb7b5e-46813158413327833598722183941328804250293517675741658106197955405879632199 69	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	63.77	5.80
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263417611411521537	248.23	1.89	1.89	1.76	1.76	1.64	1.64	1.45	1.45	1.18	1.12	1.07	0.53	0.52
0x495f947276749ce646f68ac8c248420045cb7b5e-77674177141111627268613029660198545550869301162278412343025582587737142198273	0.00	0.00	0.00	0.00	0.00	20.09	20.09	0.96	0.96	0.33	0.28	0.25	0.06	0.06
0x495f947276749ce646f68ac8c248420045cb7b5e-79598145459366138456148796052825547483993807520203766226242728957798908952577	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	128.37	11.67	6.11	0.49	0.47
0x495f947276749ce646f68ac8c248420045cb7b5e-	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Figure 13: Trends based Recommendations Heatmap Data *(self-composed)*

Appendix D2 - Model Evaluation of Test Results

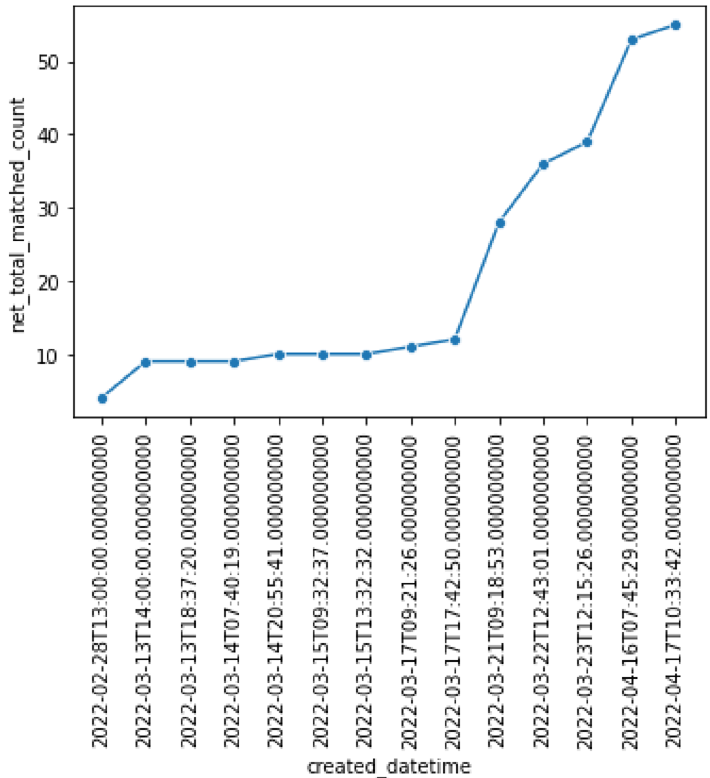


Figure 14: Total Trends based Recommendations made with time *(self-composed)*

	created_datetime	trend_count	total_matched_items_count	new_matched_items_count	net_total_matched_count
0	2022-02-28T13:00:00.000000000	50	4	3	4.0
1	2022-03-13T14:00:00.000000000	50	5	3	9.0
2	2022-03-13T18:37:20.000000000	50	0	0	9.0
3	2022-03-14T07:40:19.000000000	50	0	0	9.0
4	2022-03-14T20:55:41.000000000	50	1	1	10.0
5	2022-03-15T09:32:37.000000000	48	0	0	10.0
6	2022-03-15T13:32:32.000000000	48	0	0	10.0
7	2022-03-17T09:21:26.000000000	39	1	1	11.0
8	2022-03-17T17:42:50.000000000	48	1	1	12.0
9	2022-03-21T09:18:53.000000000	48	16	2	28.0
10	2022-03-22T12:43:01.000000000	48	8	1	36.0
11	2022-03-23T12:15:26.000000000	48	3	2	39.0
12	2022-04-16T07:45:29.000000000	50	14	5	53.0
13	2022-04-17T10:33:42.000000000	50	2	2	55.0

Figure 15: Trends based Recommendations Trends Matches Data *(self-composed)*

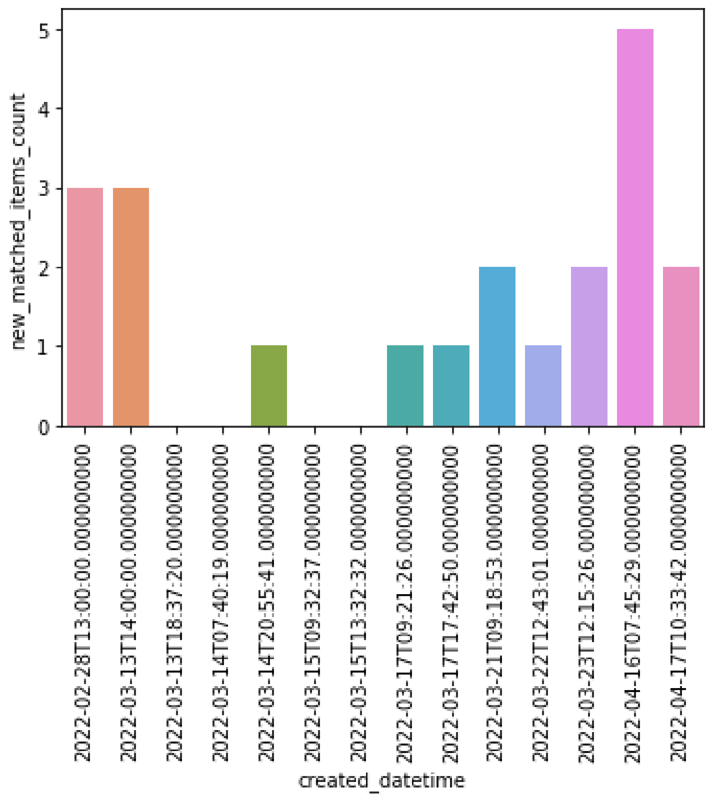


Figure 16: Trends based Recommendations Newly Matched Items *(self-composed)*

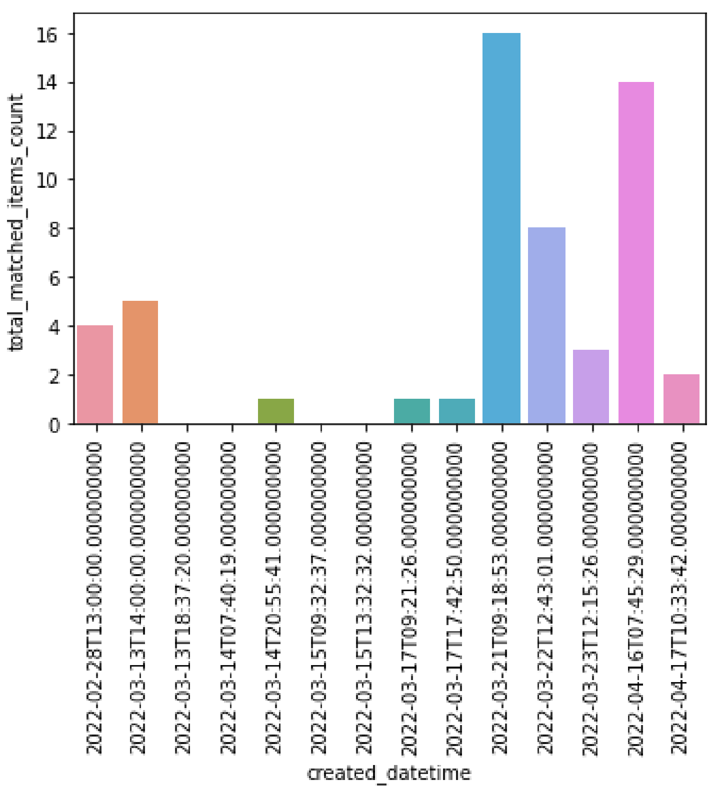


Figure 17: Trends based Recommendations Total Matched Items *(self-composed)*

Appendix D3 - Functional Testing

Table 1: Testing results of Functional Requirements

Test Case	FR ID	User Action	Expected Result	Actual Result	Result Status
1	FR1	Users adds a chosen NFT to be considered as the reference	item details are fetched and validated	item details were fetched and validated	Passed
2	FR2	Admins adds a collection of NFTs to be used as recommendations.	The details of the NFTs get fetched and pre-processed for recommendations	The details of the NFTs were fetched and pre-processed for recommendations	
3	FR3	A user enters a contract address & token Id of an NFT.	The system fetches relevant data of the NFT	The system fetched relevant data of the NFT	Passed
4	FR4	Users sets/ adjusts the bias and parameters to be used	The user specific bias and general bias get adjusted	The user specific bias and general bias were adjusted	
5	FR5	Admins adjusts the admin bias	The admin-bias gets adjusted in the system	The admin-bias was adjusted in the system	
6	FR6	Users clicks a button to generate recommendations	Recommendations are generated and made visible to the user	Recommendations were generated and made visible to the user	Passed
7	FR7	A user enters feedback regarding the satisfaction level of the generated recommendations	The feedback of the user is collected and stored in the Database	The feedback of the user was collected and stored in the Database	

8	FR8	User requests the reason for recommending the item	Reasons for recommending each item is displayed	Reasons for recommending each item was displayed	Passed
10	FR10	User requests featured trending NFT recommendations	Opinion mining trends data is used to generate NFT recommendations.	Opinion mining trends data was used to generate NFT recommendations.	Passed
11	FR11	User inputs data-points such as interested public figures, websites to use as opinion mining data for recommendations	The data-points entered are pre-processed and used for trends based recommendations	The data-points entered were pre-processed and used for trends based recommendations	
12	FR12	Admins inputs data-points such as interested public figures, websites to use as opinion mining data for recommendations.	The data-points entered are pre-processed and used for trends based recommendations	The data-points entered were pre-processed and used for trends based recommendations	

APPENDIX E - EVALUATIONS

Appendix E1 - Evaluations received by Evaluators

Table 2: Evaluations received by Evaluators

Evaluator	**Feedback**
Mr. Sharmilan Somasundaram [*CEO of Niftron - Blockchain as a Service, Certified Blockchain Solution Architect (CBSA)*]	Because there's no system like this, it's a good research. The research project is good because gathering data & domain side is difficult & new. The trends based model needs more evaluation. Try synthesizing data to show how recommended items vary across time. Try to show the significance of using the model.
Mr. Nipuna Senanayake []	The concept of the trends based recommendations system is good. Evaluation of the trends based model is a bit of a concern. Might be possible to evaluate it by web-scraping. Good amount of work has been done. Keep up the same enthusiasm for research, it will help in the long run, wherever you go.
Anonymous [*Blockchain Masters Researcher*]	The research is good, haven't seen NFT researches that much, although there're quite a lot of Blockchain researches. Will give an A for the project since proper research has been done with the identified research gap. Price prediction might be possible using art market pricing (if available) since the NFT market is similar to the market.

Appendix E2 - Evaluation of Functional Requirements

Table 3: Evaluation of the implementation of Functional Requirements

FR ID	**Requirement**	**Priority Level**	**Use Case**	**Evaluation**
FR1	Users must be able to add a chosen NFT to be considered as the reference point to generating recommendations.	M	UC1	Implemented

FR2	Admins should be able to add a collection of NFT to be used as recommendations.	S	UC1	
FR3	The system could be able to fetch relevant data of the NFT using an entered token Id.	C	UC1	Implemented
FR4	Users must be able to set/ adjust the bias and parameters to be used by the Recommendations System using parametric selections prior to generating recommendations.	M	UC2	
FR5	Admins should be able to adjust the default bias of the Recommendations System.	S	UC3	
FR6	Users must be able to view recommendations with the click of a button.	M	UC4	Implemented
FR7	The prototype could have an option to receive user feedback regarding the satisfaction level of the generated recommendations by the system.	C	UC4	
FR8	The system could show the reasons for recommending each item to users.	C	UC4	Implemented
FR9	The system should generate price predictions and consider the results for recommendations.	S	UC5	
FR10	Opinion mining trends data must be used to generate NFT recommendations.	M	UC7	Implemented
FR11	A user could be allowed to feed data-points such as interested public figures, websites to use as opinion mining data for recommendations.	C	UC8	
FR12	Admins should be able to feed data-points such as interested public figures, websites to use as opinion mining data for recommendations.	S	UC8	
FR13	User-input could be aggregated and used as a reinforcement learning bias for the Recommendations Model.	C		

FR14	The system will not act as a decentralized system.	W		
Functional Requirement Completion Percentage = $\frac{}{14} * 100 = \%$				

Appendix E3 - Evaluation of Non-Functional Requirements

Table 4: Evaluation of the implementation of Non-functional requirements

NFR ID	**Requirement**	**Priority Level**	**Evaluation**
1	Performance	Desirable	Implemented
2	Quality of Output	Important	Implemented
3	Security	Desirable	
4	Usability	Important	
5	Scalability	Desirable	
Non-Functional Requirement Completion Percentage = $\frac{}{5} * 100 = \%$			

Appendix E4 - Self Evaluation