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

May 2022

Submitted in partial fulfilment of the requirements for the

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

DECLARATION

I hereby declare that this dissertation and its associated sub-components are the result of my own research efforts, and that none of these have been or are currently being submitted/presented as content for any degree or other qualification program to any other university or institution. Extracted facts from reliable external sources have been appropriately cited.

Student Name - Dinuka Ravijaya Piyadigama
Registration Number - w17421047/2 | 2018373

Signature: **Date:**13/05/2022

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 presents ensembled Recommendations techniques to produce trending recommendations of NFT assets, while preserving user-anonymity. The data extraction methods explored for recommending NFTs, exploration of features that can be used for recommendations & the integration of social-trends into recommendations 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

1. **Exploration of the possibility of infusing Social Media Trends into generating NFT Recommendations**

   Conference: ACM RecSys 2022

   Status: Submitted, In-Review

   Venue: Seattle, WA, USA

   Date: September 18–23, 2022

2. **An Analysis of the Features Considerable for NFT Recommendations**

   Conference: 15TH IEEE International Conference on Human System Interaction

   Status: Submitted, In-Review

   Venue: Melbourne, Australia

   Date: July 29-31, 2022

   ArXiv pre-print: `https://arxiv.org/abs/2205.01802`

3. **A Review on Pushing the Limits of Baseline Recommendation Systems with the integration of Opinion Mining & Information Retrieval Techniques**

   ArXiv pre-print: `https://arxiv.org/abs/2205.01802`

ACKNOWLEDGEMENT

I had many people in my life that enabled me to go above and beyond what I used to be capable of to get to where I am today and successfully complete this final year research project.

I would first and foremost like to that my supervisor, Mr. Guhanathan Poravi sir for pushing me beyond my limits and showing me the best possible path of conducting research. Thanks to him, I have the confidence that I have the capability of conducting research at any level. I would like to thank the evaluators of this project who spent their precious time going through the work I had done and for giving advice on how I can improve my work at the ideation stage, the requirement gathering stage as well as the final evaluation stage of this research. This includes industry & academia experts as well as seniors & juniors of IIT.

I would like to thank all the lecturers at the Informatics Institute of Technology as well as the management for the opportunities, support & guidance they presented me with throughout my undergraduate degree. I may not have been able to become half the developer/ researcher I am today if not for the continued support & positive peer pressure of my friends from uni who have enabled me to grow exponentially fast over the course of 4 years. Finally, I would like to thank my parents for their support & encouragement to pursue my dreams and my family & friends from school for the encouragement shown toward me through tough times.

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.
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.
RecSys	Recommendation System.
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

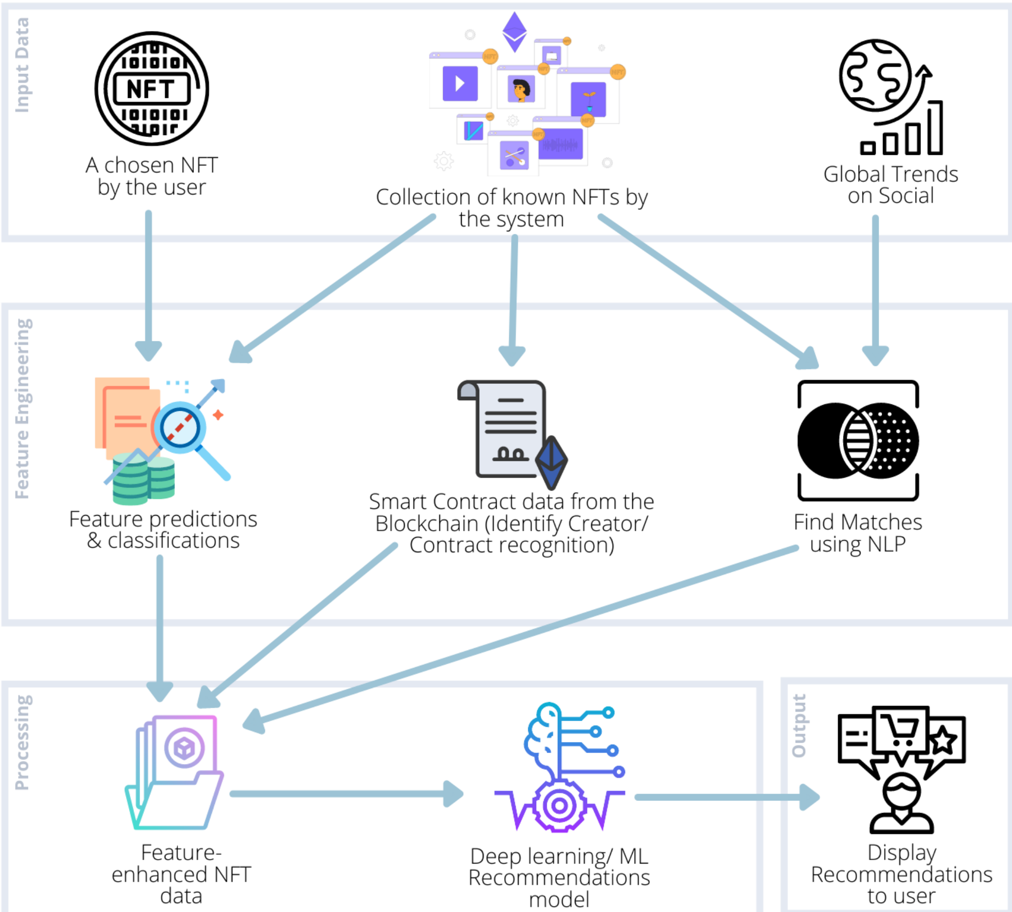


Figure 1.1: Prototype Feature Diagram *(self-composed)*

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 meta-verse (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

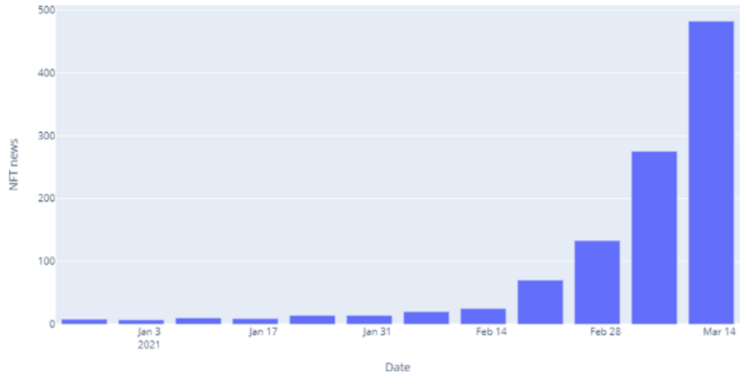


Figure 2.1: News trends in 2021 related to NFTs (Dowling, 2021a)

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.

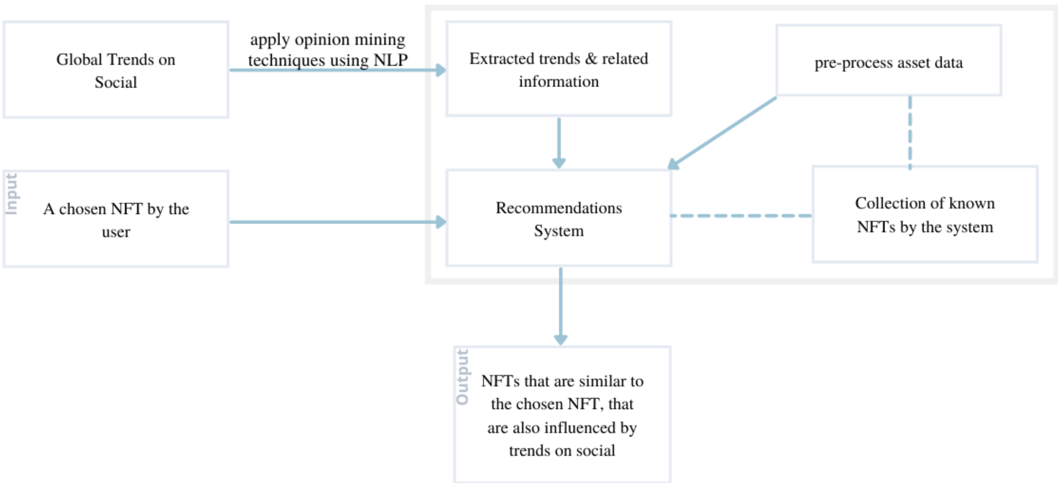


Figure 2.2: Proposed architecture of a Recommendations System for NFTs *(self-composed)*

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.

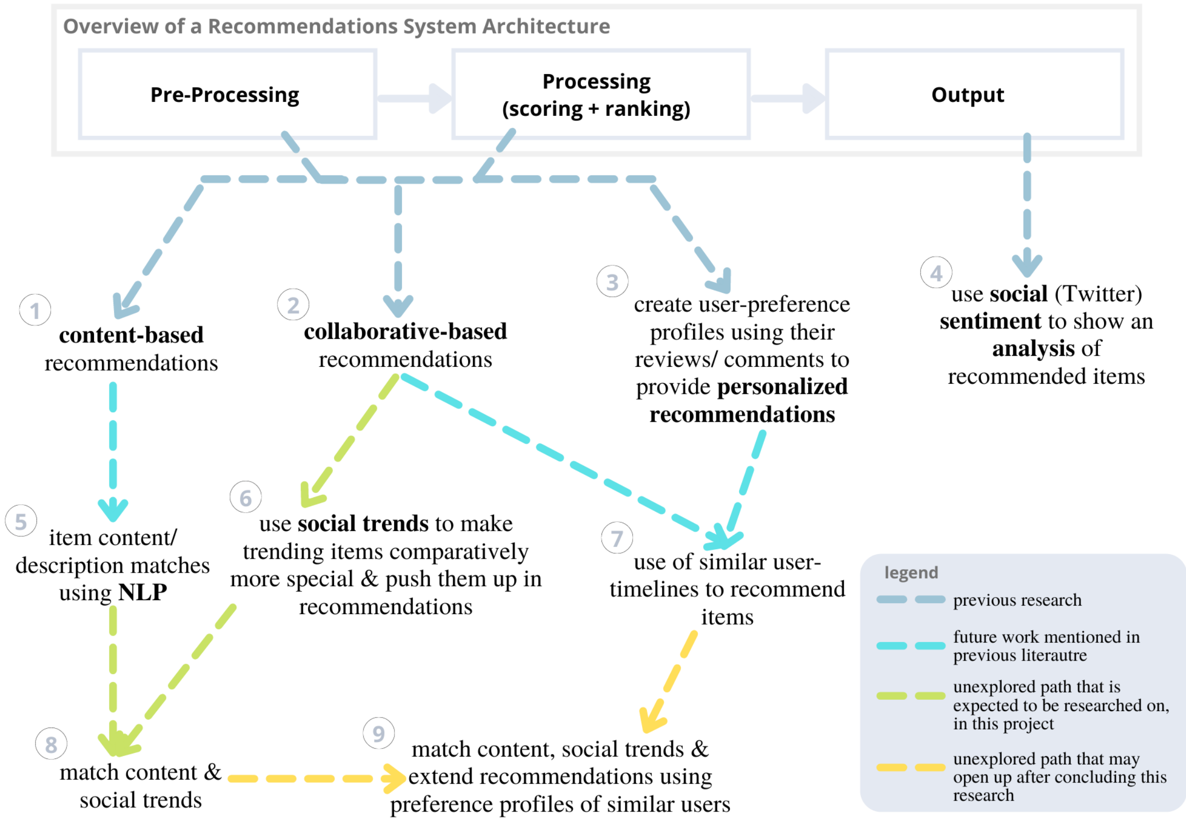


Figure 2.3: Enhancements done to Recommendation Systems using opinion mining techniques *(self-composed)*

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.

3.3 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 related Recommendation Systems	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

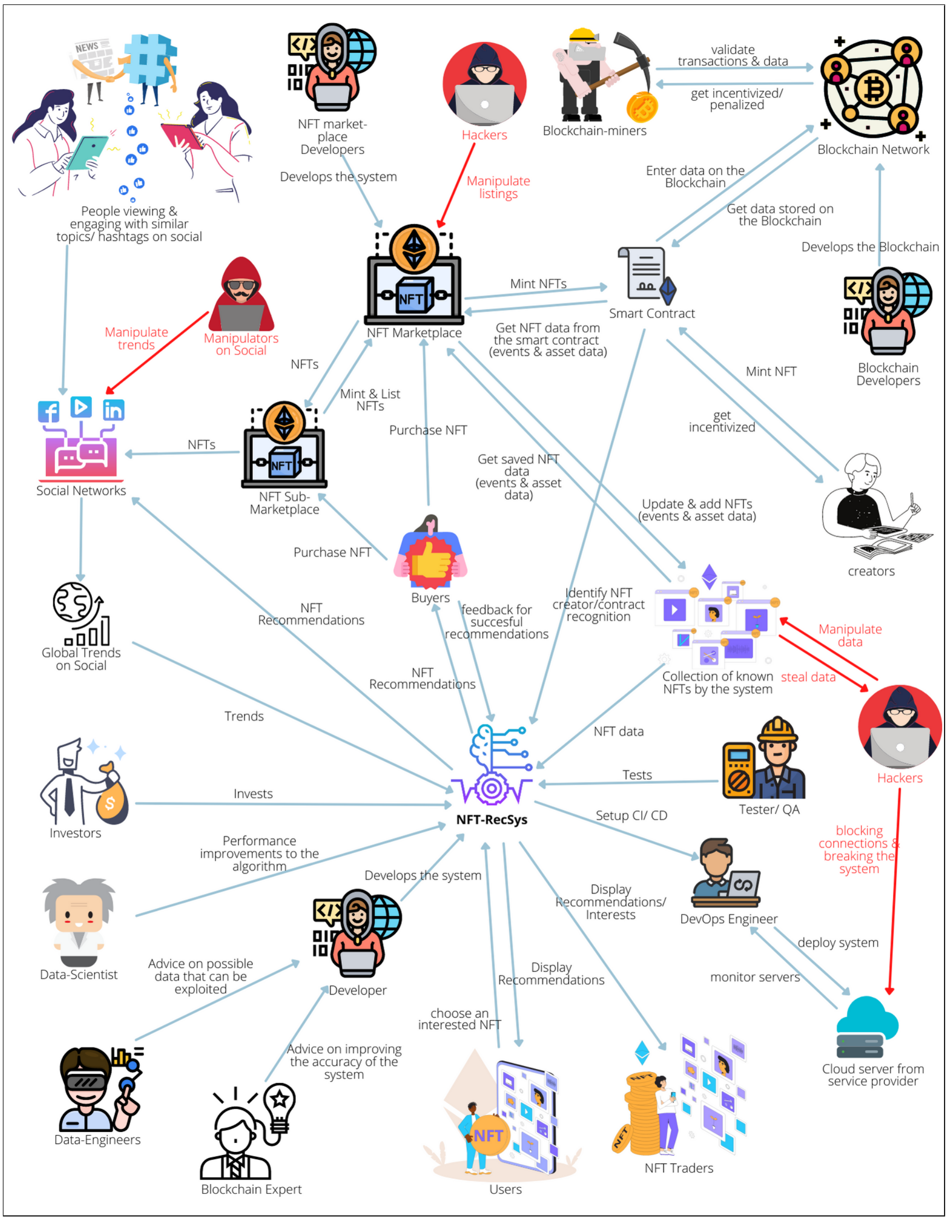


Figure 4.1: Rich Picture Diagram *(self-composed)*

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

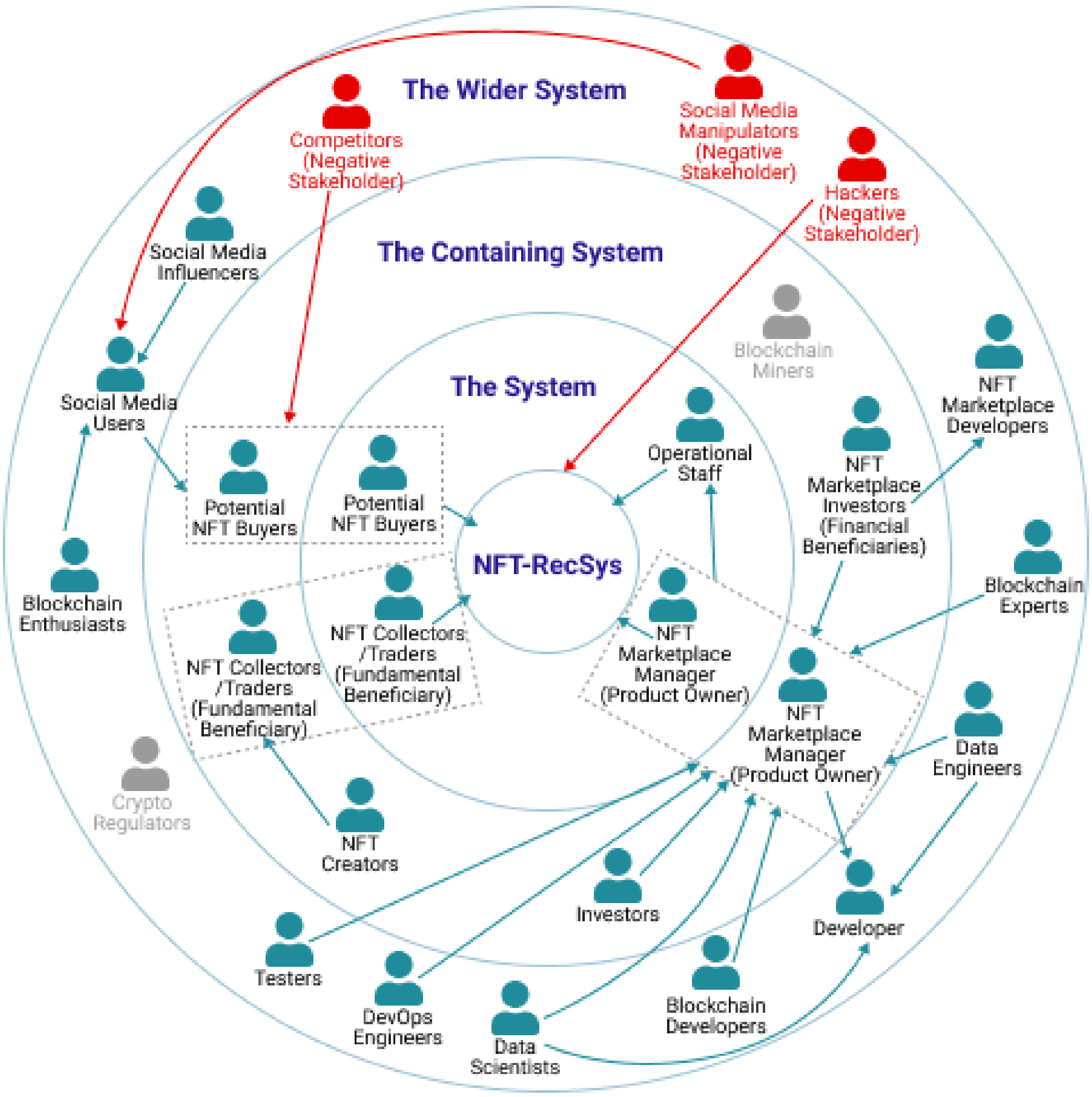


Figure 4.2: Stakeholder Onion Model *(self-composed)*

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.

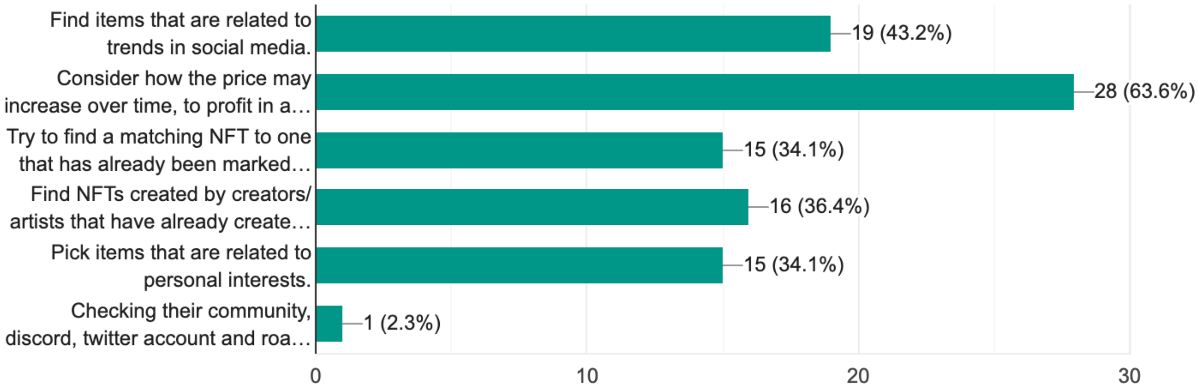
Table 4.4: Thematic analysis of interview findings

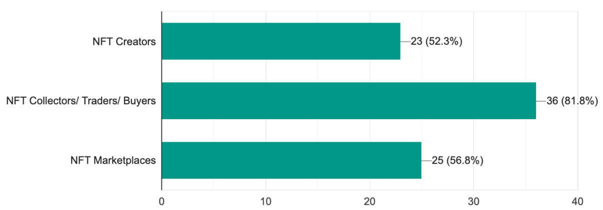
Theme	Analysis
Collection & pre-processing of available data.	As this is expected to be a Data science project, the main concern that all participants had was the availability of data. Clustering of available data was suggested to identify possible patterns by ML experts, while Blockchain experts suggested 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

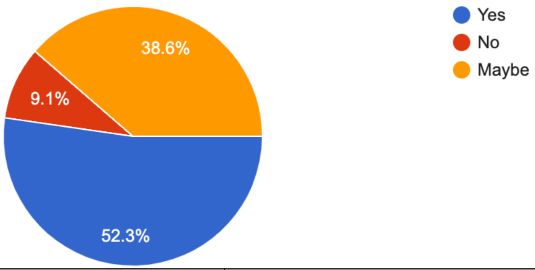
Table 4.5: Analysis of replies to questionnaire

<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>

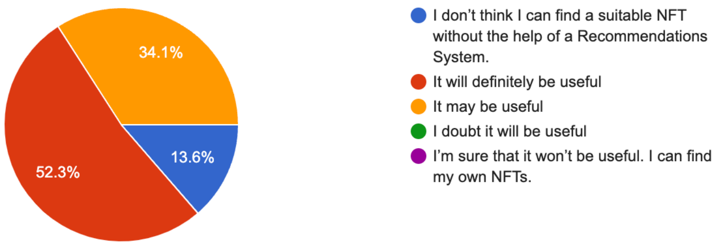
Findings & Conclusion



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.

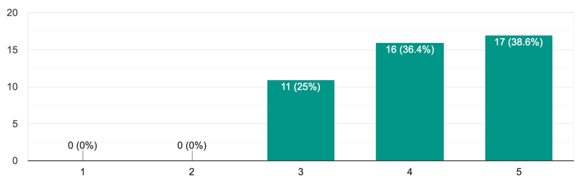
Findings & Conclusion



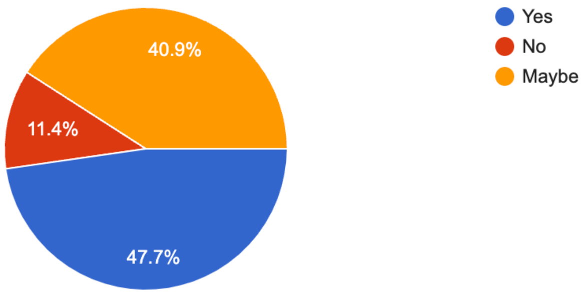
52.3% of users thought that a Recommendations System would definitely be useful to them if they plan on purchasing an NFT, while 34.1% thought that it may be useful. Meanwhile, 13.6% of users thought that they don't think that they could find a suitable NFT without the help of a Recommendations System.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

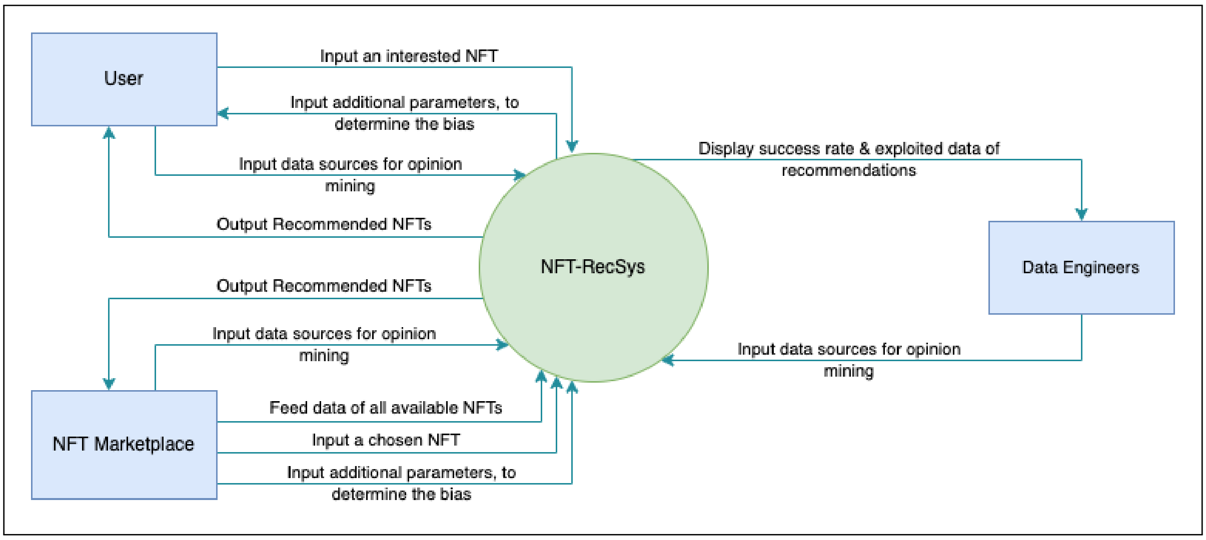


Figure 4.3: Context Diagram *(self-composed)*

4.8 Use Case Diagram

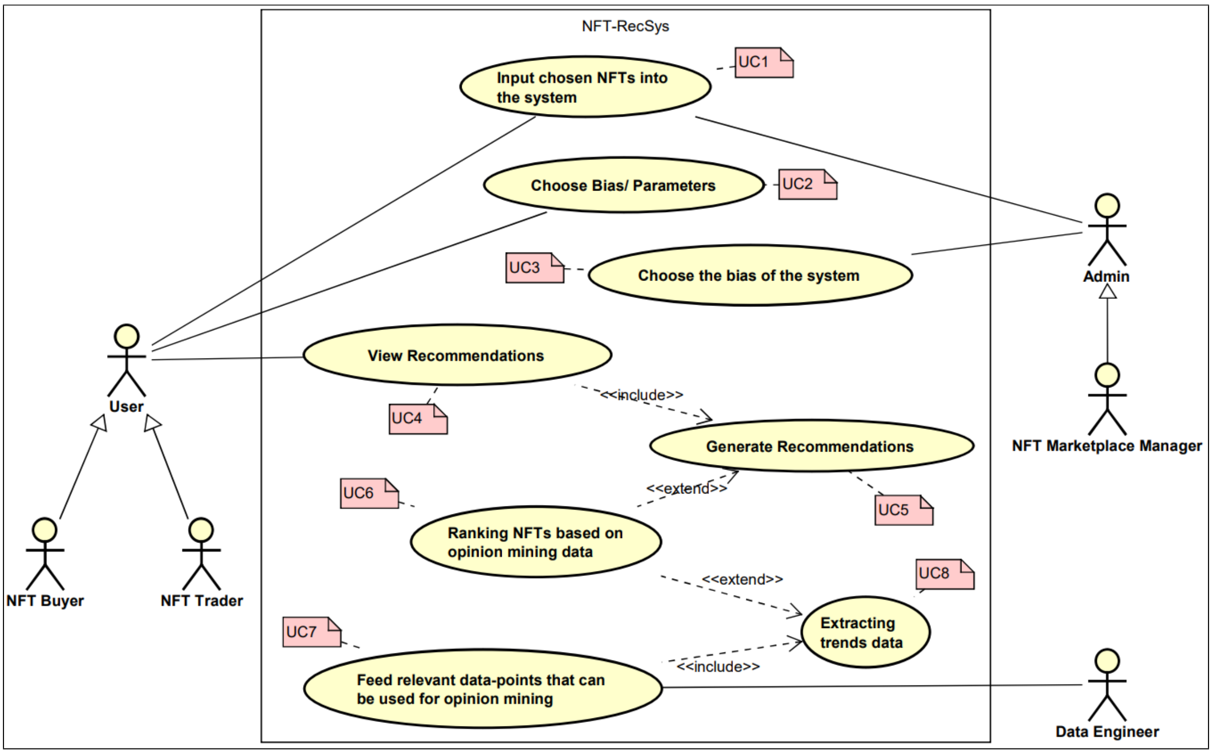


Figure 4.4: Use Case Diagram *(self-composed)*

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 contract address & 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 choose the 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 recommendations based on what the user expects to view	S	UC5
FR10	Opinion mining trends data must be used to generate NFT recommendations.	M	UC8
FR11	Admins could be allowed to feed data-points such as various social networks, interested public figures, websites to use as opinion mining data for recommendations.	C	UC7
FR12	User-input could be used as a reinforcement learning bias for the Recommendations Model.	C	NA
FR13	The system will not act as a decentralized system.	W	NA

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 was 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 an 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*.

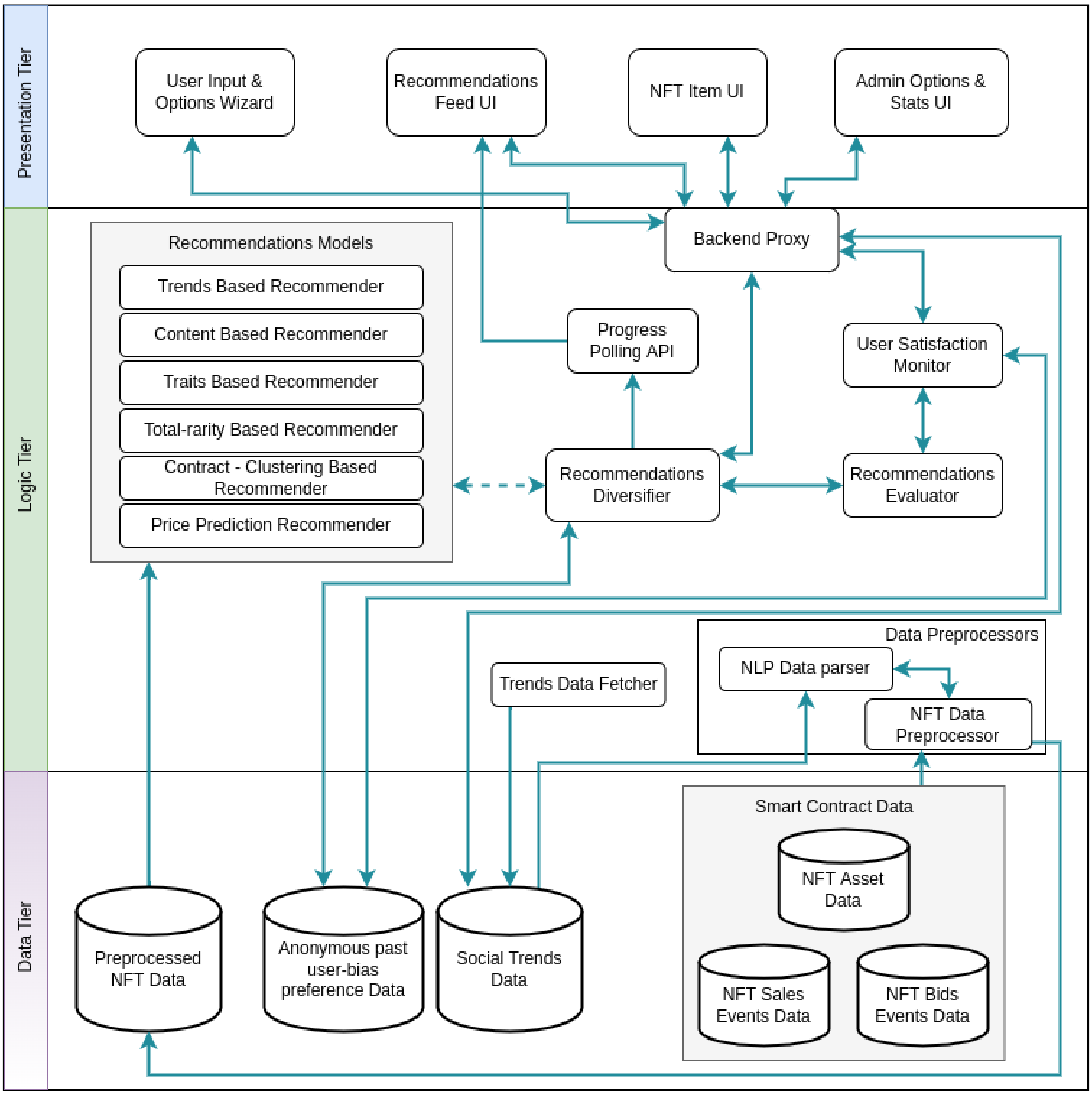


Figure 6.1: Three Tiered Architecture *(self-composed)*

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.

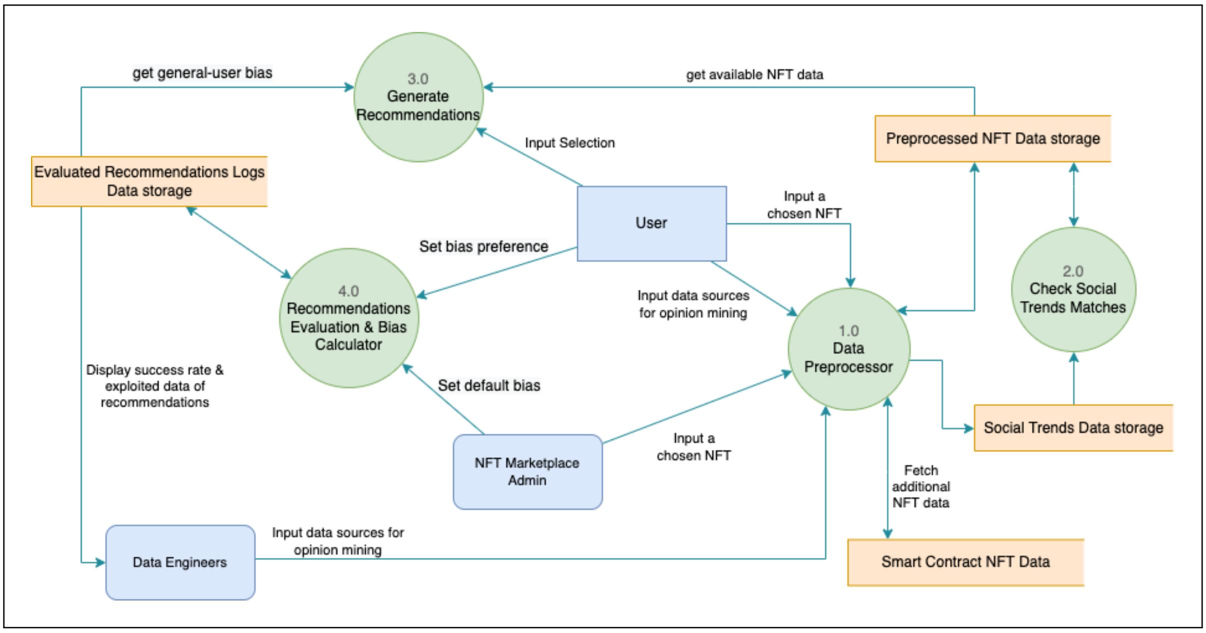


Figure 6.2: Data Flow Diagram - Level 1 *(self-composed)*

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

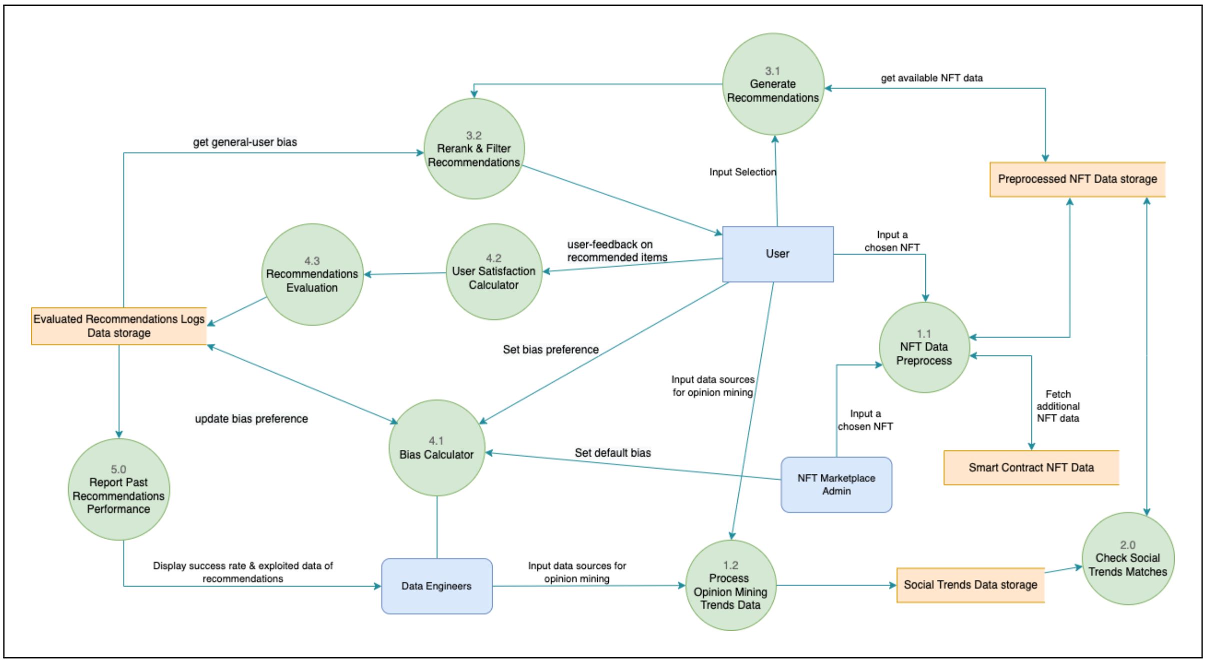


Figure 6.3: Data Flow Diagram - Level 2 *(self-composed)*

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.

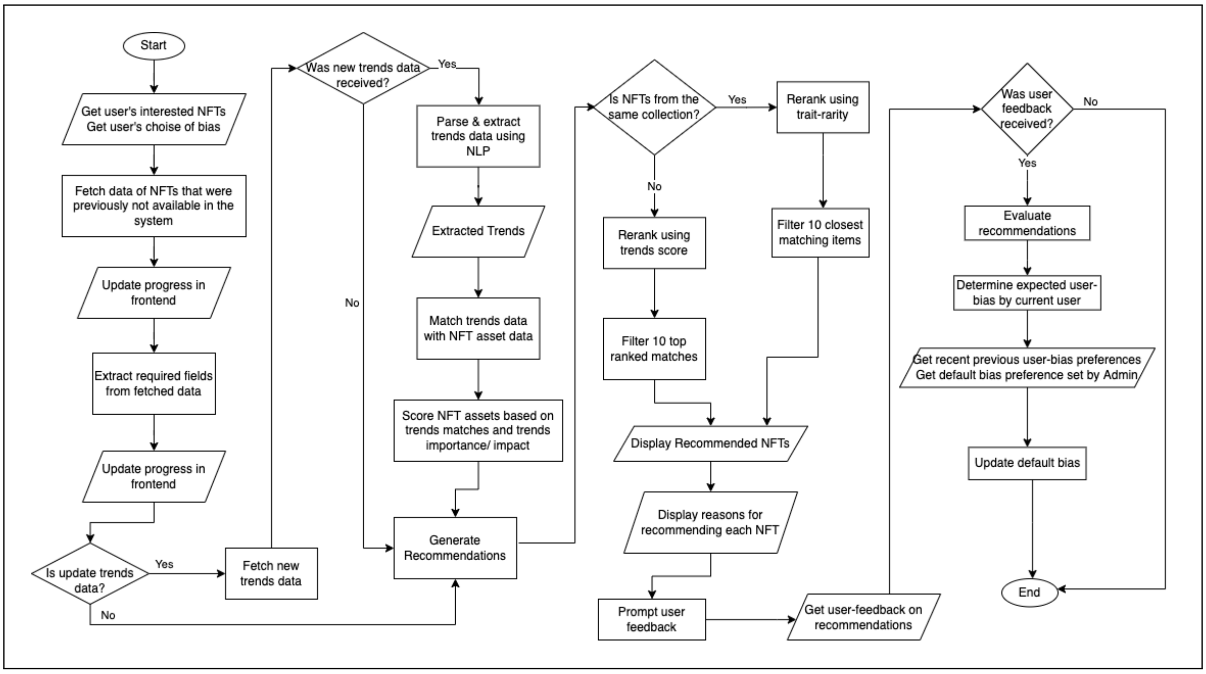


Figure 6.4: System Process Flow Chart*(self-composed)*

The process followed to generate Trend-based Recommendations has been displayed in a simplified process flow diagram in *Trends based Recommendations Process Flowchart (self-composed)* of *Appendix C - Design*.

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.

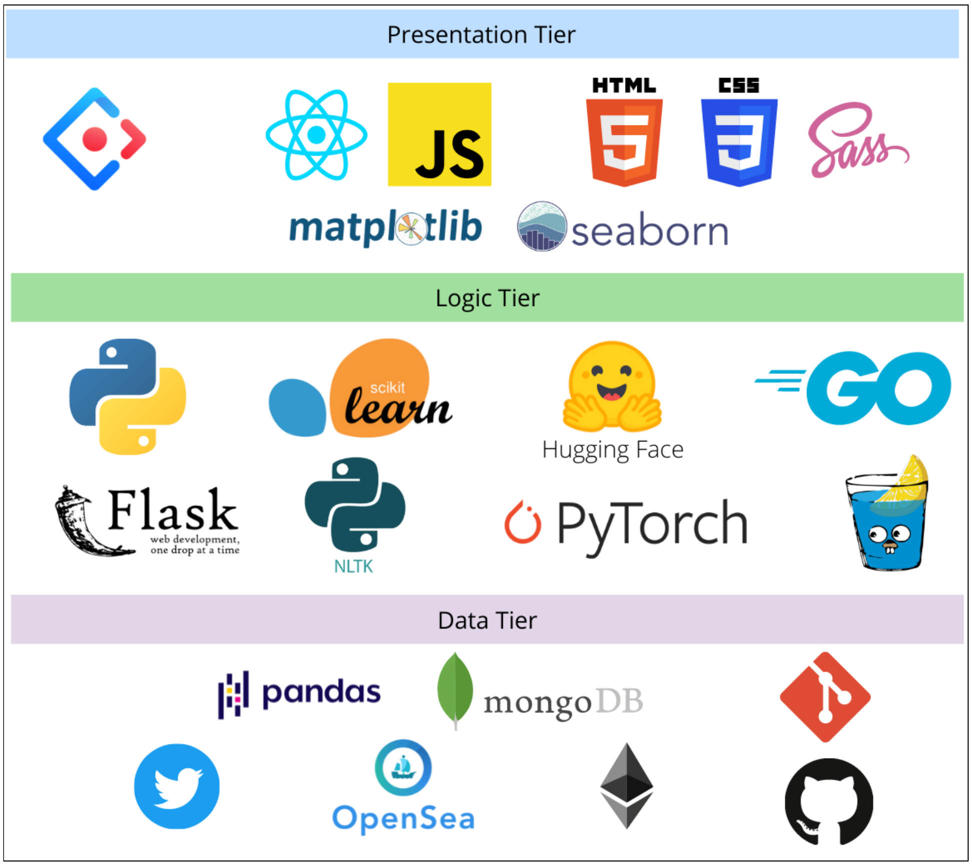


Figure 7.1: Technology Stack *(self-composed)*

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**.
- 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.

Flask	Easy to build APIs for Python.

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.
Hugging Face Transformers	Availability of pre-built high performance Open-source NLP Transformer models. A model that was built using Pytorch was chosen due to it's speed.
Matplotlib & Seaborn	Has almost any type of visualization method for data analysis.
React	A UI library that makes it easy to build interactive websites. It was important to develop an easily interactible frontend, since it will be the users' point of interaction with the system. This was easily doable thanks to the vast array of capabilities offered by React.

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.

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, Flask
UI Framework	Ant Design of React
Libraries	Pandas, Scikit-learn, NLTK, Matplotlib, React
IDE – Research	Google Colab
IDE – Product	VSCode, Golang
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.

The data extraction is done to extract information required for recommendations, to view

```python
traits = nft_dict['traits']

traits_string = ""

# save total rarity in a separate column
total_rarity = 0
for trait in traits:
  # for each trait, extract the trait_count and calculate rarity

  if trait['trait_count'] and (total_supply) and (total_supply != 0):
    # print("trait_count:" + trait['trait_count'], "total_supply:" + total_supply)
    trait_rarity = 1/ (trait['trait_count']/total_supply)

    total_rarity += trait_rarity

  # save all trait type and values in a ;; separated string. This will have to be split into an array when loaded - for content based filtering
  # lowercase -> remove any spaces between the words before adding into the string
  trait_type = trait['trait_type'].lower()

  if isinstance(trait['value'], str):
    trait_value = trait['value'].lower()
    trait_string = trait_type + trait_value
  else:
    trait_value = trait['value']
    trait_string = trait_type + str(trait_value)

  trait_string = trait_string.replace(" ", "")
  traits_string = traits_string + trait_string + ";;"    # typevalue;;typevalue
```

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

details of items & to save information for recommendation algorithms/ predictions that are potentially possible in the future.

Trait Content & Rarity Preprocessing, Vectorizing & Recommendations

```python
[ ] # 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.

```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)*

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.

The recommendation generation algorithms in Fig 7.5 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

```python
# function that takes in reference_id as input and returns the top 10 recommended nfts
def content_based_recommendations(reference_id, cosine_sim = cosine_sim):

    recommended_nfts = []
    cosine_sim_scores_of_recommendations = []

    # getting the index of the NFT that matches the reference_id
    idx = indices[indices == reference_id].index[0]

    # creating a Series with the similarity scores in descending order
    score_series = pd.Series(cosine_sim[idx]).sort_values(ascending = False)

    # getting the indexes of the 10 most similar nfts
    top_10_indexes = list(score_series.iloc[1:11].index)
    # getting the cosine similarities of the 10 most similar nfts
    cosine_sim_scores_of_recommendations = list(score_series.iloc[1:11])

    # populating the list with the reference_ids of the best 10 matching nfts
    for i in top_10_indexes:
        recommended_nfts.append(list(df.index)[i])

    return recommended_nfts, cosine_sim_scores_of_recommendations
```

```python
def trait_rarity_recommendations(reference_id):

    recommended_nfts = []
    trait_rarity_scores_of_recommendations = []

    input = df.loc[reference_id]['total_rarity']
    # print(input)

    #  This considers the entire dataframe. Need to do this only within a collection - send the filtered dataframe as a parameter
    # the dataframe with 10 closest values.
    df_sort = df.iloc[(df['total_rarity']-input).abs().argsort()[1:11]]

    recommended_nfts = df_sort.index.tolist()
    trait_rarity_scores_of_recommendations = df_sort['total_rarity'].tolist()
    # print(df_sort['total_rarity'].tolist())

    return recommended_nfts, trait_rarity_scores_of_recommendations
```

Figure 7.5: Implementation code segment: Generate Trait Rarity & Similarity Recommendations *(self-composed)*

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 Sentiment Analysis, Preprocessing & Recommendations

```python
highest_ranked_label = labels[ranking[0]]
highest_ranked_score = scores[ranking[0]]

if highest_ranked_label == "negative":
  print(f"{highest_ranked_label} sentiment")
  return - np.round(highest_ranked_score, 4) # round the output to 4 decimal places
elif highest_ranked_label == "neutral":
  print(f"{highest_ranked_label} sentiment")
  return np.round(highest_ranked_score, 4) # round the output to 4 decimal places
elif highest_ranked_label == "positive":
  print(f"{highest_ranked_label} sentiment")
  return 2*np.round(highest_ranked_score, 4) # round the output to 4 decimal places
```

Figure 7.6: Implementation code segment: Tweet Sentiment Analysis *(self-composed)*

The output of the Sentiment analysis model was returned as negative for negative sentiment, as it is for neutral sentiment & multiplied by 2 for positive sentiment to make an impact on the rankings of the output produced.

```
pre_processed_twitter_trends = []

bag_of_trends_phrases = []
min_tweet_volume = None

for trend in twitter_trends:
  if trend['name'] not in bag_of_trends_phrases :
    # ignore duplicates (twitter API bug? sometimes trends are duplicated)
    bag_of_trends_phrases.append(trend['name'])

    pre_processed_trend = trend

    #  remove hashtags
    if trend['name'][0:1] == '#':
      # remove hashtag
      pre_processed_trend['name'] = trend['name'][1:]

    # update min_tweet_volume
    if trend['tweet_volume']:
      if min_tweet_volume == None:
        # first trend which has a tweet_volume
        min_tweet_volume = trend['tweet_volume']
      elif trend['tweet_volume'] < min_tweet_volume:
        # update min_tweet_volume
        min_tweet_volume = trend['tweet_volume']

    #  convert name to lower case
    pre_processed_trend['name'] = pre_processed_trend['name'].lower()

    pre_processed_twitter_trends.append(pre_processed_trend)
```

Figure 7.7: Implementation code segment: Preprocess Trends Data *(self-composed)*

The code segment in Fig 7.7 preprcoesses trends that are fetched from the live Twitter API.

The code segment in Fig 4 of *Appendix D - Implementation* 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 code segment in Fig 7.8 is used to calculate the trends score for each NFT and finally make trends-based recommendations.

```
def calculate_trend_score(date_diff, trend_volume, median_tweet_volume, sentiment):
  mu = 0.1 # constant

  trend_impact_score = (trend_volume/ median_tweet_volume)

  trend_score = sentiment * trend_impact_score/ (mu + date_diff)
  return trend_score
```

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

7.4 Testing & Evaluation Code of Models

All code snippets that were created for Testing & Evaluation purposes have been mentioned under *Appendix D - Implementation*

```
current_date = datetime.now().date()

def calculate_trend_score_for_all_trends(curr_date = current_date):
  for trend in pre_processed_twitter_trends:
    volume = trend['tweet_volume']
    # trend_impact_score = (volume/ median_tweet_volume)

    trend_datetime = trend['created_datetime']
    date_diff = get_date_diff(trend_datetime.date(), curr_date)

    sentiment = 1 # will be taken 1 (which is in between neutral and positive) if sentiment hasn't been calculated

    if 'sentiment_score' in trend:
      sentiment = trend['sentiment_score']

    trend_score = calculate_trend_score(date_diff, volume, median_tweet_volume, sentiment)
    trend['trend_score'] = trend_score

calculate_trend_score_for_all_trends()
```

Figure 7.9: Implementation code segment: Calculating All Trends Scores *(self-composed)*

7.5 User Interface

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

7.6 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. the UIs expectable in the Minimum Viable Porduct of the project have been presented.

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

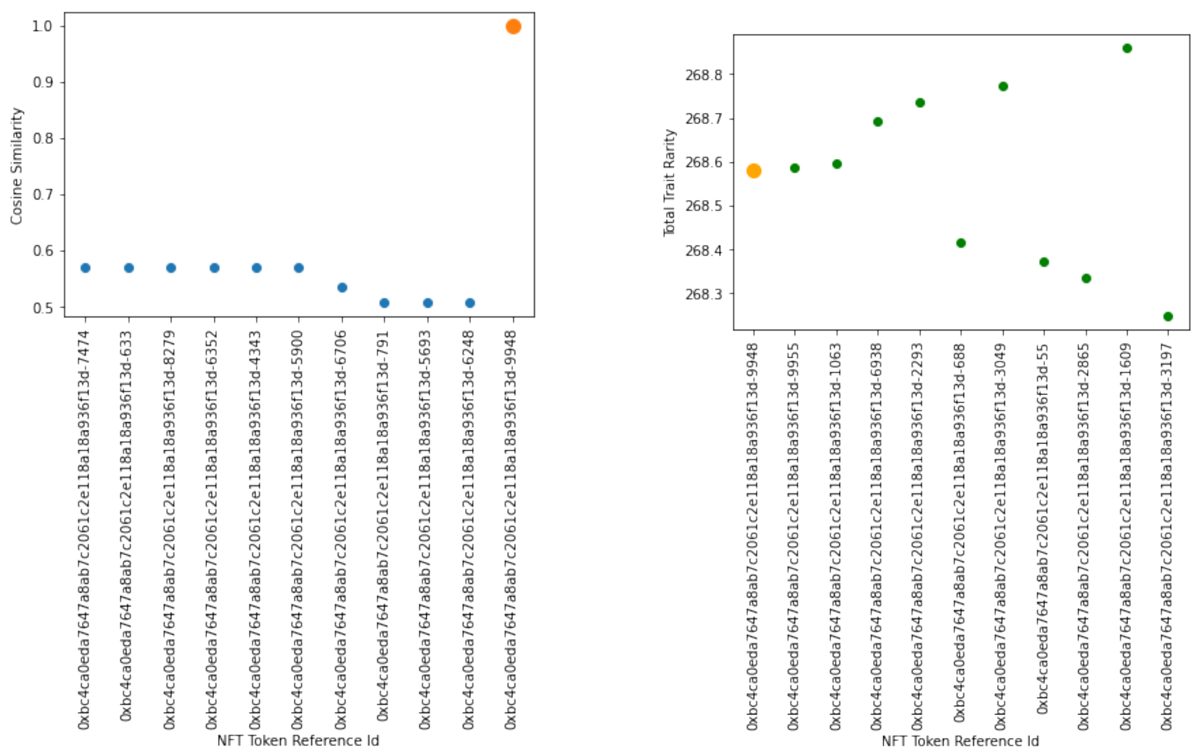
The multiple ensemble models developed in the project were tested based on the following conditions.

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 have the highest trend-score will be recommended and visualize how the scores change over time.

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

Testing this model was a bit tricky as there was no ground truth that the output could be compared with. Instead, the behavior of the algorithm was tested. The heatmap in Fig 8.2 shows how the trend-score for items decreases with time, from the date of matching with the trend. Additional heatmaps with detailed outputs have been placed under *Appendix E - Testing*.

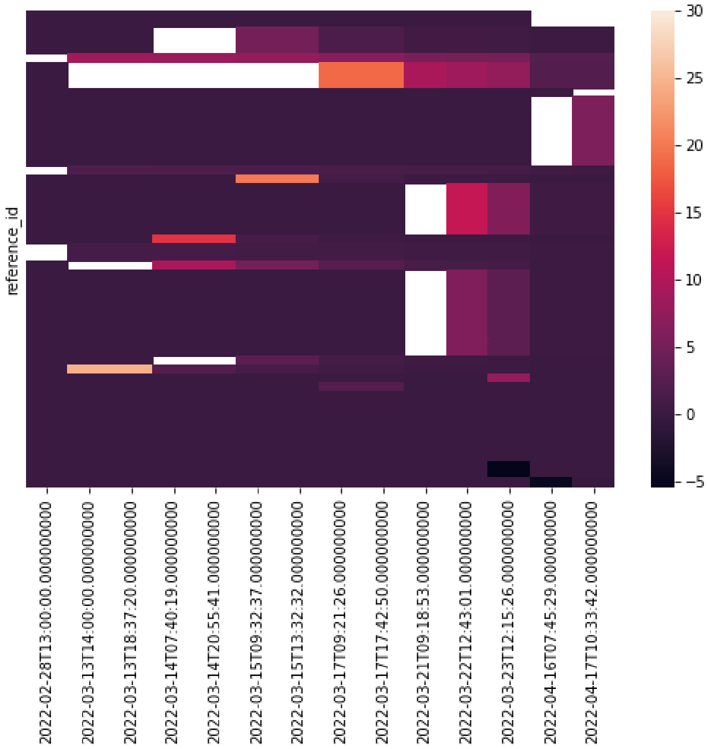


Figure 8.2: Trends based Recommender - Trend Score Heatmap *(self-composed)*

High impact trends staying relevant for a longer period of time, sometimes even better than those matched on the same day was an interesting revelation made from the generated heatmap.

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.

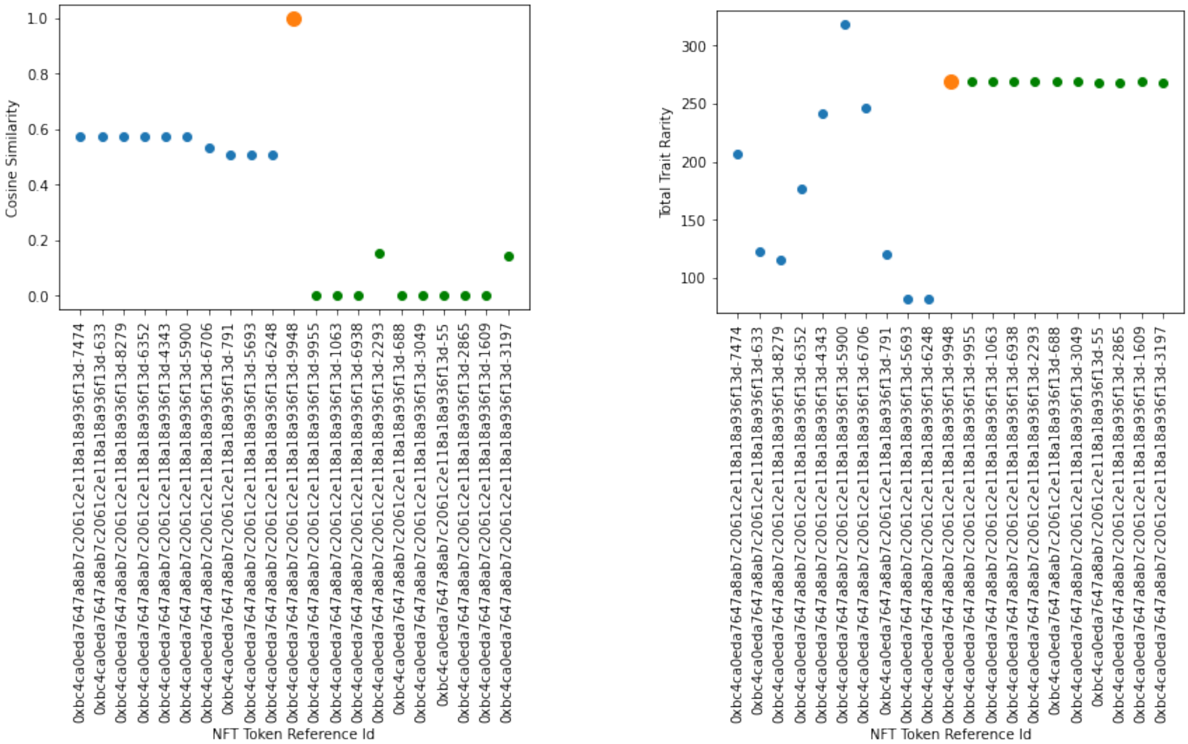
These were calculated using the below formulae.

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

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

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 graphs in Fig 8.3 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 following graph shows the count of the items that were matched with the trends of each datetime, highlighting the counts of newly matched items.

It is also important to note that the rankings of these items that are recommended are updated each time a new set of trends are entered into the system (this is an automated process) based on the trend score that is calculated.

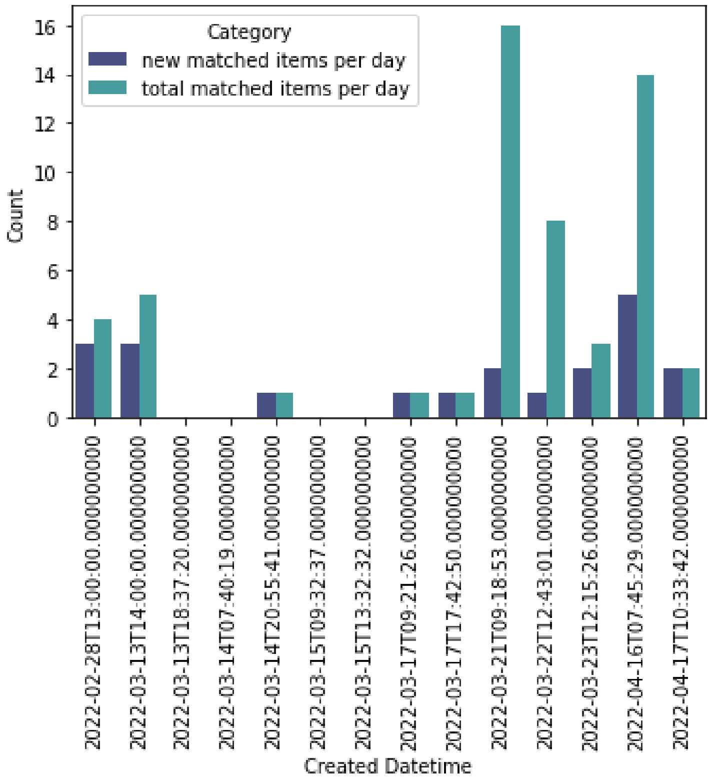


Figure 8.4: Evaluation of Trends based Recommender *(self-composed)*

8.5 Benchmarking

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

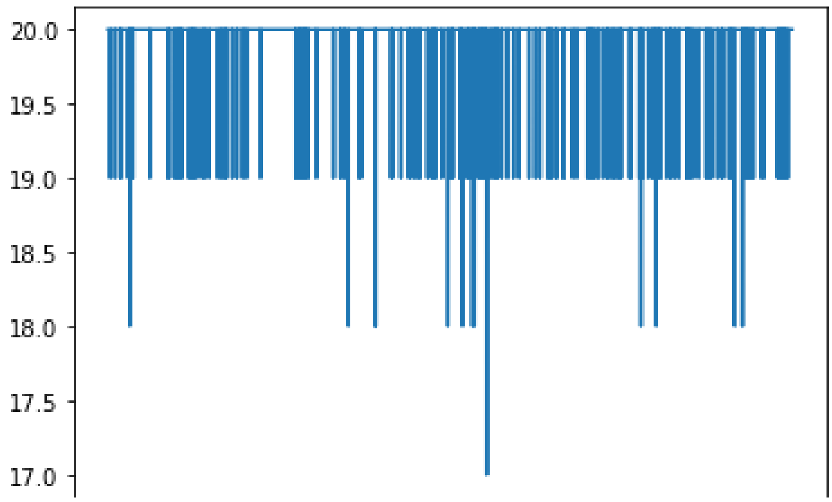


Figure 8.5: Aggregate diversity of generated Recommendations *(self-composed)*

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

The uniqueness & novel approach taken in the Trends-based Recommendation System (RecSys) meant that it couldn't be benchmarked and compared with any other existing model.

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 E - 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 the required information and calculates rarity score	Filtered out the 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

The table, *Evaluation of the implementation of Non-functional requirements* of **Appendix F - Evaluations** ceritifies that all Non-functional requirements had been met, resulting in a completition percentage of 100%.

8.9 Limitations of Testing Process

As an initial study of recommending NFTs, it was difficult to pinpoint 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 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 were 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, and 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.
Possible improvements	To unveil possible improvements that could be worked on as Future Work related to the conducted 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 to 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 meager amount of data. Cutting-edge languages & tools have also been used in the process. In the requirements of a Data-scientist job at OpenSea, which was posted on LinkedIn shown in *Appendix F - Evaluations* emphasizes the value and requirement of the approach taken to recommend items using social media trends & sentiment.
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.
Possible improvements	After getting expert & domain evaluators' feedback, possible improvements that could be addressed were attempted. Especially a more comprehensive quantitative evaluation of the trends-based model.
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, DApp (Decentralized App) Development & NFTs.
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 analyzed 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	Recommendation Systems choice gap	The study of Recommendation Systems is valuable impactful, especially in an unexplored e-commerce domain that is new and has had high trading volumes.
		Technical research gap	Due to the nature of NFTs, it was good not to stick to the standard collaborative filtering method. This opened up a good research area.
	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. Nothing like what the author has attempted in his research has been attempted before. It's interesting.
Research contribution	1	Technical Contribution towards Recommendation Systems	Innovative methods of solving the research gap have been identified. The clear gap & issue with traditional Recommendation methods have been addressed.
	2	Domain Contribution	The contribution is good because there's no other system like this and gathering data is difficult.
	3	Domain Contribution	It's a major contribution towards the domain. The author has been successful at unveiling impactful recommendation techniques & features considered.

Quality of research documentation	1	Content presentation of content	The use of latex was immediately noticed and commended. In-depth research has been conducted with the presentation of statistics.
	1, 2, 3	Approach Taken to achieve the solution to the problem	Many angles have been considered to approach the solution. A scientific approach has been taken.
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.
	1, 2	Selection of the Sentiment Analysis model	The selection use of the output of the Sentiment Analysis model has been well-thought-out and justified.
	3	Development approach	Everyone agreed that it was a good direction to take and a straightforward and methodical approach has been taken, without jumping directly to model construction.
Quantitative analysis of results	1, 2	Analysis of the social trends based RecSys	The current evaluation method makes sense and is clearly understandable. Could try to scrape the internet/ OpenSea to validate generated recommendations from the social trends RecSys. Could try to synthesize demonstrate results from the social trends RecSys to show why it's needed. Technical aspects have been well-evaluated. it would be good if the researcher get feedback from people based on the recommendations produced. Model accuracy could've been emphasized.

	1, 2, 3	Analysis of trait-based RecSys	The graphical analysis of the models is very clear. Looks like the best way to evaluate these models.
Possible improvements	1	Additionally considerable parameters	Consider the price for recommendations.
	2	Analysis of the social trends based RecSys	Could have been evaluated by different parameters such as country, age, domain(art, game NFT), etc. Consider NFT utilities, buy and sell trend, no of bids, and visibility as well to identify the top NFTs.
	3	Credibility of source	Identify artists by tracking their activity to recommend items by credible artists. Identifying fraudulent NFTs would also be interesting.
Usability, UI/ UX of MVP	1	Requirement of a separate application	Since the prototype produces clear results, a separate application is not required.
	3	Present descriptions of recommendations	Would've been able to better understand the results in comparison with the reference NFT if the details of them were displayed.

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 set up 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 F - 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 F - 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 the reasoning of choosing each method. The criteria for evaluation were defined prior to the author's self-evaluation & feedback from evaluators. The opinions received from 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

10.1 Chapter Overview

This chapter brings the thesis of the covered research to a conclusion by marking the concluding statements of the project. The project's unique contribution to the research community is discussed in line with the project aims & objectives. The challenges that were encountered, how the author's prior knowledge and the modules of the degree program were used, and the new knowledge and skills developed are documented.

10.2 Achievement of Research Aim & Objectives

10.2.1 Achievement of Aims

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.

The aim of the research was successfully achieved by designing, developing & evaluating a novel Recommendations Architecture to produce relevant, trending, and timely. Multiple steps of the process were automated to meet the requirement. Furthermore, research was conducted to identify what models could be used to recommend worthy NFTs.

10.2.2 Achievement of Objectives

The achievement of the objectives of the research that were mentioned in Chapter 1 has been marked with each of their completion statuses in the Completion Status of Research Objectives table of *Appendix G - Conclusion*.

10.3 Utilization of Knowledge from the Course

Table 10.1: Utilization of knowledge gained from the course

Module(s)	**Utilized Knowledge**
Software Development Group Project	From recognizing an issue to designing, developing, and testing a prototype, this module gave the initial spark to work on research & publish research.
Algorithms: Theory Design and Implementation, Applied AI	The knowledge gained from these modules were extremely important when designing new performant algorithms in this research. They also gave quite a lot of knowledge that was used for data science model development.

Programming Principles 1, 2 & Object-Oriented Programming	These modules laid the foundation for programming design used for documentation & coding concepts used in the development of the project.
Enterprise Application Development	Gave a thorough understanding of design documentation & standard that need to be maintained in Enterprise applications.
Web Design and Development	The prototype was made with UI/ UX guidelines taught in this module. The foundation laid by HTML, CSS & JavaScript was also helpful as a foundation to go beyond and build performant & advanced UIs.

10.4 Use of Existing Skills

- **Fullstack R&D** - The author completed his internship at Zone24x7 where he got to work on full-stack development of R&D Big Data projects while getting exposed to cutting-edge technologies.
- **Blockchain** - The research required quite a lot of understanding in the domain of Blockchain since NFTs are one such application of Blockchain technology. It was important to understand people's thinking patterns & in decision-making steps to come up with automation tactics required to produce recommendations. The knowledge for this was gained by the author's involvement in Blockchain & decentralized systems projects at Niftron, the Blockchain Research Group of IIT which was done in collaboration with 99x technology on Blockchain projects& personal reading.
- **ML/ DL** - The author self-learned basic ML & DL before the start of the final year by watching tutorial videos on Youtube & Coursera.

10.5 Use of New Skills

- **Recommendation Systems** - The author had no prior experience working with Recommendation Systems. Multiple methods of building Recommendation Systems were learned using online freely available material such as Google ML courses, Coursera, Medium, Kaggle & Youtube.
- **Data Engineering/ Data Mining & Information Retrieval** - The author had to learn rigorous data mining & information retrieval techniques that were required to find the most important information that could be used for the solution since the quality of pre-processing would directly affect the outputs produced.

- **NLP** - Natural Language Processing was used heavily for data extraction & matching, especially in the pre-processing steps. Additional tutorials & blogs were followed to understand these concepts.
- **LaTeX**- Used for clear documentation of all research documents for professional typesetting of lasting value. This includes the project proposal, PSPD, Thesis, 2 research papers & 1 review paper.

10.6 Achievement of Learning Outcomes (LOs)

The achievement of the objectives of the research that were mentioned in Chapter 1 has been marked with each of their completion statuses in the Achievement of Learning Outcomes table of *Appendix G - Conclusion*.

10.7 Problems and Challenges Faced

Table 10.2: Mitigations to Problems and Challenges Faced

Problem/ Challenge	**Mitigation**
Low battery & no internet connectivity were caused due to long hours of power cuts for more than one month towards the end of the project.	Worked overnight, at co-working spaces and bought a UPS to power the Wifi router and required peripherals.
Since the NFT domain is very new & there wasn't any research done on NFTs before starting the research, it was extremely difficult to find evaluators for the project. Those who were able to evaluate the project without much difficulty had a comparatively low amount of paper qualifications/ prior research experience.	The author had to explore the domain quite a lot. Evaluations were taken from multiple perspectives. Domain & project evaluations from domain experts & enthusiasts who were quite young & senior researchers who could evaluate the project from the research process & ML perspectives.
Lack of NFT & trends data & rate-limited APIs with API-keys.	Scripts were written with time-outs to fetch & preprocess data. API keys were requested and received after multiple requests from OpenSea & Twitter.

Testing & evaluating the suggested models was challenging since there was no ground truth to evaluate the models against. Especially, the trends-based model. Testing & evaluating RecSys models have been known to be unclear/ challenging even based on Literature.	Modified testing & evaluation were conducted for possible models, while algorithmic testing & evaluation was conducted for the Trends-based model.
The novelty of the research & specificities of the domain made it pretty much impossible to benchmark the system against an existing solution.	2 of the newly created ML models were benchmarked against each other.
Since this project was one of the very first of its kind, there was a very limited number of scientific documents, data & a clear direction to follow.	A direction of interest was chosen by the author based on the requirements gathered from surveys, interviews & remotely related LR.

10.8 Deviations

The initial goal of the author was to integrate price prediction into the system at least in an insignificant manner, but the available data was not sufficient for this purpose and the amount of time & effort required to fetch & pre-process data made it clear that it would be required to be done as a separate project.

After considering possible DL methods to build a Recommendations Model, it was understood that the amount of data available was not sufficient to attempt it. Algorithmic ML models were created instead.

10.9 Limitations of the Research

- The limited time allocated for the research and the need to spend a lot of time mining & pre-processing data constrained the author from working on price prediction and considering the trading & value aspect of NFTs for recommendations.
- The trends-based model is only effective if accurate, diverse descriptions are provided for NFTs.
- Personalized recommendations cannot be made with the current trends-based model.
- The words in hashtags aren't split although received as trends. This decreases the possibility of matching with keywords of items.

- The system works in a centralized manner.
- The current system may struggle with large-scale system & data since pure Python & Pandas are used for the data science component.
- The current prototype support only Twitter Trends, but can be expanded to use trends from other social platforms.

10.10 Future Enhancements

- Identify possibilities of sourcing trends from more platforms such as Reddit, Discord, Google & private forums.
- Attempt to create a decentralized Recommendations Eco-system using the Trends-based RecSys model since the trends and items can come from two different sources.
- The current solution does a string match with the keywords of each item. This may cause some matches to be skipped due to appearing in different forms. The NLP technique, lemmatization could be a possible solution for this. Name Entity Recognition is another NLP technique that could enhance the quality of trends data used. The significance of introducing such techniques will have to be tested since they may not have a significant impact on the output as most trends appear to be nouns.
- One of the short-comings to help match trends for this purpose is that Twitter trends contain hashtags as trend names at times. Either the developers from the end of Twitter could give a possible solution to it or hashtags may have to be pre-processed and separated.
- Work on a price-prediction model for NFTs. This may be extremely difficult due to the uniqueness of NFTs and due to the low amount of available data. A suggestion that I received for this was to attempt using a dataset of rare-physical artwork since they tend to resemble the nature of NFT pricing. *Closer to the completion of this research, the author came across an NFT dataset (ZOMGLINGS and Avery Smith, 2021) that may be usable for price/ bid prediction training purposes.*
- As a substitute or addition to recently released movies, Amazon's DL Neural Network Model could make use of trends, maybe to bolster recommendations for movies as well as e-commerce items. Due to the lack of NFT data, this DL-based approach could not be attempted.
- The trends could be categorized to identify similar trends that users seem to show interest in. It would be almost impossible to attempt this level of personalization without collecting user data. Therefore, the value of such an attempt may have to be justified.

10.11 Achievement of the Contribution to Body of Knowledge

By concluding the research project, the author has managed to make contributions in the domain of NFTs, the technology of Recommendation Systems, and towards the research process.

10.11.1 Technical Contribution (Recommendations Systems)

1. Social Trends influenced Recommendation Model - a novel & innovative concept & approach taken.
2. Trends-score calculation equation & algorithm

Currently, no Sri Lankan-based e-commerce site currently provides recommendations. The introduced Trends-based RecSys will help generate timely & trending recommendations without having to collect & store large amounts of user data.

10.11.2 Domain Contribution (NFTs)

Identification & analysis of factors that can be used to produce NFT item recommendations.

10.11.3 Additional Contributions

1. Data Preprocessing scripts
   (a) Social media trends extraction
   (b) NFT Data extraction
      - Trait rarity calculation
2. NFT Asset datasets
3. Created a Latex template for the expected thesis structure that can be used by IIT students in the future (*Contributions made by Visal Rajapakse, Isala Piyarisi & Akassharjun Shanmugarajah*).

   The latex structures created & all project related documents can be found here: `https://github.com/dinuka-rp/nerdy-panda`

10.12 Concluding Remarks

This concludes a research that lays the groundwork for future NFT research projects & novel ways of generating recommendations using sparse data. The introduced algorithm has the potential to evolve into a much more complex & utilitarian Recommendation System which may be embraced by digital systems & internet of the next decade. The research process was conducted at the highest possible standards signifying the contributions made to the body of knowledge.

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).

ZOMGLINGS and Avery Smith (Sept. 25, 2021). *Ethereum NFTs | Kaggle*. Kaggle. URL: https://www.kaggle.com/datasets/simiotic/ethereum-nfts (visited on 04/28/2022).

APPENDIX A - CONCEPT MAP

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

APPENDIX B - GANTT CHART

Figure 2: Gantt Chart

APPENDIX C - DESIGN

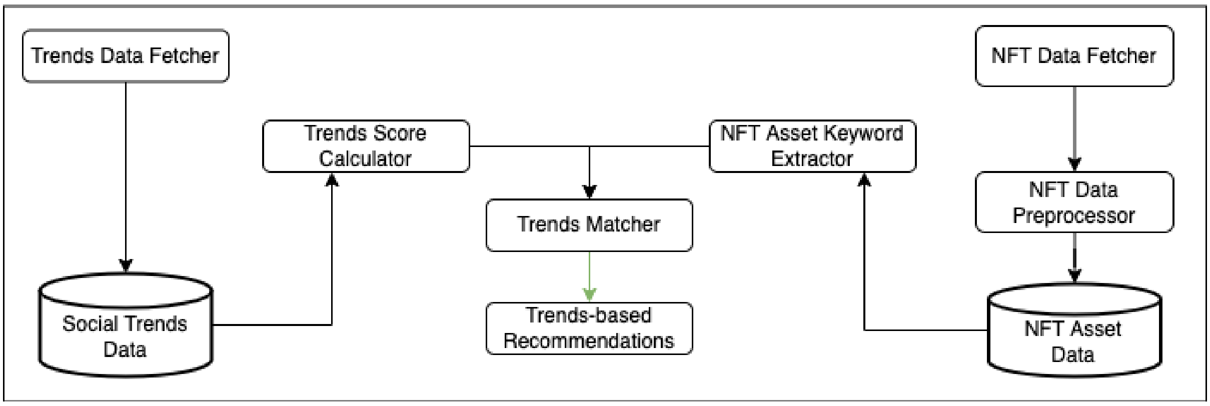


Figure 3: Trends based Recommendations Process Flowchart *(self-composed)*

APPENDIX D - IMPLEMENTATION

Appendix D1 - Implementation Code Snippets

```
# add min tweet volume for tweets with no volume
for index, trend in enumerate(pre_processed_twitter_trends):
  if trend['tweet_volume'] == None:
      pre_processed_twitter_trends[index]['tweet_volume'] = min_tweet_volume - 1


# print(pre_processed_twitter_trends)
# pp.pprint(pre_processed_twitter_trends)
```

Calculate Median Tweet volume

An impact score can be calculated separately as well.

```
[ ] import statistics

    # add all tweets with tweet volumes into an array
    tweet_volumes_array = []
    for tweet in pre_processed_twitter_trends:
      if tweet['tweet_volume'] != None:
        tweet_volumes_array.append(tweet['tweet_volume'])

    print("tweet_volumes_array:", tweet_volumes_array)

    # calculate median tweet volume
    median_tweet_volume = statistics.median(tweet_volumes_array)
    print(median_tweet_volume)
```

Figure 4: Implementation code segment: Tweet volume calculation *(self-composed)*

Appendix D2 - Testing & Evaluation Code Snippets

```
# for each nft from the matched nft df (NFTs that got matched with a trend-score)
# calculate the trend for each day from the day of the trend until the last created datetime of trends

max_trend_score = 0

for index, row in top_trending_nfts_df.iterrows():
  reference_id = index

  # take each matched trend in a row
  matched_trends = row['matched_trends']

  for matched_trend in matched_trends:
    # query for this trend in the all_trends_df -> get the date of this trend
    trend_data = all_trends_df[all_trends_df['name'].str.contains(matched_trend)]
    trend = trend_data.iloc[0]

    dt_ts = trend['created_datetime'] # datetime in Timestamp format is given
    # get date
    matched_trend_date = dt_ts.to_pydatetime().date()

  # for each day after the created_datetime of the trend, calculate trend score and add to a new df row with test_score & tested_datetime
  #  if the checking created_datetime is before the matched datetime, give 0 and add to a new df row.
    for dt64 in unique_created_dt64s:
      unique_test_date = pd.Timestamp(dt64).to_pydatetime().date()
      date_diff = get_date_diff(matched_trend_date, unique_test_date)
      # print(date_diff)
      if date_diff < 0:
        # score 0
        heatmap_df.loc[reference_id, dt64] = 0
      else:
        volume = trend['tweet_volume']

        sentiment = 1 # will be taken 1 (which is in between neutral and positive) if sentiment hasn't been calculated
        if 'sentiment_score' in trend:
          sentiment = trend['sentiment_score']

        # calculate trend score
        trend_score_of_day = calculate_trend_score(date_diff, volume, median_tweet_volume, sentiment)
        # update dataframe row
        heatmap_df.loc[reference_id, dt64] = round(trend_score_of_day, 2) # rounded to 2 decimals for readable representation in graph
        #  (Conclusion Note: *rounding 'causes the value to become 0 after a few days*)

        if trend_score_of_day > max_trend_score:
          # update max_trend_score
          max_trend_score = trend_score_of_day
          # print(max_trend_score)
          # print(trend['name'])

# print("FINAL max_trend_score:", max_trend_score)
heatmap_df
```

Figure 5: Testing code segment: Heatmap Generation *(self-composed)*

```
def calculate_precision_at_k(relevant_recommendations_count, recommended_count):
  return relevant_recommendations_count / recommended_count

def calculate_recall_at_k(relevant_recommendations_count, all_possible_relevant_recommendations_count):
  return relevant_recommendations_count / all_possible_relevant_recommendations_count

def calculate_f1_score_at_k(precision, recall):
  return 2 * (precision * recall) / (precision + recall)

# ---
def get_formatted_output_line(title, result):
  return '{:>12}  {:>12}'.format(title, result)
```

Figure 6: Evaluation code segment: Trait based RecSys *(self-composed)*

```
#  ----- evaluate trait content based recommender - self scored
print("----- Evaluation of Trait Content based Recommender (self-scored) -----")
relevant_recommendations_count = len(lists_of_trait_content_based_recommendations)
recommended_count = len(lists_of_trait_content_based_recommendations)
all_possible_relevant_recommendations_count = 10

precision_at_k = calculate_precision_at_k(relevant_recommendations_count, recommended_count)
recall_at_k = calculate_recall_at_k(relevant_recommendations_count, all_possible_relevant_recommendations_count)
f1_score_at_k = calculate_f1_score_at_k(precision_at_k, recall_at_k)

print(get_formatted_output_line("precision@k", precision_at_k))
print(get_formatted_output_line("recall@k", recall_at_k))
print(get_formatted_output_line("f1_score@k", round(f1_score_at_k, 2)))
print("\n")

#  ----- evaluate trait content based recommender - combined scored
print("----- Evaluation of Trait Content based Recommender (combined scored) -----")
relevant_recommendations_count = len(lists_of_trait_content_based_recommendations)
recommended_count = len(lists_of_trait_content_based_recommendations)
# all_possible_relevant_recommendations_count = 20
all_possible_relevant_recommendations_count = len(unique_set_of_all_recommendations)

precision_at_k = calculate_precision_at_k(relevant_recommendations_count, recommended_count)
recall_at_k = calculate_recall_at_k(relevant_recommendations_count, all_possible_relevant_recommendations_count)
f1_score_at_k = calculate_f1_score_at_k(precision_at_k, recall_at_k)

print(get_formatted_output_line("precision@k", precision_at_k))
print(get_formatted_output_line("recall@k", recall_at_k))
print(get_formatted_output_line("f1_score@k", round(f1_score_at_k, 2)))
print("\n")
```

Figure 7: Evaluation code segment: Trait based RecSys *(self-composed)*

```
----- Evaluation of Trait Content based Recommender (self-scored) -----
 precision@k          1.0
    recall@k          1.0
  f1_score@k          1.0


----- Evaluation of Trait Content based Recommender (combined scored) -----
 precision@k          1.0
    recall@k          0.5
  f1_score@k         0.67


----- Evaluation of Trait Rarity based Recommender (self-scored) -----
 precision@k          1.0
    recall@k          1.0
  f1_score@k          1.0


----- Evaluation of Trait Rarity based Recommender (combined-scored) -----
 precision@k          1.0
    recall@k          0.5
  f1_score@k         0.67
```

The reason to consider combined score seperately is because both the recommenders acrtually give relevant recommendations, but we need to show a user the best possible recommendations at the top. Even if a weighted bias is used, we need data (that is not available at present) to define what weightage to give to each pipeline.

Therefore, it was decided to design the system in a way that allows a user to choose a relavant bias. This is updated for all users who don't choose a bias based on everyone else's selection.

Figure 8: Evaluation code segment: Trait based RecSys *(self-composed)*

```python
matched_trends_set = set() # used to record matched trends with items

# iterating through items dataframe with trend_scores (only these items will eventually get matched anyway)
for index, row in trending_df.iterrows():
  # get row['matched_trends']
  matched_trends_per_item = row['matched_trends']

  for matched_trend in matched_trends_per_item:
    # filter all_trends_df by matched_trends
    filtered_trends_df = all_trends_df[all_trends_df['name'] == matched_trend]
    # print(filtered_trends_df.head())

    # get created_datetime of matched_trend
    dt = str(filtered_trends_df.iloc[0]['created_datetime'].to_numpy()) # https://pandas.pydata.org/docs/reference/api/pandas.Timestamp.

    updated_eval_value_dict = eval_trends_count_dict[dt]

    total_matched_items_count = updated_eval_value_dict.get('total_matched_items_count')

    if total_matched_items_count:
      updated_eval_value_dict['total_matched_items_count'] = total_matched_items_count + 1
    else:
      updated_eval_value_dict['total_matched_items_count'] = 1

    if matched_trend not in matched_trends_set:
      new_matched_items_count = updated_eval_value_dict.get('new_matched_items_count')

      if new_matched_items_count:
        updated_eval_value_dict['new_matched_items_count'] = new_matched_items_count + 1
      else:
        updated_eval_value_dict['new_matched_items_count'] = 1

      matched_trends_set.add(matched_trend)

    # update eval_trends_count_dict for each created_datetime
    eval_trends_count_dict[dt] = updated_eval_value_dict

pp.pprint(eval_trends_count_dict)
```

Figure 9: Evaluation code segment: Trends Data Clustering *(self-composed)*

Appendix D3 - UI Wireframes

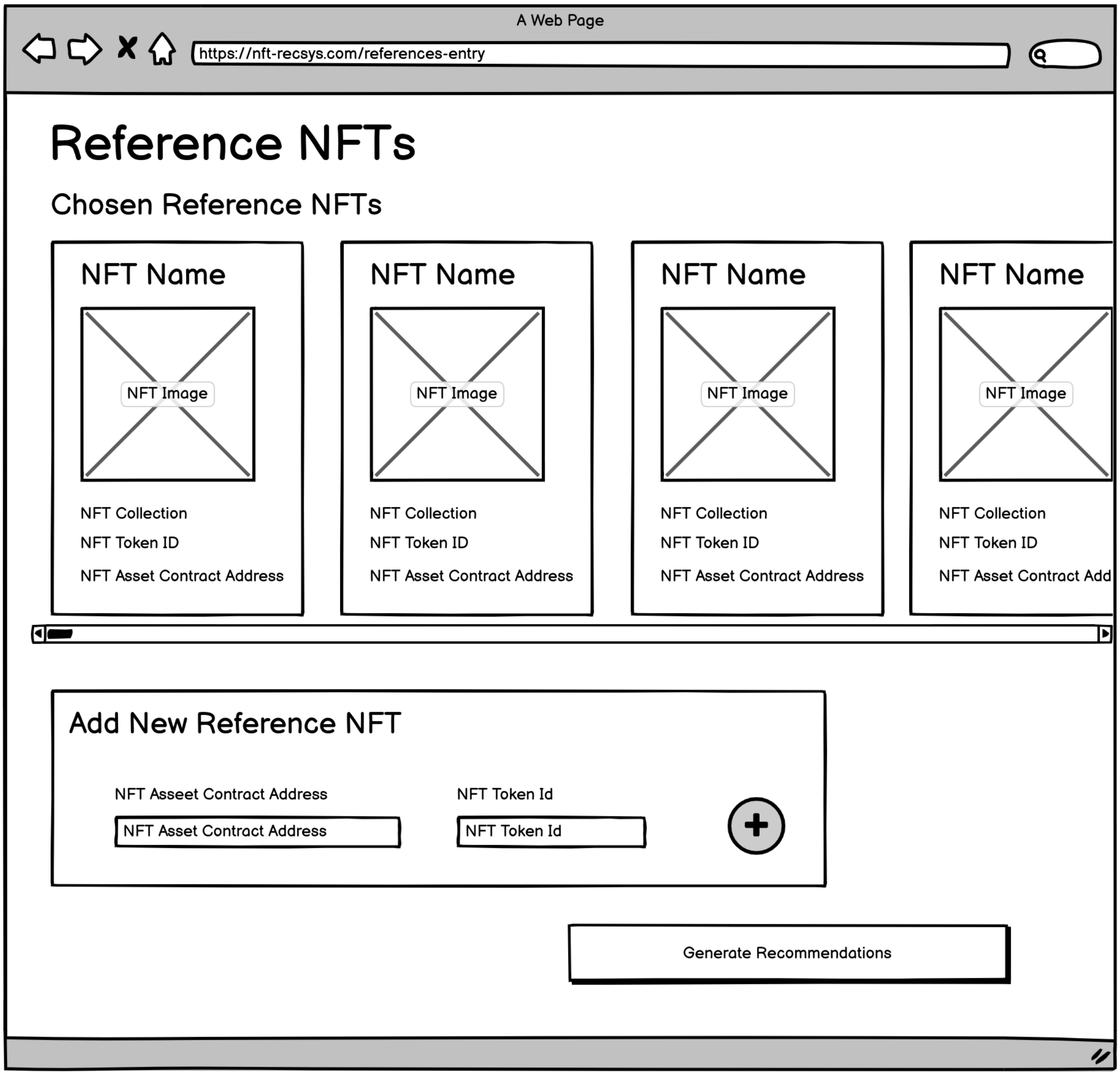


Figure 10: UI Wireframe - Reference NFT entry *(self-composed)*

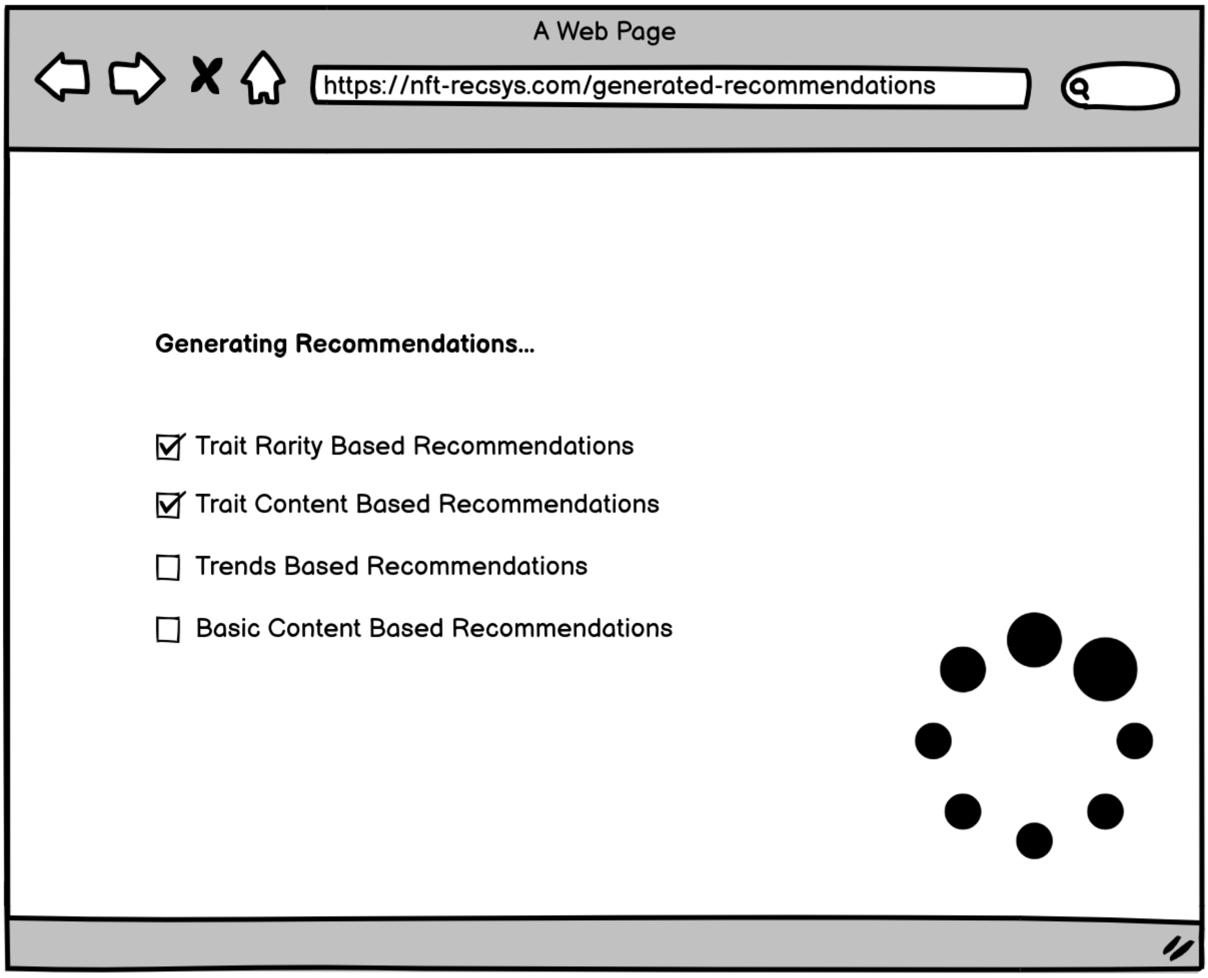


Figure 11: UI Wireframe - Loading-Generating Recommendations *(self-composed)*

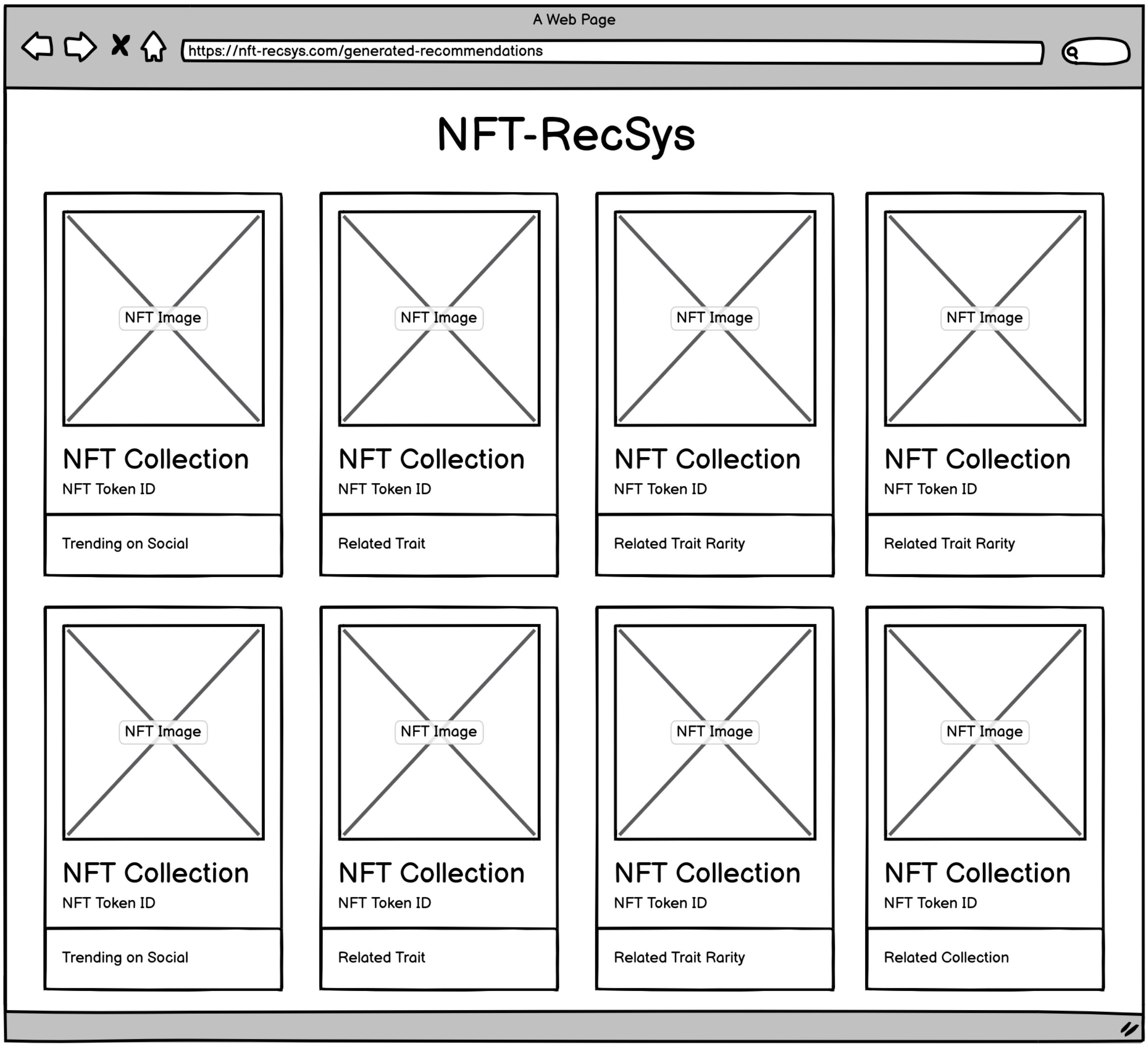


Figure 12: UI Wireframe - Generated Recommendations *(self-composed)*

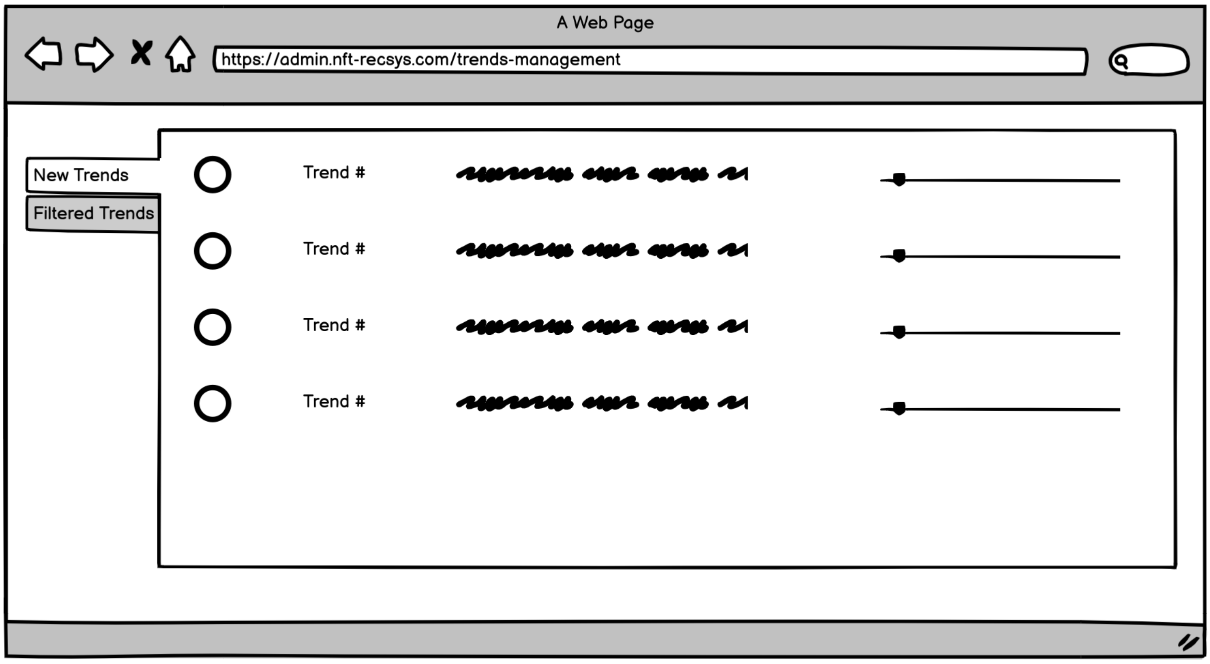


Figure 13: UI Wireframe - Admin Trends *(self-composed)*

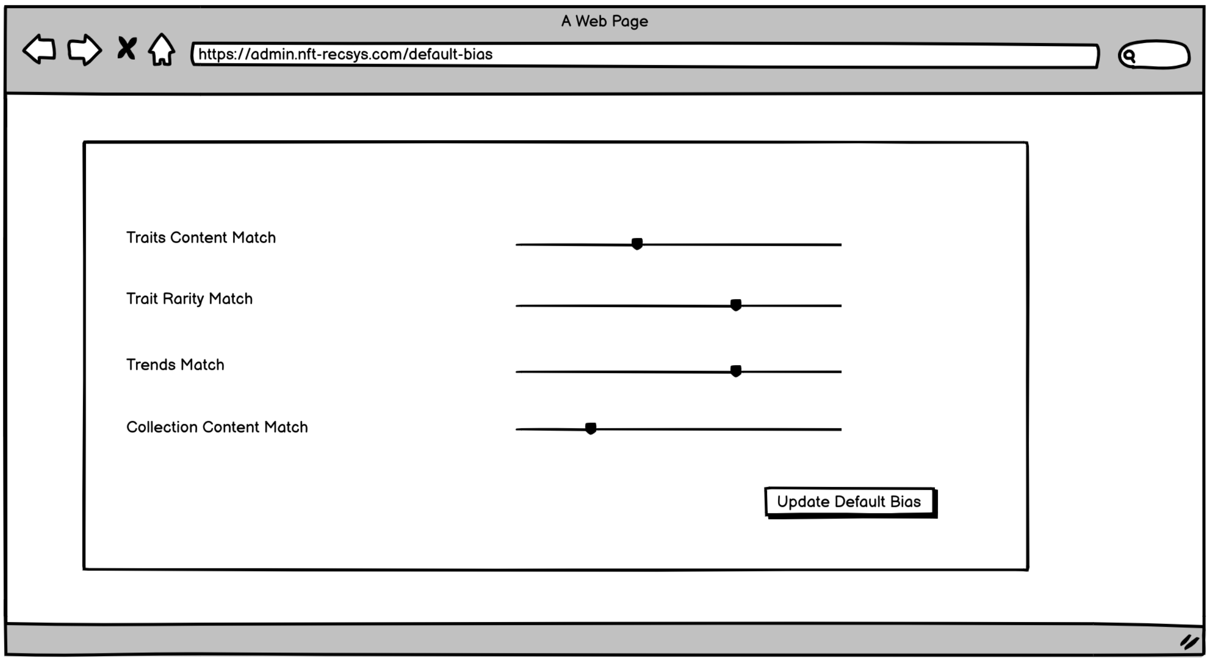


Figure 14: UI Wireframe - Admin Default Bias selection *(self-composed)*

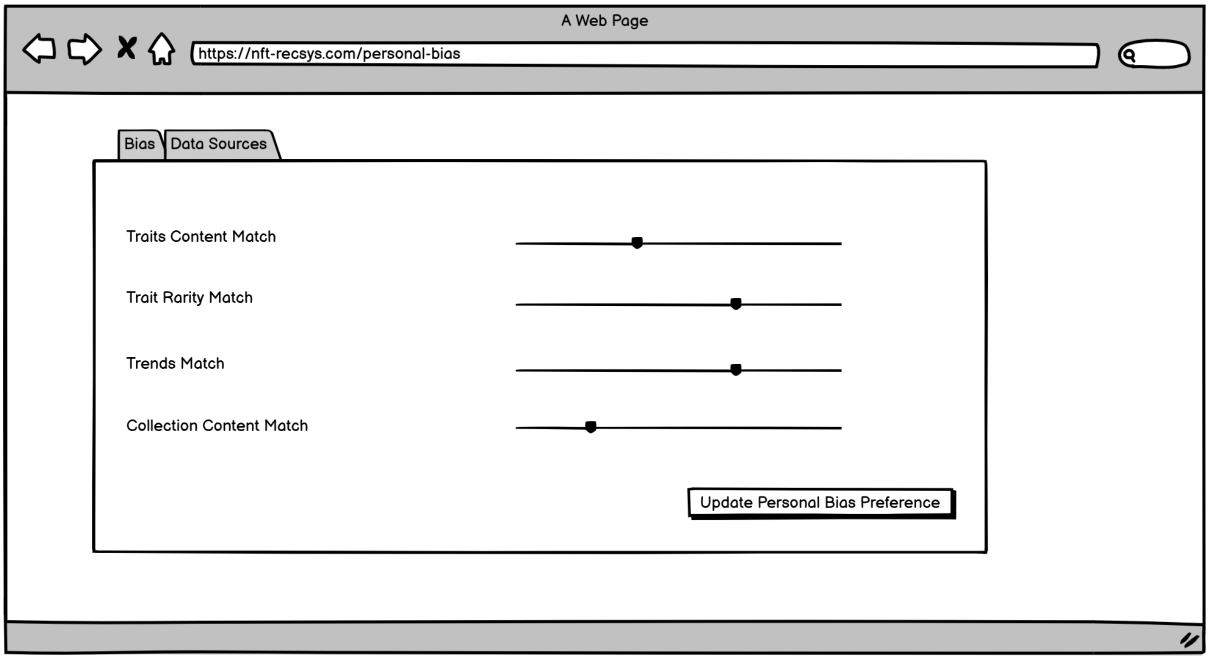


Figure 15: UI Wireframe - User Bias selection *(self-composed)*

Appendix D3 - Prototype UI

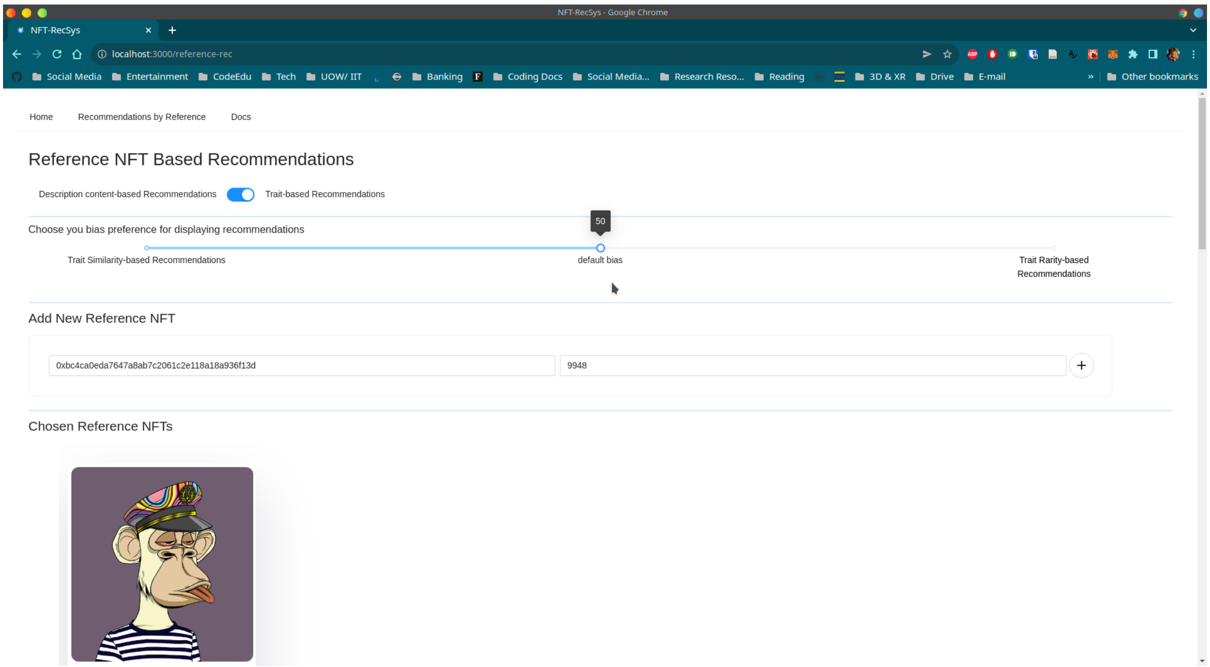


Figure 16: Prototype UI - Trait Based RecSys input *(self-composed)*

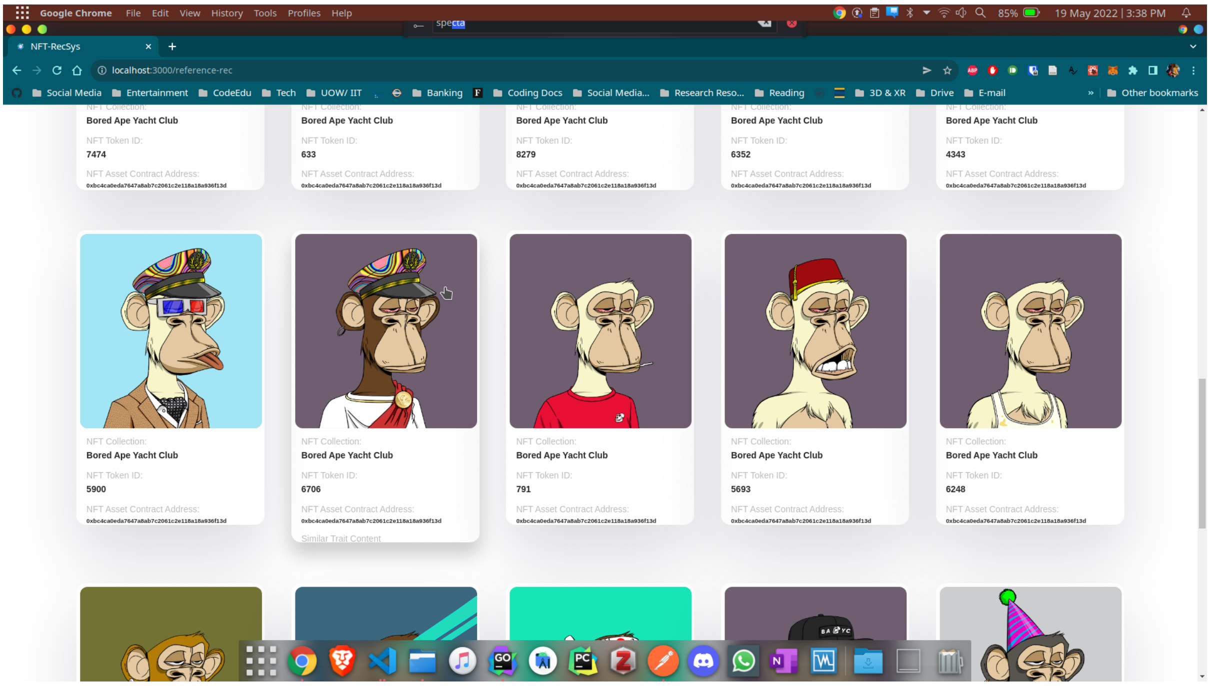


Figure 17: Prototype UI - Generated Trait-based Recommendations *(self-composed)*

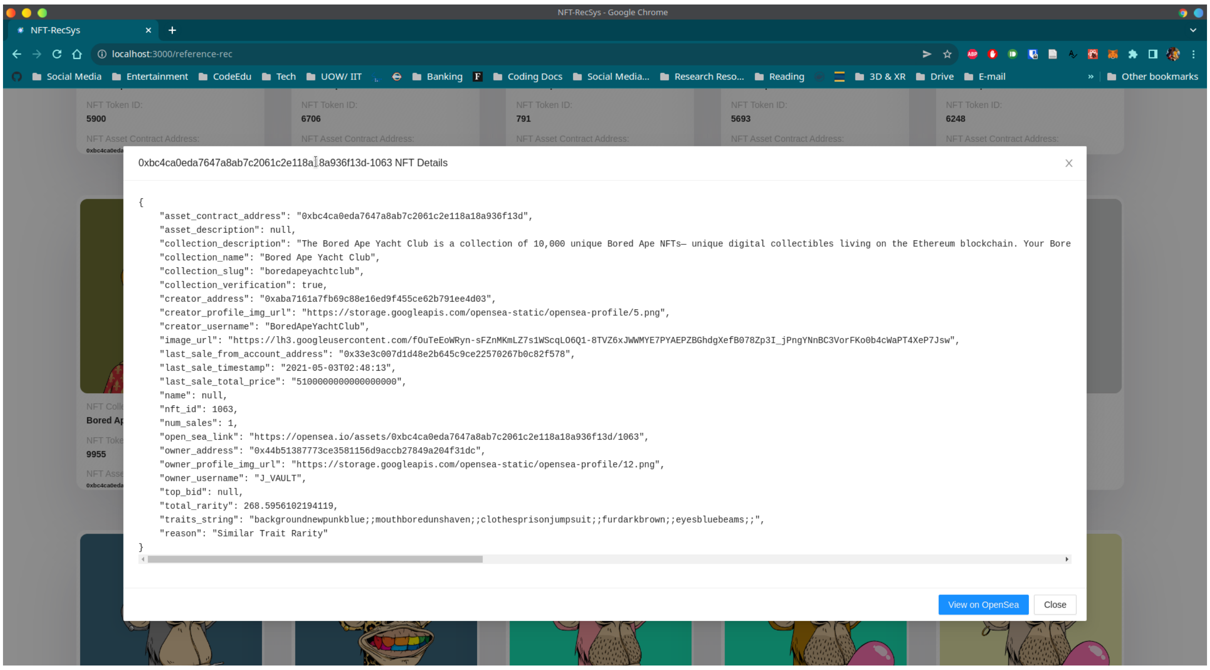


Figure 18: Prototype UI - Trait-based Recommendation - Item Details *(self-composed)*

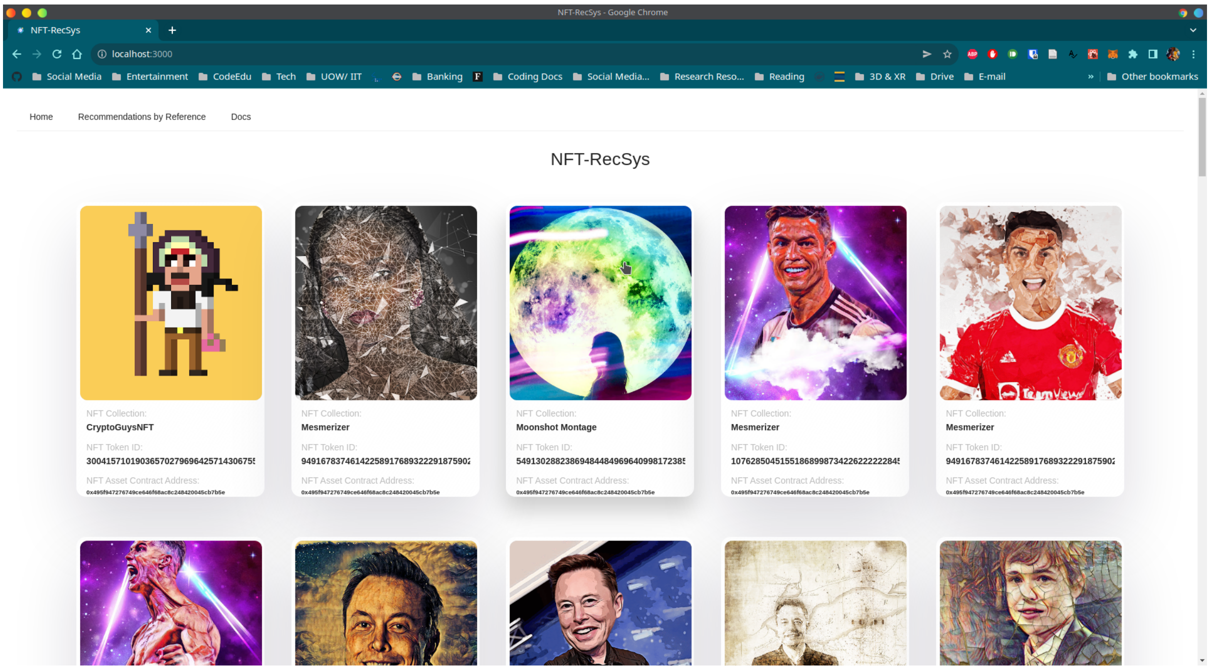


Figure 19: Prototype UI - Trends-based Recommendations - display featured items *(self-composed)*

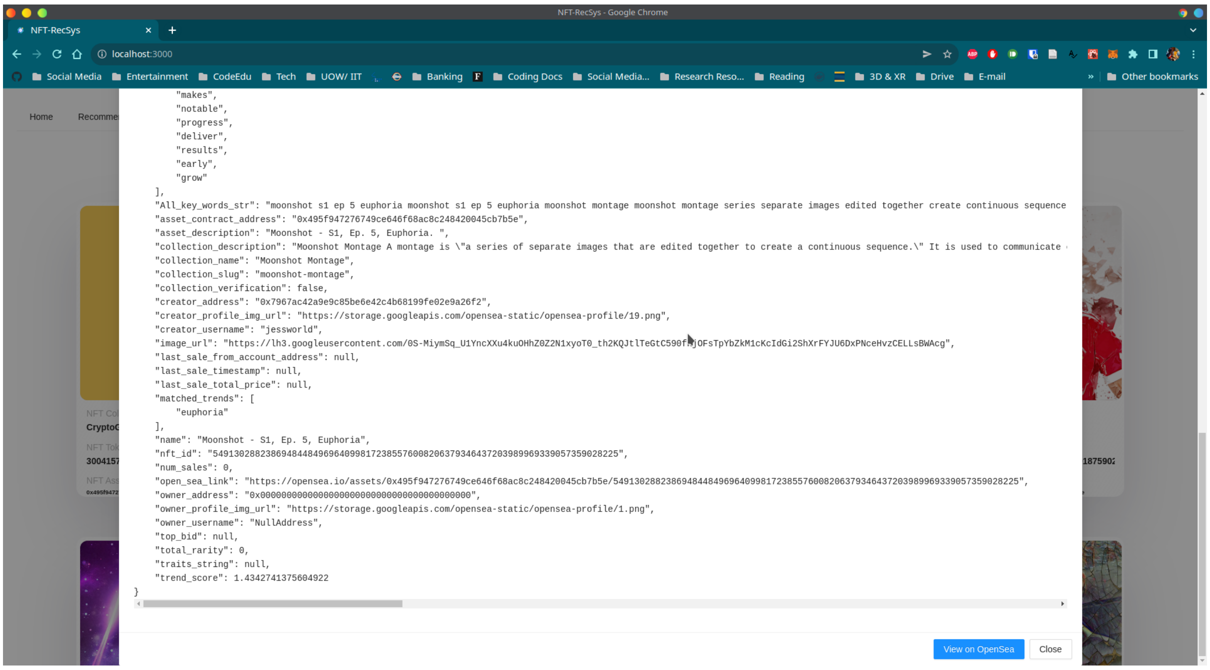


Figure 20: Prototype UI - Trends-based Recommendations - Item Details *(self-composed)*

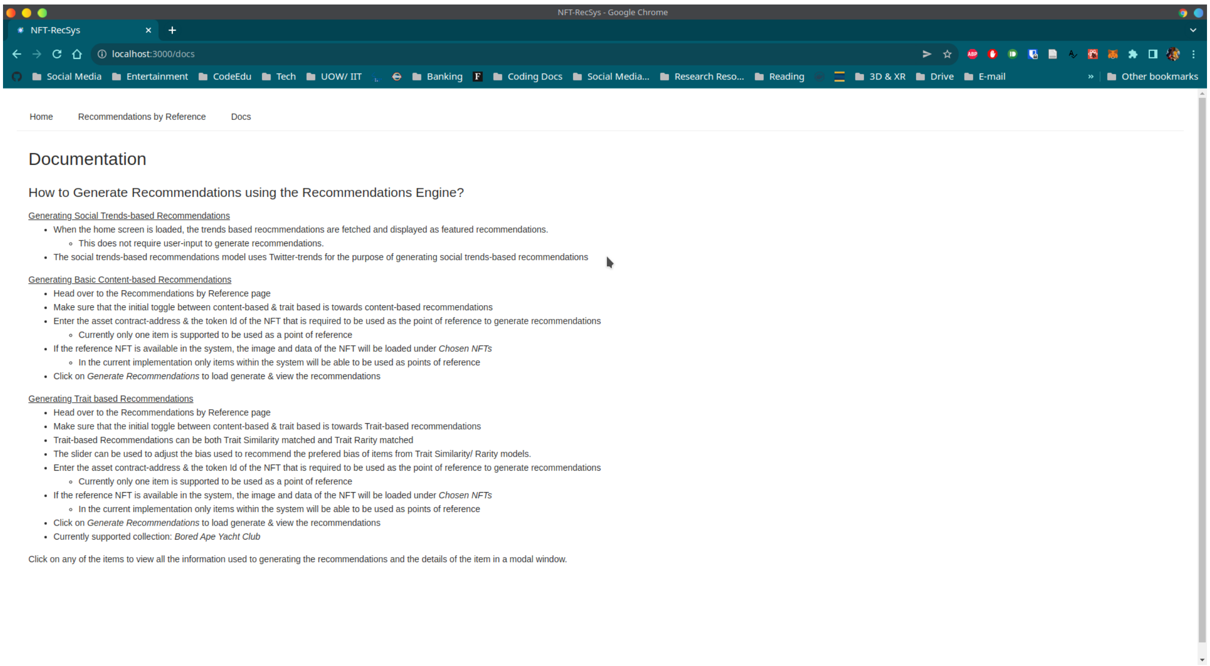


Figure 21: Prototype UI - Prototype Documentation *(self-composed)*

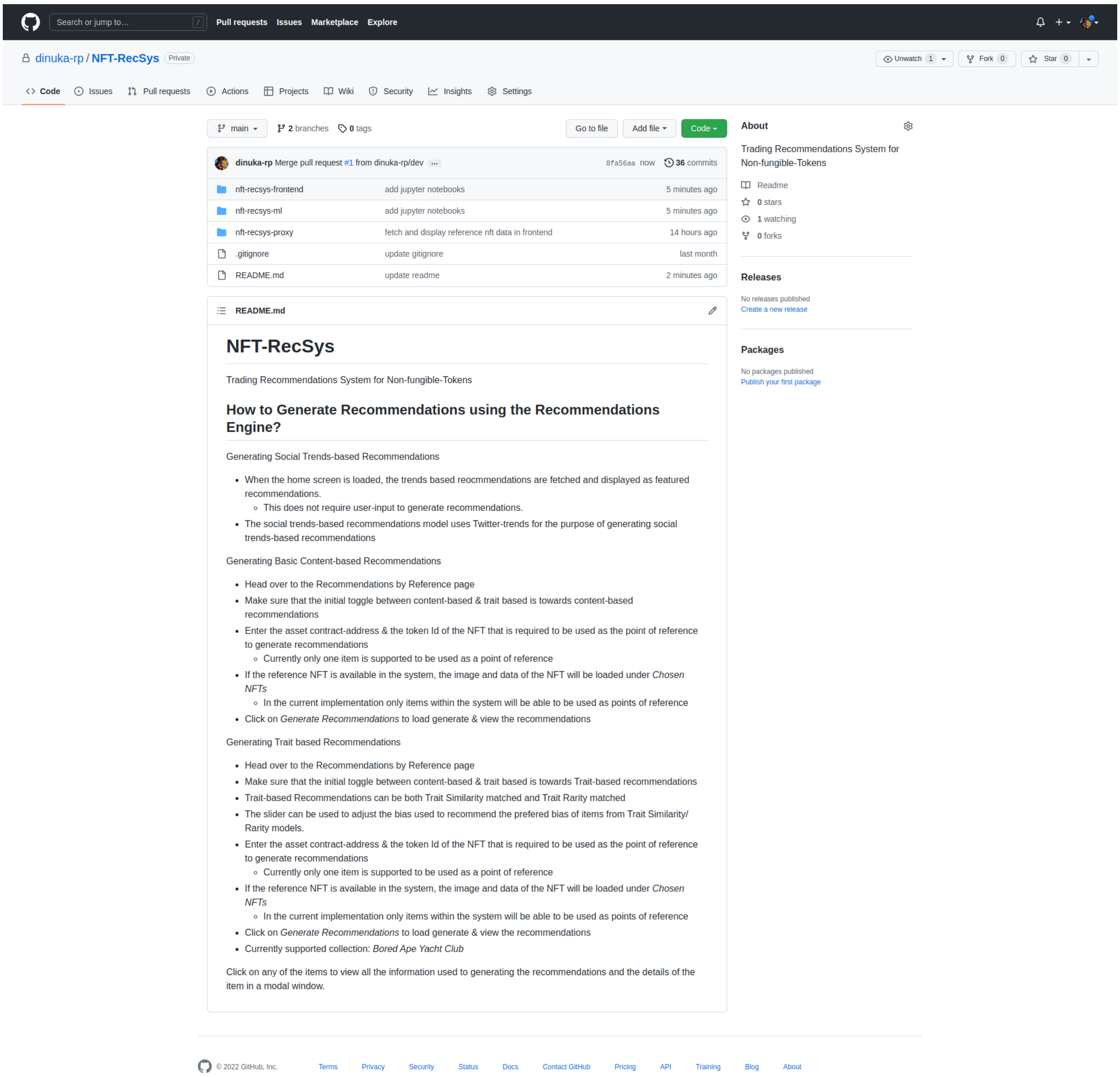


Figure 22: Prototype - GitHub Readme *(self-composed)*

APPENDIX E - TESTING

Appendix E1 - Model Testing

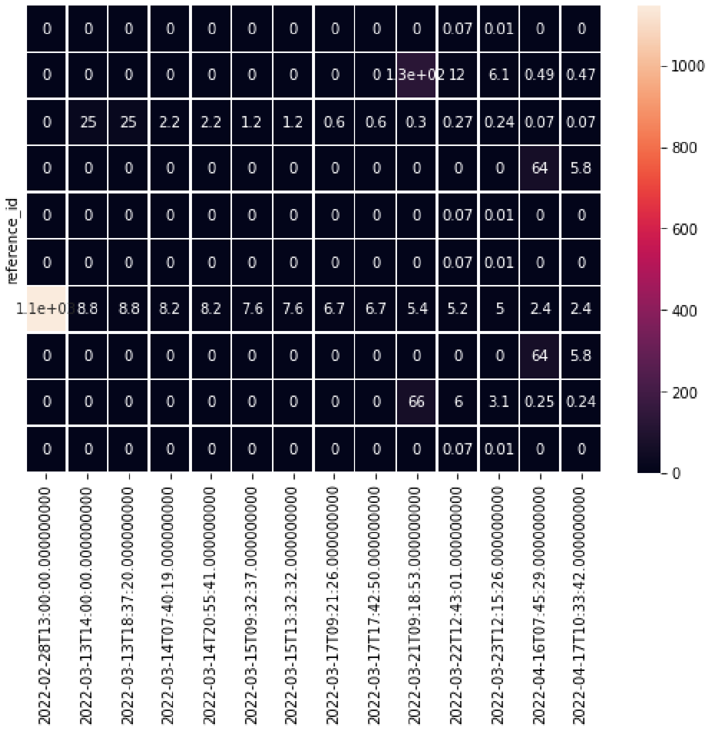


Figure 23: Trends based Recommender Testing Annotated Heatmap - 10 random items *(self-composed)*

Figure 24: Trends based Recommender Testing Heatmap - max score 30 *(self-composed)*

Figure 25: Trends based Recommender Testing Heatmap - max score 100 *(self-composed)*

Figure 26: Trends based Recommender Testing Heatmap - All items *(self-composed)*

```
top_trending_nfts_df = trends_based_recommendations()
top_trending_nfts_df.head()
```

reference_id	All_key_words_list	All_key_words_str	trend_score	matched_trends
0x495f947276749ce646f68ac8c248420045cb7b5e-30041571019036570279696425714306755064417936087066583674697121488131734896641	[jesus, simmons, cryptoguysnft, preparing, nft...	jesus simmons cryptoguysnft preparing nft game...	6.817898	[jesus]
0x495f947276749ce646f68ac8c248420045cb7b5e-94916783746142258917689322291875902080657954662840061252830161925292293095425	[triangles, 109, rihanna, mesmerizer, theme, t...	triangles 109 rihanna mesmerizer theme traits ...	4.530678	[rihanna]
0x495f947276749ce646f68ac8c248420045cb7b5e-110044771230368441551571992070040963530623495929530335175865263817154854125569	[map, art, 135, elon, musk, item, 135, officia...	map art 135 elon musk item 135 official map ar...	2.270546	[elon]
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263433004574310401	[midnight, art, 2, elon, musk, mesmerizer, the...	midnight art 2 elon musk mesmerizer theme trai...	2.270546	[elon]
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263434104085938177	[bobblehead, 3, elon, musk, mesmerizer, theme,...	bobblehead 3 elon musk mesmerizer theme traits...	2.270546	[elon]

Figure 27: Trends based Recommendations Example Output *(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-110044771230368441551571992070040963530623495929530335175865263817154854125569	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-74915313939230038050448181057740401670280602835803512556494204368644347700289	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-46813158413327833598722183941328804250293517675741658106197955405879632l9969	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 28: Trends based Recommendations Heatmap Data *(self-composed)*

Appendix E2 - Model Evaluation of Test Results

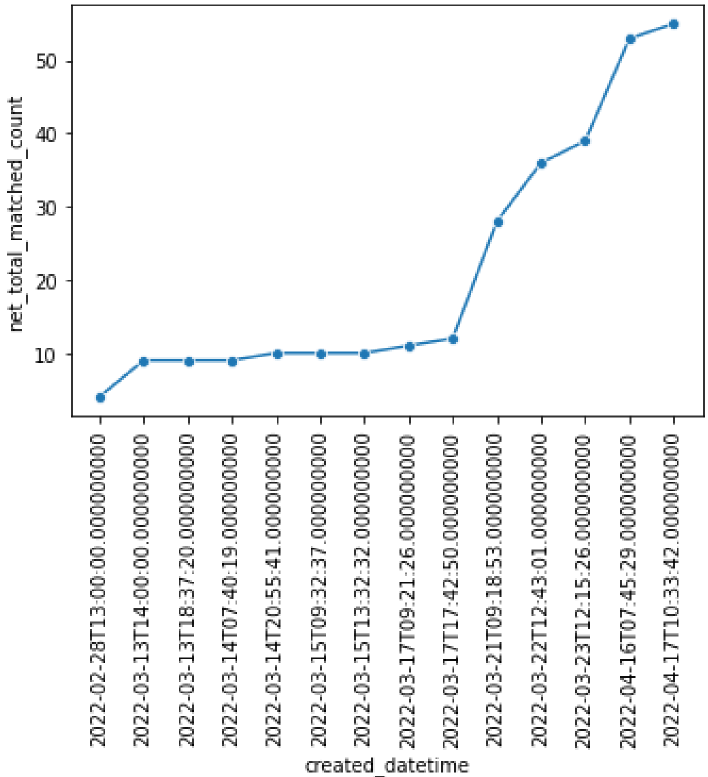


Figure 29: Total Trends based Recommendations made with time *(self-composed)*

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

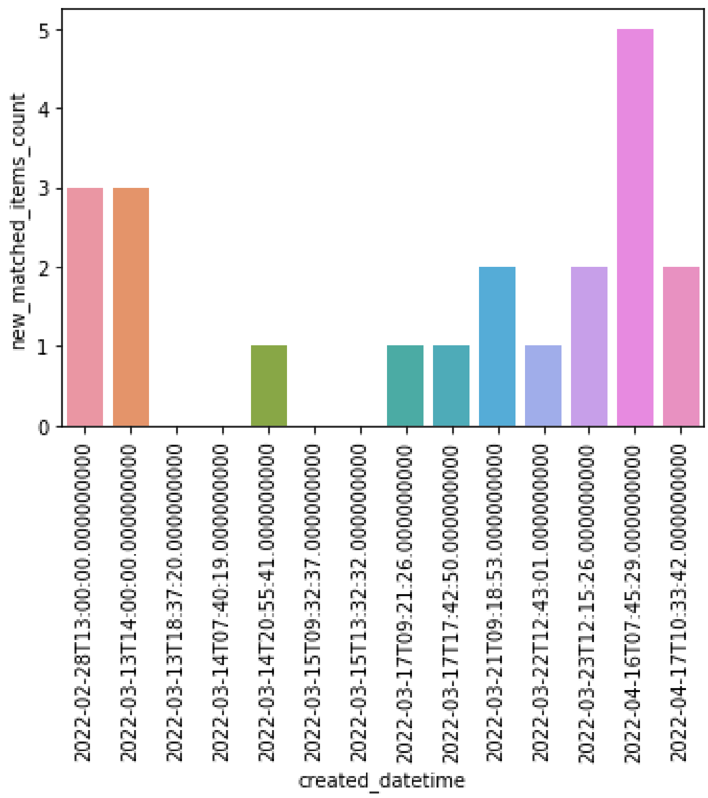


Figure 31: Trends based Recommendations Newly Matched Items *(self-composed)*

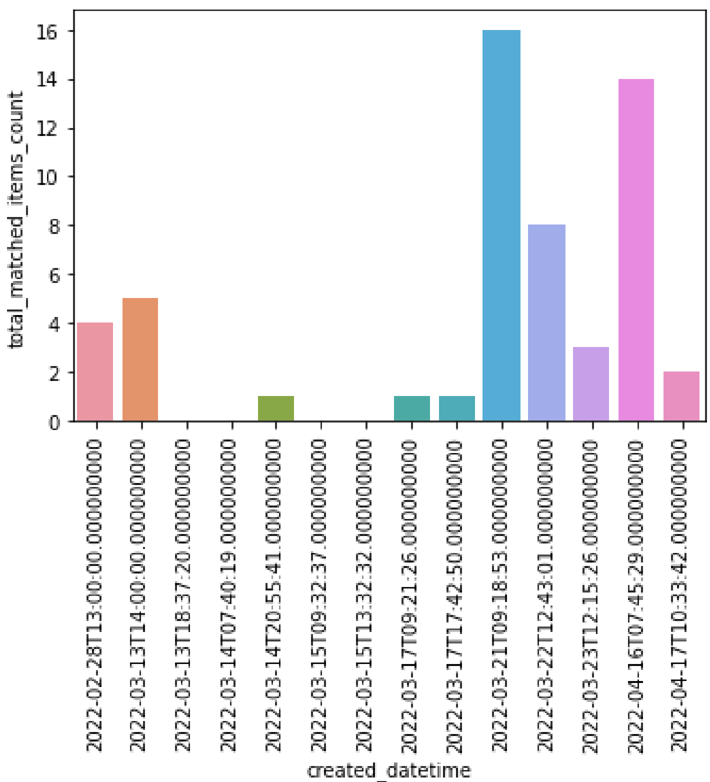


Figure 32: Trends based Recommendations Total Matched Items *(self-composed)*

Appendix E3 - Functional Testing

Table 3: 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	Passed
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	Passed
5	FR5	Admins adjusts the admin bias	The admin-bias gets adjusted in the system	The admin-bias was adjusted in the system	Passed
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
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
9	FR9	No Action - previously pressed view recommendations	The expected Recommendations by the user are generated by the system	The expected Recommendations by the user are generated by the system	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

Appendix E4 - Non-functional Testing

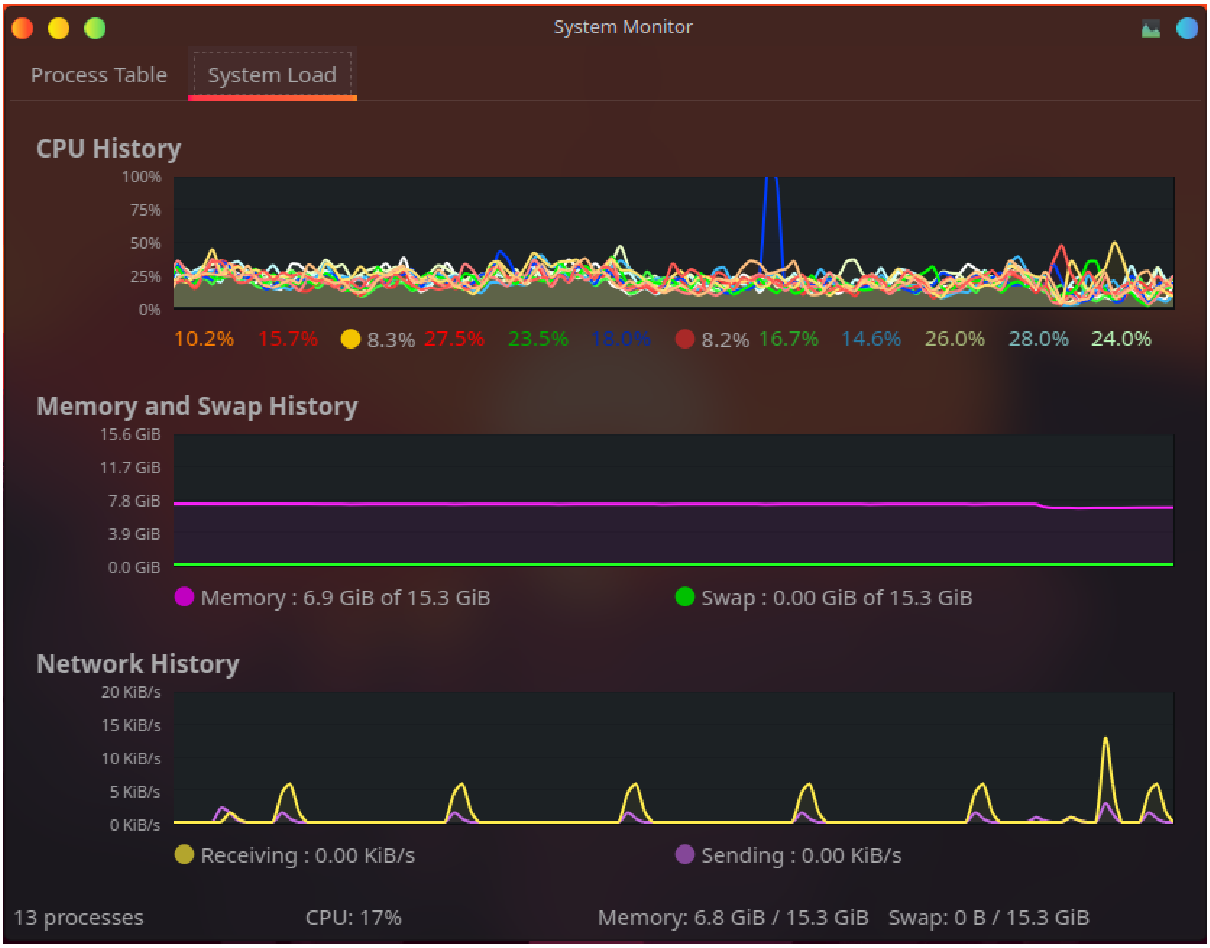


Figure 33: Non-functional Testing - System Resource Manager Graph *(self-composed)*

APPENDIX F - EVALUATIONS

Appendix F1 - Evaluations received by Evaluators

Please refer to the categorization in the table ***Categorization of selected evaluators*** to understand the Category Ids used to categorize the evaluators.

Table 4: Evaluations received by Evaluators

Evaluator	Feedback
Prof. Narada Warakagoda Principal Scientist at FFI/Associate Professor at University of Oslo *Category Id(s): 1*	*"This research is well motivated and attempts to address an interesting research gap which is due to the unique nature of NFTs and the way they are traded. The researcher has investigated alternatives to the widely used technique, Collaborative Filtering which may be difficult to adapt to the case of NFT recommendations. The research is thus focused on three main approaches; trait content-, trait rarity- and trend-based techniques. In this way, the research has been able to satisfactorily address the research gap. The researcher has approached the solution in a straight-forward manner with well founded choices. His solution is based on the state-of-the-art domain knowledge as well as adequate software. The researcher has provided solutions of good quality. His contributions are impressive, considering the fact that the work has led to (at least) two publications and is carried out as an undergraduate project. The work also touches upon many areas, including different recommendation systems, sentiment analysis, deep learning (including LSTM and Transformers), NLP and different evaluation metrics etc. But most importantly, the researcher has implemented and evaluated recommendation systems suitable for NFTs, and hence contributed with novel results. "*

	The models have been tested/evaluated and the results give some insight into their properties. However, this could have been done in a more systematic manner, tying evaluation more purposefully to a concrete research question. For example, a research question such as "which of the three approaches and their combinations is the best one?" could have been formulated and evaluations could have been designed to answer this question. Further, the models could have been evaluated with respect to a quantitative metric such as recommendation accuracy in addition to the qualitative metric Aggregate Diversity. Documentation of the work could have been improved. The researcher could have described the aim of the work, method and data sets (both training and testing) with more clarity and details using a better organized text. It is also important to provide a clear set of conclusions anchored to the plots, calculated metrics and other numerical results.
Mr. Sharmilan Somasundaram CEO of Niftron - Blockchain as a Service, Certified Blockchain Solution Architect (CBSA), MSc Big Data Analytics *Category Id(s): 1, 2*	*"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 MS Computer Science (USA), Senior Lecturer - IIT *Category Id(s): 1*	*"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."*

Dr. Kaneeka Vidanage PhD, Artificial Intelligence, Senior Lecturer - University of Moratuwa *Category Id(s): 1*	*"Good topic. I noticed, the mixing of NFT idea and use of recommendation systems are not clearly tallied in the research problem / gap. The conducted research has been able to 'Satisfactory'-ly address the identified research gap. The touch on recommendation systems perspective is lacking , specialling integration with NFT. Why NFT and Recommendation systems combination is not very prominent in the video clip shared. Why other approches not eligible to link with NFT ? Contributions are good, however, strong justificications are required, by compling the findings with evaluation results. Didn't notice a strong evaluation segment, rather than, few stasitical outputs. Satisfactory - Testing & Evaluation of the developed module. I feel more testing is needed. Not only technical and statistical tests. But stakeholder view assesment and relating it with the research gap justification is required. As suggested improvements: Presentation of your idea is complex. Audio is also very low. It should have a gradual flow for the audience to slowly and steadily understand it. Convincing a complex technical ideal in an understandable manner is also a mandatory skill need to be developed by a researcher / scholar. "*
Akshika Wijesundara, PhD Senior Technical Lead - Data Scientist (Growth) at WSO2, PhD in Computer Science - The Open University (UK), Founder and Chairperson of SEF *Category Id(s): 1*	*"The research is interesting. Especially, considering the fact that this is an undergraduate project, the researcher has put in a lot of effort into conducting the research. The contributions are clear and the gap has been addressed. I would recommend synthesizing data to evaluate the models. Especially if a user-preference is required to be evaluated for personalized recommendations. Synthetic data could be used to check if the matched trends of the recommended items comply with the items containing the required trending terms."*

Anonymous Blockchain Masters Researcher *Category Id(s): 1, 2*	*"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."*
Mr. Aadhil Rushdy Senior Data Engineer/Team Lead - Sysco Labs, MSc Computer Science - specialized in Data Science Engineering & Analytics (University of Moratuwa) *Category Id(s): 1*	*"This research becomes a novel solution due to the application of multiple recommendation algorithms to recommend a digital object like NFT that has a high willing to pay and fast moving nature. Considering the value of the object with the time and not being sticking to collaborative filtering opens up a good research area. Applying a ensemble of a recommendation approach to mitigate the timely recommendation of objects can be considered as the appropriate approach and the novelty is reflected through this approach. Identifying the research problem followed by identifying technical gaps in recommendation systems is a good starting point to go for the solution. Application of social trends to the recommendation model and the novel trends score calculation algorithm can be considered as a a great contribution by the researcher. Testing and evaluation done on the multiple models and a fair comparison done to show the importance of using both the models. Could have emphasized on the accuracy of the model since this solution going to impact the end users business or the capital."*

Ms. Areefa Thassim Software Engineer - MillenniumIT ESP, MSc Big Data Analytics *Category Id(s): 1, 3*	*"The research is interesting with NFTs being a relatively new area. The research gap identified is useful for identifying new NFTs. With the NFT marketplace growing rapidly, recommendation systems are going to be needed. The researcher has considered many angles to address the research gap by experimenting with multiple models and approaches. The researcher has excelled in identifying the necessary areas to provide a recommendation to a potential buyer, considering the data that can be accessed for each NFT. The researcher has approached the problem by relating the NFTs as individual products and using their specific traits for recommendation. Since the blockchain is used for privacy and protection of users, this is a good approach. The researcher has provided a good model that has been researched, implemented and evaluated well. A novel approach was provided for recommendations in the NFT domain. The researcher has evaluated the solution well in technical aspects. it would be good if the researcher could approach people and survey them based on the recommendations provided by the recommendation engine. The researcher has evaluated the solution well in technical aspects. it would be good if the researcher could approach people and survey them based on the recommendations provided by the recommendation engine."*

Anonymous Data Engineer *Category Id(s): 1*	*"The identified gap seems wide and looks good for an undergraduate research. Recommendation systems and algorithms have been there for a long time now but getting that approach and applying it in a newly rising field such as NFTs is great. The identified wide research gap has been addressed with this research. It is a good approach, implementing multiple types of recommendation models for recommendations based on different aspects. The solution that has been implemented seems to be working fine and the contributions are also great for an undergraduate research. I would suggest to evaluate the implemented models by their performance against some other models in the domain. That would make it easier for anyone to identify where this model stands compared to other models in the domain."*
Mr. Jajeththanan Sabapathipillai Chief Product Officer - Niftron, BE (Hons) Computer Software Engineering *Category Id(s): 2, 3*	*"Research gap is much needed to analyze the trend as its a NFT hype period and people would love a system to filter the best NFTs to trade. Research has covered the overall area of the research gap. Researcher has approached the research in more generic state where it could have been categorized and analyzed based on several parameters. Valuable NFTs could have evaluated by different parameters such as country, age, domain(art, game NFT), etc. The solution covers recommendations made on top trending NFTs on social media which are most famous ones, but does trending only enough to correlate with the value is the real question here. Should consider NFT utilities, buy and sell trend, no of bids, and visibility as well to identify the top NFTs"*

Mr. Sasila Hapuarachchi Incoming Engineer - Amazon CA, BE Electrical Engineering (Co-op) (Canada) *Category Id(s): 3*	*"The research provides useful background to the topic and the concept map further displays the various sectors of the research. The identified research gap is made clear as NFTs are a relatively new technology and the challenges of NFT recommendation systems can be understood due to their uniqueness. The conducted research discovers areas of exploration to address the research gap by identifying previously researched attempts to address the problem that include content-based, collaborative-based and personalized recommendation systems. The research highlights the potential of matching content with social trends along for better recommendations. The researcher has approached the solution in a comprehensive manner and expands on relevant previous work and ideas. The incorporation of social media throughout the project is also done well. The solutions provided in this project take into account a variety of factors that can be useful for the recommendation of NFTs. The use of a trend score in the design is particularly interesting and can be of great benefit to the recommendations as social media has been a major driving force behind the sudden rise in popularity of NFTs. Additionally, incorporating a rarity based model in the design fits well with the overall theme of NFTs and their inherent uniqueness which can drive user interest based on scarcity. Moreover, the content similarity matching model serves as a good integration of existing methodology. The developed models were tested and evaluated appropriately. However, it would be useful to see a more comparable evaluation metric for the Trends Based Model to allow for easy comparison with the other models. Although specific examples of NFTs are mentioned in the project, it would be useful to see details about the reference NFTs and further clarification on the obtained recommendation results for each model in order to clearly observe their connection to the reference NFT."*

Mr. Narada Wickramage Assistant Director - MIS & Statistics, Public Utilities Commission of Sri Lanka, MSc, MBA *Category Id(s): 3*	*"NFT market is expected to reach around $150 billion by 2026 and it is fast growing. NFT trading platforms are available but recommendation systems are not common. Therefore this is a good research area. The research gap has been addressed to a considerable extent. A scientific approach has been taken to approach the solution. Based on tweets but it is sufficient work for an undergraduate project. Evaluation is comprehensive. A suggested improvement is, making this social by developing a social network of participants so that they can collaboratively take part in trading."*
Mr. Achala Aponso Senior Lecturer - IIT, MSc Artificial Intelligence *Category Id(s): 1*	*"It's a good project overall. What you have discovered is very interesting. Since only 1 or 2 students in the batch will come up with a new algorithm, the entire focus of the research will be on this. The selection of parameters have to be confidently explained. Change constants/parameters and check the output, to show why your selection is correct."*

Lasal Jayawardena Undergraduate - BSc (Hons) Artificial Intelligence and Data Science *Category Id(s): 1, 3*	*"It is a interesting research gap that's been addressed here. It is surely the proper direction. The work shown has addressed the research gap. I would say this is a good starting point and will make a good MVP. The solutions & contributions made have a strong foundation and is well presented. With systems like these you'll need a continuous training system to keep the model up-to-date. I would prefer the testing to be more extensive and have a maybe keep track of ROIs of the recommended NFT to determine performance. In a trader's point of view I would prefer trading bots for making quicker trades. In terms of the AI, I feel like the evaluation must be more convincing. And yes its best if you could benchmark the inference speeds of the system just to get a rough idea. I think your solutions could be better, if you could direct you focus on artists in particular. Tracking their activity or something along those lines. Reason being I would opt for long term trades where the main factor is the artist rather than the hype. And need to differentiate the model from pump and dumps. Most common pitfall I have seen."*

Gibran Kasif Trainee Software Engineer - Vestoria, Undergraduate - BSc (Hons) Computer Science *Category Id(s): 3*	*"This is a new venture especially in the NFT space as this approach has not yet surfaced or ever implemented among major NFT marketplaces such as Opensea, Axie Marketplace and Binance etc. The gap identified is relevant to the concept of a NFT, since an NFT asset is usually sold in limited supply or as a single unit. This is one of the main challenges that come with purchasing an NFT, to which the following research is expected to resolve. Yes, the conducted research has been able to address the identified research gap. This approach taken isn't yet noticed on NFT marketplaces, but by introducing such a system would become a versatile tool to the user/consumer interested in purchasing an NFT. Using the trend based model in filtering out NFTs, could present a list of viable assets that are close enough to its existing predecessors traits that had been sold out from the collection, which would have the potential to become more valuable. With that in mind, social influence also does have a positive correlation on the success of an NFT collection/brand. For e.g. CryptoPunks. With the combined use of the trait content and rarity models, both of which are normally included on a NFT, could allow the end user to tailor the results based on their preferences over such features. Each model's outcome and results were evidently shown on the summarized research. Since the research currently focuses on the social trends revolving around NFTs. It might be interesting to see in the future whether the system could also prevent fraudulent NFTs from appearing in the results. As most common NFT scams artificially or hack in order to boost up their social presence of their NFTs, which would affect the end user if encountered."*

Nazhim Kalam Software Engineering Trainee - 99x, Undergraduate - BSc (Hons) Computer Science *Category Id(s): 1*	*"Since NFTs started getting attention recently, the identified research gap seems new, which is good. Yes, the conducted research has been able to address the identified research gap. I feel like you have done an in depth research with a lot of statistics which I saw from the video attached. The solution seems pretty good and how the scores for negative, neutral and positive is calculated as a part of the solution is very understandable."*
Ammar Raneez Trainee Software Engineer - 99x, Undergraduate - BSc (Hons) Computer Science *Category Id(s): 1, 3*	*"Clear gap and issue with the traditional collaborative filtering method has been addressed. Yes, the conducted research has been able to address the identified research gap. A straightforward and methodical approach solving each problem at a time, rather than having a direct jump to model construction, especially since this is a novel research topic. The issue of using only a single model to perform recommendations was clear and the approach taken was explained well. The approach of using the pre-trained twitter model to conduct a sentiment analysis evaluation was well done and the reasoning for doing so was explained well. This technique of evaluation is pretty much best in class."*

Appendix F2 - Evaluation of Functional Requirements

Table 5: 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	Implemented
FR3	The system could be able to fetch relevant data of the NFT using an entered contract address & 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	Implemented
FR5	Admins should be able to choose the bias of the Recommendations System.	S	UC3	Implemented
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	Not-Considered
FR8	The system could show the reasons for recommending each item to users.	C	UC4	Implemented
FR9	The system should generate recommendations based on what the user expects to view	S	UC5	Implemented
FR10	Opinion mining trends data must be used to generate NFT recommendations.	M	UC7	Implemented
FR11	Admins could be able to feed data-points such as interested public figures, websites to use as opinion mining data for recommendations.	S	UC8	Not-Considered

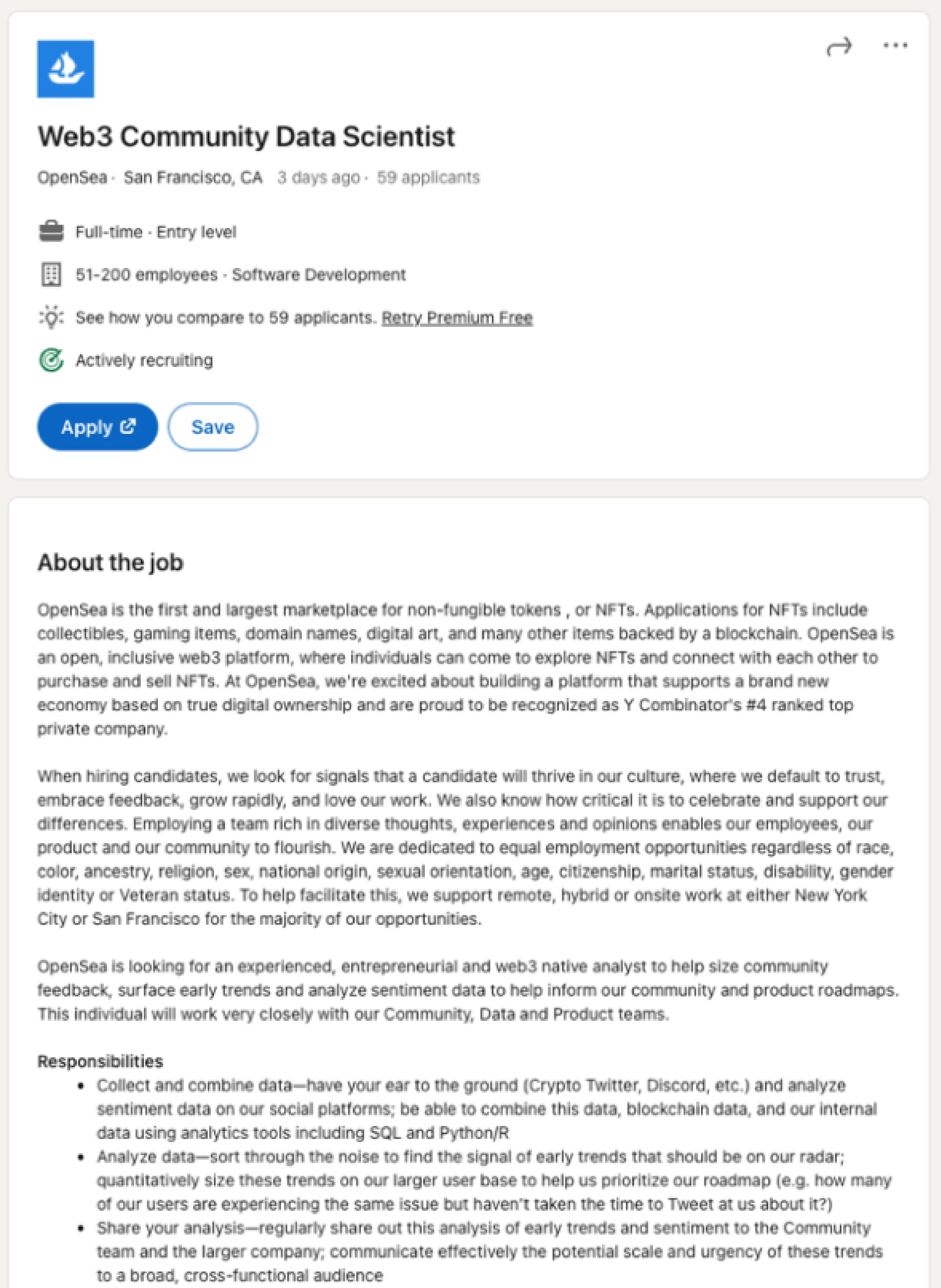
FR12	User-input could be aggregated and used as a reinforcement learning bias for the Recommendations Model.	C	NA	Not-Considered
FR13	The system will not act as a decentralized system.	W	NA	Not-Considered
Functional Requirement Completion Percentage = $\frac{9}{14} * 100 = 64\%$				

Appendix F3 - Evaluation of Non-Functional Requirements

Table 6: 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	Implemented - minimal
4	Usability	Important	Implemented
5	Scalability	Desirable	Implemented
Non-Functional Requirement Completion Percentage = $\frac{5}{5} * 100 = \%$			

Appendix F4 - Self Evaluation



Web3 Community Data Scientist

OpenSea · San Francisco, CA 3 days ago · 59 applicants

Full-time · Entry level

51-200 employees · Software Development

See how you compare to 59 applicants. Retry Premium Free

Actively recruiting

Apply Save

About the job

OpenSea is the first and largest marketplace for non-fungible tokens , or NFTs. Applications for NFTs include collectibles, gaming items, domain names, digital art, and many other items backed by a blockchain. OpenSea is an open, inclusive web3 platform, where individuals can come to explore NFTs and connect with each other to purchase and sell NFTs. At OpenSea, we're excited about building a platform that supports a brand new economy based on true digital ownership and are proud to be recognized as Y Combinator's #4 ranked top private company.

When hiring candidates, we look for signals that a candidate will thrive in our culture, where we default to trust, embrace feedback, grow rapidly, and love our work. We also know how critical it is to celebrate and support our differences. Employing a team rich in diverse thoughts, experiences and opinions enables our employees, our product and our community to flourish. We are dedicated to equal employment opportunities regardless of race, color, ancestry, religion, sex, national origin, sexual orientation, age, citizenship, marital status, disability, gender identity or Veteran status. To help facilitate this, we support remote, hybrid or onsite work at either New York City or San Francisco for the majority of our opportunities.

OpenSea is looking for an experienced, entrepreneurial and web3 native analyst to help size community feedback, surface early trends and analyze sentiment data to help inform our community and product roadmaps. This individual will work very closely with our Community, Data and Product teams.

Responsibilities

- Collect and combine data—have your ear to the ground (Crypto Twitter, Discord, etc.) and analyze sentiment data on our social platforms; be able to combine this data, blockchain data, and our internal data using analytics tools including SQL and Python/R
- Analyze data—sort through the noise to find the signal of early trends that should be on our radar; quantitatively size these trends on our larger user base to help us prioritize our roadmap (e.g. how many of our users are experiencing the same issue but haven't taken the time to Tweet at us about it?)
- Share your analysis—regularly share out this analysis of early trends and sentiment to the Community team and the larger company; communicate effectively the potential scale and urgency of these trends to a broad, cross-functional audience

Figure 34: Screenshot of OpenSea DataScience job posting on LinkedIn - 26/04/2022

APPENDIX G - CONCLUSION

Table 7: Completion Status of Research Objectives

Objective	Description	Status
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.	Completed
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.	Completed

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.	Completed
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.	Completed
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 P@K score, compared against baseline models.	Completed

Documenting the progress of the research	Documenting and notifying the continuous progress of the research project and any faced obstacles.	Completed
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.	Completed

Table 8: Achievement of Learning Outcomes

Learnings	**LOs**
The identified problem was tackled after selecting & applying appropriate methods with justifications.	LO1
Requirements of the project were gathered from literature, surveys & interviews. Then, the collected requirements were analyzed and clearly identified.	LO2
Over 50 literature were read, surveyed, then critically reviewed.	LO4
There were many tasks that the author had defined and was expected to carry out throughout the research. These tasks were carried out after analyzing all the possible available options.	LO5, LO3
The supervisor was regularly met and the deliverables were produced as agreed based on a developed project plan that scheduled the required activities to be carried out.	LO6, LO3
SLEP Issues related to the research were taken note of and mitigated by identifying possible issues and planning ahead of handling those issues.	LO7

The entire research was documented at the best possible standard & structure with critical evaluations, following the guidance given by the module leader & supervisor. The documents included the project proposal, Project Specification & Prototype Document (PSPD), this thesis, 2 research papers & 1 review paper. All the papers were published as pre-prints on ArXiv and the research papers were submitted to research conferences.	LO8

APPENDIX - RESEARCH PAPER 1

An Analysis of the Features Considerable for NFT Recommendations

Dinuka Piyadigama
Computer Science and Engineering
University of Westminster
London, UK
drpiyadigama@gmail.com

Guhanathan Poravi
Department of Computing
Informatics Institute of Technology
Colombo 06, Sri Lanka
guhanathan.p@iit.ac.lk

***Abstract*—This research explores the methods that Non-fungible Token (NFT)s can be recommended to people who interact with NFT-marketplaces to explore NFTs of preference and similarity to what they have been searching for. While exploring past methods that can be adopted for recommendations, the use of NFT traits for recommendations has been explored. The outcome of the research highlights the necessity of using multiple Recommender Systems to present the user with the best possible NFTs when interacting with decentralized systems.**

***Index Terms*—Non-fungible Tokens, Recommender Systems, Information Retrieval, Data mining, Data science**

I. Introduction

In recent months, the NFT market has been growing exponentially as it appears to be the most widely accepted business application of Blockchain technology [1], since the introduction of crypto. With more and more people expected to enter connected digital environments such as the metaverse [2], it is clear that NFTs will play a huge role in tomorrow's internet [3] due to its ability to make digital items have scarcity, uniqueness, and proof of ownership, similar to physical items [4]. Human interactions of the next decade on the internet may entirely rely on NFTs.

A. What are NFTs?

NFTs are provably scarce unique digital assets that can be used to represent ownership [5]. They can be one-of-a-kind rare artworks, collectible trading cards, and other assets with the potential to increase in value due to scarcity [6], [7]. 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 [8].

NFTs were introduced by Ethereum [9] as an improvement proposal [10], [11] in the Ethereum Request for Comments (ERC)-721 standard [5]. 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.

B. Smart Contracts & ERC standards

Smart Contracts are code that is running on the Blockchain. 3 of the notable ERC standards can be identified in table I.

TABLE I
COMPARISON OF ERC STANDARDS

ERC-721	ERC-777	ERC-1155	ERC-20
Non-fungible tokens	Non-fungible tokens [12]	Semi-fungible tokens [13]	Fungible tokens
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
CryptoKitties [14]		Concert tickets, gift vouchers, coupons	Crypto currencies - Bitcoin, ETH

Each of the 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 [15] on the Ethereum network. One Bitcoin can be swapped withn 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 [16].

For each NFT, the contact address & unit256 tokenId are globally unique on any blockchain. This allows Decentralized Applications (DApps) [17], [18] to take the tokenId and present the image/ asset that is identified by the particular NFT.

C. 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 [19], following a $100 million raise in a Series B round, ended the company in a valuation of $1.5 billion [20], [21]. 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 [22].

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 [5].

II. Motivation to explore how to Recommend NFTs

Recommendation System (RecSys)s play a significant role in the resolution of the problem of information overload [23]. 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.

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 relevant items, it enables businesses in growing their revenue. 35% of Amazon's revenue [24] & 60% of watch time on YouTube [25] comes from recommendations. 75% of Netlfix viewer activity [26] 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

Since generating relevant recommendations 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.

In this research, the author attempts to identify features that could be considered for recommending NFTs and the importance of using multiple feature sets and algorithms to recommend relevant items.

III. Value-driving factors of NFTs

A. 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, on any date in the future, no matter how many wallets it gets passed through [22]. 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 [5].

B. Pricing of NFTs

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

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"* [27]. As a future study, the author has suggested identifying if there's a fundamental model that drives the price determination in NFTs.

> *"The value of an NFT is entirely determined by what someone else is willing to pay for it."*
>
> [6]

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."*
>
> [5]

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.

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 [1]. These models can be used separately on each NFT asset to anticipate the pricing related to time, sales & bids.

IV. Existing Work

A. NFT Collections Recommendation System

Conderation of the use of a basic Machine Learning (ML) technique called **Multiple Regression** with data gathered from OpenSea in a blog article on *OpenSea* [28].

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 of 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 is 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 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, to get a recommendation from the system. Although getting a users' public key can by no means cause any threat of losing the NFTs, it can lead to a lack of privacy, which is a tradition that the people into crypto-related assets have a tendency to be concerned about.

B. *Data Mining NFT Data from OpenSea*

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 Collection [29] NFT sales [30].

This work analyzes the outputs of the following data in the dataset.

1) Top 10 Meebits Creators, Buyers & Sellers
2) The total number of Meebit Creators and Owners
3) Stats about Bundle/Single Sales
4) Types of Payment Currencies
5) Total Number of Sales per Day
6) Total Sales per Day in ETH & USD
7) Average, Max & Floor Meebit Price per Day in ETH

While this work helps a lot with data mining, cleaning, preprocessing the data, and identifying the best possible users to target from a business perspective, it's doesn't explore how recommendations can be generated using the available data for specific items within the dataset.

C. *What may be the reasons for the lack of research related to recommending NFTs?*

> *"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."*
>
> [28]

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

V. Proposed Approaches for Recommendations

When conducting a requirement survey prior to building the prototype, the author understood that there was a clear necessity for NFT creators, buyers & sellers to find items based on traits of an NFT. Traits are the properties that describe whatever that is contained in the image/ NFT asset.

Recommending items using NFT traits was attempted by the author in 2 different ways.

The ***Bored Ape Yacht Club***'s 10,000 NFTs were used to generate recommendations in this research. The Reference Id represented in the graphs is in the format of *NFT Contract Address - Token Id*

A. *Trait Similarity Content-based Recommendations Approach*

The trait type and value were combined as lowercased strings to create a single string that would be unique even if NFTs from multiple collections were used to generate the cosine similarity matrix. A Count-Vectorizer was used to get a vectorized similarity score between all items considered for recommendations. The reason for choosing a Count-Vectorizer over a Tf-Idf Vectorizer was because all traits were considered equally important, to calculate an aggregate similarity score of all traits per item.

The top 10 items that had the cosine similarity score of the reference item's traits were taken as the recommendations here.

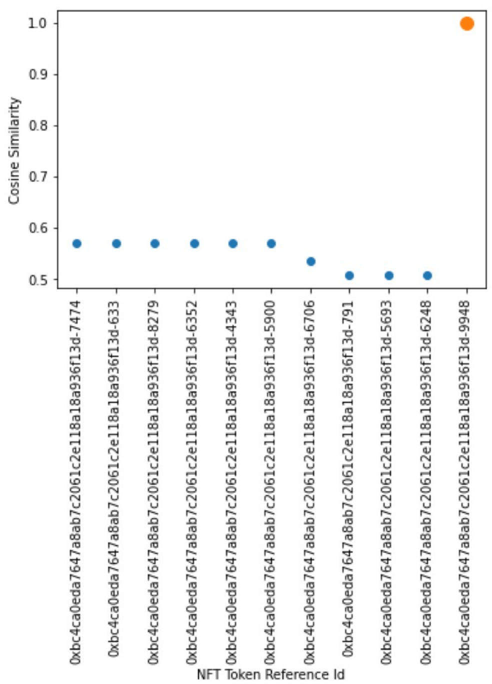


Fig. 1. Trait Similarity Content based recommendations

B. *Trait Rarity based Recommendations Approach*

The rarer the traits are, the more valuable it would make an NFT. A rarity-score calculation method was introduced by rarity tools to calculate the total rarity-score of an item [31].

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

Fig. 2. Equation to calculate the total trait rarity score of an NFT [32]

$T_{r,t}$ - 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.

The top 10 items that had the total rarity as close as possible to the reference item's rarity were taken as the recommendations here.

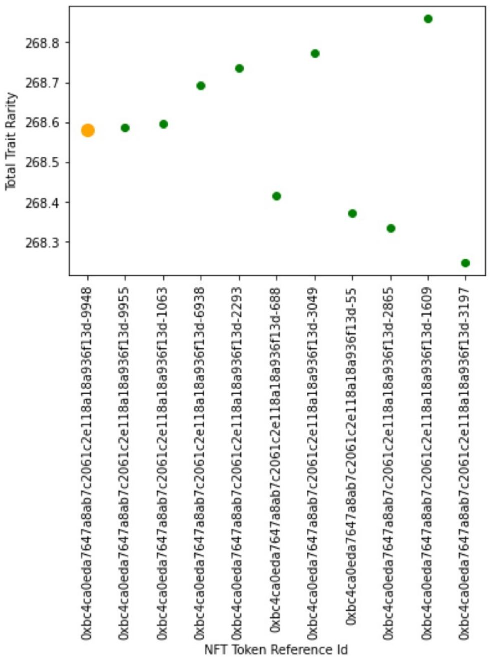


Fig. 3. Total Rarity based recommendations

VI. Evaluation

The perfect measurement of evaluating a Recommendation System hasn't been the most straightforward. Since the Recommendation Systems introduced in this research attempt to recommend items in a very specific domain, the author decided to place the outputs produced by the two Recommendation Models in opposite graphs of measurements to visualize & evaluate the produced outputs.

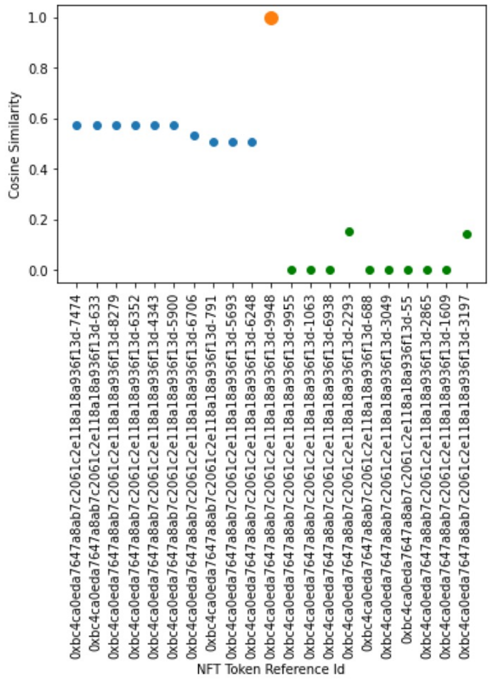


Fig. 4. Cosine similarities of recommendations generated by both models

In the generated graphs represented in figures 4 & 5, the item used to generate recommendations for is represented in orange, the ones in blue were generated by the Trait similarity content based model, and the ones represented in green were generated by the Trait rarity model. When taking a look at Figure 4, it's clear that none of the items recommended by the trait rarity model were even close to being content-wise similar.

When taking a look at Figure 5, the items recommended by the cosine similarity model were scattered around inconsistently.

As it has already been established that both these methods are necessary to be of value to a user when searching for items, it is clear that both the models are required to generate recommendations for a particular NFT using its traits.

VII. Conclusion

In this research, the author set about exploring work related to utilizing available features of NFTs to recommend NFTs

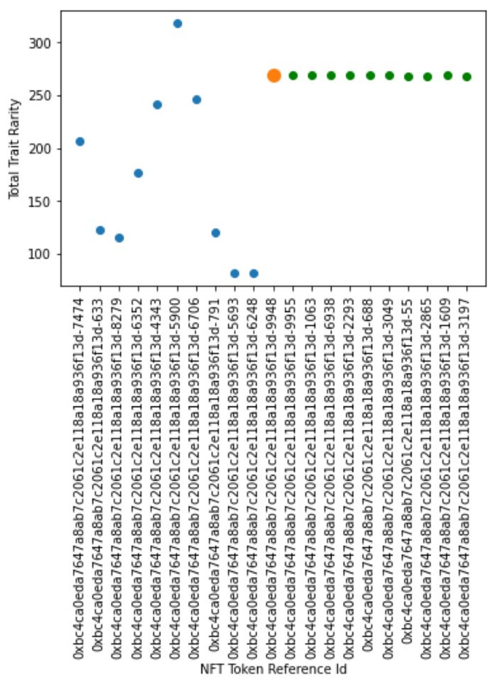


Fig. 5. Total rarities of recommendations generated by both models

to users. After heading into the possibilities of recommending NFTs using their traits, the outputs produced by two suggested Recommendation Models were depicted in graphs. Finally, the importance of the two models were explained due to the outputs produced being contrastingly diverse. The foundation laid by the findings of this research could be built upon in future work to make the explorability and relevance of recommended items. This could help create better links between NFTs and users, resulting in better connections & interactions among users as well as digital assets on the internet in the next decade.

VIII. Future Enhancements

If required to limit the recommendations produced, in order to recommend a fewer number of items with the most value to a user, it may be valuable to take the user's preference of each output into consideration. This could be done in a future study involving responses from human subjects.

Although the author initially planned to attempt recommending items using an Long short-term memory (LSTM) Deep learning (DL) model since similar attempts have been researched with cryptocurrencies [33], it was understood that the very nature of uniqueness brought forwards by NFTs would not produce consistent results across collections or even items within the same collection. Even to attempt creating a bid price prediction model was challenging due to the lack of data and strict rate limits in the OpenSea Application Programming Interface (API). Closer to the completion of this research, the author came across an open dataset [34] that may be usable for this purpose.

During the evaluation phase of the project, one of the feedback received regarding this was to attempt recommending NFTs using a dataset of price fluctuations in expensive, valuable physical artworks.

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, in relation to past price fluctuations. This could be an interesting area to dive into since human-desire can be a major factor of consideration of the acceptance of a particular item.

Since the acceptance of NFTs seem to have a very common connection with social media such as *Twitter*, *Reddit* & *Discord communities*, it may be possible to recommend items using such data.

Acknowledgment

The author of this paper acknowledges the guidance & evaluation insights received for the project from Mr. Sharmilan Somasundaram & Mr. Ragu Sivaraman.

References

[1] M. Dowling. (Apr. 29, 2021). "Is non-fungible token pricing driven by cryptocurrencies? — elsevier enhanced reader," [Online]. Available: https : / / www . sciencedirect . com / science / article / pii / S1544612321001781 ? via % 3Dihub (visited on 06/23/2021).

[2] Casey Newton. (Jul. 22, 2021). "Mark zuckerberg is betting facebook's future on the metaverse - the verge," The Verge, [Online]. Available: https://www.theverge.com / 22588022 / mark - zuckerberg - facebook - ceo - metaverse-interview (visited on 11/20/2021).

[3] Peter Allen Clark. (Nov. 15, 2021). "What is the metaverse? here's why it matters — time," Time, [Online]. Available: https://time.com/6116826/what-is-the-metaverse/ (visited on 11/20/2021).

[4] (). "Non-fungible tokens (NFT)," ethereum.org, [Online]. Available: https : / / ethereum . org (visited on 11/20/2021).

[5] (). "ERC-721 non-fungible token standard," ethereum.org, [Online]. Available: https://ethereum.org (visited on 07/19/2021).

[6] R. Conti. (Apr. 29, 2021). "What you need to know about non-fungible tokens (NFTs)," Forbes Advisor. Section: Investing, [Online]. Available: https://www.forbes.com/advisor/investing/nft-non-fungible-token/ (visited on 07/19/2021).

[7] J. Fairfield, "Tokenized: The law of non-fungible tokens and unique digital property," Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 3821102, Apr. 6, 2021. [Online]. Available: https://papers.ssrn.com/abstract=3821102 (visited on 04/18/2021).

[8] (Jul. 5, 2021). "Off the chain: NFT market surges to $2.5b so far this year," Aljazeera, [Online]. Available: 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).

[9] D. G. Wood, "ETHEREUM: A SECURE DECENTRALISED GENERALISED TRANSACTION LEDGER," p. 39, 2014. (visited on 07/15/2021).

[10] (). "EIP-2309: ERC-721 consecutive transfer extension," Ethereum Improvement Proposals, [Online]. Available: https://eips.ethereum.org/EIPS/eip-2309 (visited on 07/20/2021).

[11] (). "ERC," Ethereum Improvement Proposals, [Online]. Available: https://eips.ethereum.org/erc (visited on 08/24/2021).

[12] J. Dafflon, Jordi Baylina, and Thomas Shababi. (Nov. 20, 2017). "EIP-777: Token standard," Ethereum Improvement Proposals, [Online]. Available: https://eips.ethereum.org/EIPS/eip-777 (visited on 11/19/2021).

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

[14] CryptoKitties. (). "CryptoKitties — collect and breed digital cats!" CryptoKitties, [Online]. Available: https://www.cryptokitties.co (visited on 11/20/2021).

[15] (). "ERC-20 token standard," ethereum.org, [Online]. Available: https://ethereum.org (visited on 08/31/2021).

[16] (). "Introduction to smart contracts," ethereum.org, [Online]. Available: https://ethereum.org (visited on 08/31/2021).

[17] J. Frankenfield. (). "Decentralized applications – dApps," Investopedia, [Online]. Available: https://www.investopedia.com/terms/d/decentralized-applications-dapps.asp (visited on 09/01/2021).

[18] (Sep. 1, 2021). "Decentralized applications (dapps)," ethereum.org, [Online]. Available: https://ethereum.org (visited on 09/01/2021).

[19] R. Hackett. (Mar. 18, 2021). "This crypto marketplace just raised $23 million to be the 'amazon of NFTs'," Fortune, [Online]. Available: https://fortune.com/2021/03/18/nft-art-crypto-marketplace-opensea-amazon/ (visited on 07/15/2021).

[20] dfinzer. (Jul. 20, 2021). "Announcing our $100m raise, led by a16z," OpenSea Blog. Section: Company Announcements, [Online]. Available: https://opensea.io/blog/announcements/announcing-our-100m-raise-led-by-a16z/ (visited on 07/23/2021).

[21] L. Matney. (Jul. 20, 2021). "NFT market OpenSea hits $1.5 billion valuation," TechCrunch, [Online]. Available: https://social.techcrunch.com/2021/07/20/nft-market-opensea-hits-1-5-billion-valuation/ (visited on 07/23/2021).

[22] S. Chevet, "Blockchain technology and non-fungible tokens: Reshaping value chains in creative industries," *SSRN Electronic Journal*, 2018, ISSN: 1556-5068. DOI: 10.2139/ssrn.3212662. [Online]. Available: https://www.ssrn.com/abstract=3212662 (visited on 04/18/2021).

[23] L. C. Cheng and M.-C. Lin, "A hybrid recommender system for the mining of consumer preferences from their reviews," *Journal of Information Science*, vol. 46, no. 5, pp. 664–682, Oct. 2020, ISSN: 0165-5515, 1741-6485. DOI: 10.1177/0165551519849510. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0165551519849510 (visited on 07/16/2021).

[24] M. Naumov, D. Mudigere, H.-J. M. Shi, J. Huang, N. Sundaraman, J. Park, X. Wang, U. Gupta, C.-J. Wu, A. G. Azzolini, D. Dzhulgakov, A. Mallevich, I. Cherniavskii, Y. Lu, R. Krishnamoorthi, A. Yu, V. Kondratenko, S. Pereira, X. Chen, W. Chen, V. Rao, B. Jia, L. Xiong, and M. Smelyanskiy, "Deep learning recommendation model for personalization and recommendation systems," *arXiv:1906.00091 [cs]*, May 31, 2019. arXiv: 1906.00091. [Online]. Available: http://arxiv.org/abs/1906.00091 (visited on 04/29/2021).

[25] (). "Recommendations: What and why? — recommendation systems," [Online]. Available: https://developers.google.com/machine-learning/recommendation/overview (visited on 08/24/2021).

[26] T. Vanderbilt, "The science behind the netflix algorithms that decide what you'll watch next," *Wired*, Section: tags, ISSN: 1059-1028. [Online]. Available: https://www.wired.com/2013/08/qq-netflix-algorithm/ (visited on 08/27/2021).

[27] M. Dowling, "Fertile LAND: Pricing non-fungible tokens," *Finance Research Letters*, p. 102 096, Apr. 29, 2021, ISSN: 1544-6123. DOI: 10.1016/j.frl.2021.102096. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S154461232100177X (visited on 07/17/2021).

[28] (Jan. 30, 2020). "What are you missing? using basic machine learning to predict and recommend NFTs with OpenSea data - OpenSea blog," [Online]. Available: https://opensea.io/blog/analysis/predict-and-recommend-nfts/ (visited on 08/27/2021).

[29] Larva Labs. (). "The meebits," [Online]. Available: https://meebits.larvalabs.com/ (visited on 11/20/2021).

[30] Adil Moujahid. (Jun. 28, 2021). "Data mining meebits NFTs using python and OpenSea API // adil moujahid // data analytics and more," [Online]. Available: http://

adilmoujahid.com/posts/2021/06/data-mining-meebits-nfts-python-opensea/ (visited on 10/09/2021).
[31] rarity.tools. (Apr. 28, 2021). "Introducing rarity.tools. rarity.tools is a website dedicated to. . . — by rarity.tools — medium," [Online]. Available: https://raritytools.medium.com/introducing-rarity-tools-9b0138e992b3 (visited on 04/28/2022).
[32] ——, (Oct. 31, 2021). "Ranking rarity: Understanding rarity calculation methods," Medium, [Online]. Available: https://raritytools.medium.com/ranking-rarity-understanding-rarity-calculation-methods-86ceaeb9b98c (visited on 04/28/2022).
[33] F. Ferdiansyah, S. H. Othman, R. Zahilah Raja Md Radzi, D. Stiawan, Y. Sazaki, and U. Ependi, "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)*, Oct. 2019, pp. 206–210. DOI: 10.1109/ICECOS47637.2019.8984499. (visited on 07/15/2021).
[34] ZOMGLINGS and Avery Smith. (Sep. 25, 2021). "Ethereum NFTs — kaggle," Kaggle, [Online]. Available: https://www.kaggle.com/datasets/simiotic/ethereum-nfts (visited on 04/28/2022).

APPENDIX - RESEARCH PAPER 2

Exploration of the possibility of infusing Social Media Trends into generating NFT Recommendations

DINUKA RAVIJAYA PIYADIGAMA, University of Westminster, UK
GUHANATHAN PORAVI, Informatics Institute of Technology, Sri Lanka

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 has been attempted to improve the recommendations that can be provided by baseline methods in this research when user-click data is lacking or is difficult to be collected due to privacy concerns.

Utilizing social trends to influence the recommendations generated for a set of unique items has been explored with the use of a suggested scoring mechanism. Embracing concepts from decentralized networks that are expected to change how users interact via the internet over the next couple of decades, the suggested Recommendations System attempts to make use of multiple sources of information, applying coherent information retrieval techniques to extract probable trending items.

The proposed Recommendations Architecture in the research presents a method to integrate social trends with recommendations to produce promising outputs.

CCS Concepts: • **Information systems → Recommender systems**; *Retrieval models and ranking*; *Data mining*; • **Human-centered computing → Social recommendation**; • **Applied computing →** Online shopping.

Additional Key Words and Phrases: Recommendation Systems, Opinion Mining, Non-fungible Tokens, Data Science, Algorithm Design

ACM Reference Format:
Dinuka Ravijaya Piyadigama and Guhanathan Poravi. 2022. Exploration of the possibility of infusing Social Media Trends into generating NFT Recommendations. In . ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Non-fungible Token (NFT)s allow people to trace the origin of digital items with the help of Blockchain technology. Since the introduction of crypto, NFTs have stood out to be the most widely accepted application of Blockchain technology.

One NFT is expected to be unique from another. As these items are unique from each other, as expressed by the name itself, they are *not fungible*. They cannot be replaced like crypto, which is fungible.

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 are trending among the community, timely and relevant to each user's identified interests.

Recommendations Systems have been identified to be one of the integral elements of driving sales in e-commerce sites. They have been driving engagement and consumption of content as well as items on almost every corner of the internet over the last decade. 30% of Amazon's revenue is said to come from the items recommended to users [14]. 60% of watch time on Youtube and 75% on Netflix were also reported to have come about as a result of recommendations [15, 17].

In the month of June of 2021, OpenSea which is poised to be the Amazon of NFTs facilitated sales of $150 million and was valuated at $1.5 Billion [9, 7, 6]. Therefore, it is clear that Recommendation Systems could help bolster sales on such NFT-marketplaces, bringing in more revenue to businesses, and creators while helping users explore trending & relevant items.

2 REVIEW OF RELATED WORK

2.1 Identified Challenges & Requirements to build a Social Trends aware Recommendations System for NFTs

When considering possible options to explore the possibility of recommending NFTs, it appeared that there hadn't been much past work that could relate to this specific purpose.

> *"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."*
>
> [18]

Being loosely related to the crypto market and community [8] it is understood that the early adopters & pioneers in this space are concerned about their privacy.

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 which was done related to a modelling framework that exposes this area of research [2] assumes that two main features, namely security and stability can be used to determine the user's desire to own a specific crypto asset. 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.

2.2 User Opinion & Sentiment Aware Recommendation Systems

> *"Catching opinions from social media could be a cheap, fast and effective way to collect feedbacks from users"*
>
> [20]

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 [10] expressing how sentiment analysis of user reviews can be used to point to the direction of personalized recommendations.

The utilization of opinion mining data extracted from trends to improve the recommendations that can be provided by baseline methods was expected to address the restraints of recommending items presented by the very nature of NFTs.

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 and it has been the standard baseline technique for Recommendations for over a decade [13, 16]. But, it can't be taken as the only recommendations model in this use-case 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 for a profitable cost as the value would've sky-rocketed due to high demand over a long period of time.

User data on social media has been integrated into Machine Learning (ML) Hybrid Recommendation Architectures in several ways to produce user-opinion & social context-aware Recommendations.

One of these methods was to apply opinion mining & sentiment analysis on users' reviews to create a preference profile and create collaborative filtering like recommendations [5]. While this method is effective in dealing with insufficient data, the issue related to the required use case is that users still have to place reviews on previous movies to create a preference profile, which would add to the privacy concern mentioned before. A similar Deep learning (DL) model was attempted to generate possible user ratings, again based on user comments [4], letting the same issue prevail.

A hybrid approach of combining content-based recommendation, user-to-user collaborative filtering and personalized recommendation techniques has been attempted, to finally show the sentiment analysis polarity of the recommended item based on a user's tweets [1]. While this model does a good job in addressing the limitation of single domain analysis such as data sparsity & cold start problem, it doesn't consider the sentiment for the particular recommendation. It is only calculated and made visible to the user. The user is required to make the decision of selecting if the item is worth. It also doesn't make use of trends that are happening on social media.

2.3 Research Motivation

In recent research done by **Amazon** [12] 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. Chronologically sorted movie-viewing data managed to outperform item-to-item collaborative filtering applied with the bestseller list. This was done by getting the model to recommend at least 2 recently released movies. The idea behind this was that a user is more likely to watch a recently released movie rather than a very popular and highly rated old movie.

The method followed to recommend items to the user in this case motivated the authors to pursue to attempt to infuse social media trends into Recommendations. The reason for this was that trends on social media will give an idea of the things, people that are very popular at that moment in time. A person would be more interested in getting a recommendation with whatever that is related to a popular topic rather than an old item that was very popular back for a while and highly rated.

This can be applied even to an e-commerce setting. But, especially in the case of NFTs, this opens up another way to get items that may not otherwise surface in front of users' eyes. It would also keep updating over time, as trending topics change. Therefore, new trends may open up new valuable, relevant items rather than old items that are already high in demand and owned by owners who may not wish to sell.

3 PROPOSED SYSTEM ARCHITECTURE DESIGN

In the attempt to find a method to recommend items based on people's aggregated opinions in the form of trending topics on social platforms, without having to track user clicks and online behaviour in a way that can expose individuals, the author decided to move in the direction of searching possible methods to integrate social media trends data that is sourced from external sources for recommendations. Due to the privacy concern mentioned earlier, the author's goal was to build the Recommendation System in a way that potential buyers' privacy isn't threatened by the collection of user click data.

3.1 System Process Flowchart

The flow of data in the system with the various processing steps these data flows through has been represented in Figure 1.

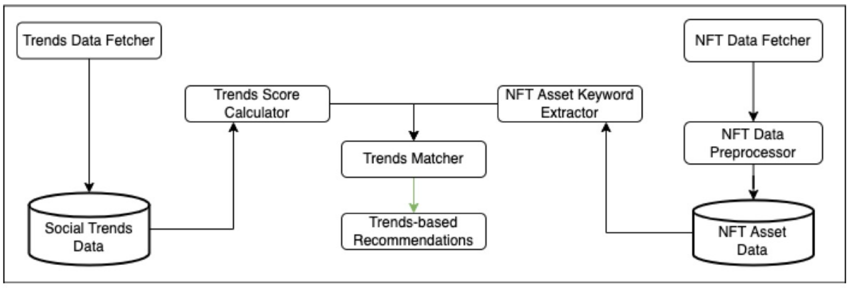


Fig. 1. System Process Flowchart *(self-composed)*

3.2 Algorithm Design

The following equation was designed to calculate the impact of a trend.

$$i_t = \frac{t_{vt,c}}{Med(T_{vt})} \tag{1}$$

The volume of a trend is divided by the median volume here to get a relative impact of a trend. When the trend score is kept as low as possible, when applied to the next total trend-score calculation equation, the score will drop down to a low value faster.

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 following equation was designed to calculate the total trends score of each item.

$$T_{t_s,i} = \frac{\sum_{i_s=1}^{N_{i_s}} \left[\sum_{k_w=1}^{k_w} s_c \left(\frac{t_{vt,c}}{Med(T_{vt})} \right) \frac{mu}{(\mu+n_m)} \right]}{N_{i_s}} \tag{2}$$

$T_{t_s,i}$ - 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.

Although the equation supports the calculation of a trend-score using multiple sources of trends, only Twitter trends were used for the testing & evaluation purposes of this research.

As much as the constant μ helps get rid of division by zero error for trends that happened on the same day of calculating the trend-score, it also helps multiply the score by 10 to significantly increase the trend-score of those trends.

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.

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. In this research, the author decided to test this with the top N items strategy, to generate featured items recommendations at a given date & time.

4 IMPLEMENTATION CHOICES

4.1 Extracting the Keywords of each item

NFT asset name, description, collection name & description were used to extract words that describe the asset. The *RAKE Vectorizer* of the **NTLK** Python Natural Language Processing (NLP) library was used for the purpose of extracting keywords from these descriptions.

4.2 Selection of a Sentiment Analysis Model

Sentiment scores of the top mentions of the trend on social media (top tweets) were generated using a pre-built sentiment analysis model. 3 Models were tested for this purpose. The 3 models that were tested were:

(1) SpacyTextBlob
(2) HappyTransformer
(3) Twitter-roBERTa-base for Sentiment Analysis

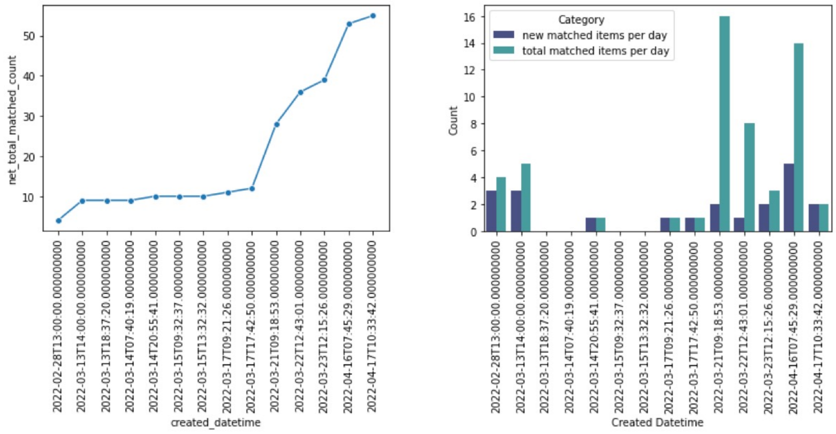
The 3rd model, which is a state of the art Transformer model [3, 19] that outputs negative, neutral & positive sentiment scores was chosen. This model outperformed the other two both in the accuracy of the sentiment and the speed. Another advantage of choosing this model was that it was trained on past Tweet data. Therefore, all complexities such as hashtags were handled by the model itself.

This model gave 3 parameters as output. Namely, negative, neutral and positive sentiment scores. The highest score from each of these was taken into consideration. In the case of negative sentiment, the trend score would become negative. It made sense to leave it as it was since items with negative sentiment may not be suitable to show to users. This could be modified based on the use case of the system. A neutral sentiment score was taken without any modification as well, while positive sentiment was multiplied by 2. This was done to have a clear increase in trends surrounding positive sentiment since it was expected by the author of the research that a user would be more likely to purchase an item with positive sentiment.

5 TESTING & EVALUATION

Out of 3872 randomly fetched NFT asset data & 677 randomly fetched trends from the OpenSea & Twitter Application Programming Interface (API)s respectively, across 14 different datetimes, 55 recommendations were produced using this method. All the trends & items mined from external APIs were randomly sourced & heavily pre-processed.

The practicality & value of the Recommendations Architecture suggested in this research can be better understood by a qualitative evaluation of the outputs produced by the generated graphs that are represented below.



(a) Total Trends based Recommendations made with time (b) Evaluation of Trends based Recommender

Fig. 2. Count of Generated Recommendations Recommendations generated *(self-composed)*

The graph in Figure 2b shows the count of the items that were matched with the trends of each datetime, highlighting the counts of newly matched items. It is important to note that the rankings of these items that were recommended are updated each time a new set of trends are entered into the system (using an automated process) based on the trend score that is calculated.

The count of matched items in Figure 2 may have been low over the first few days due to fetching worldwide trends, which included trends in languages such as Chinese & Korean that were not contained in the descriptions of the items. Towards the latter half of the experiment, trends from only the UK were fetched to overcome this constraint.

A sample output that was generated by the model from a pandas data-frame output of a Jupyter notebook is shown in Figure 3. The keywords have been used to find matches between an NFT which is referenced by the reference_id (a combination of the Contract Address of the Smart Contract [11] that minted the NFT & the Token Id of the NFT). This reference_id can be used to track the NFT on the Blockchain.

The heatmap generated by the output produced in Figure 4 shows how the trend-score for items decreases with time, from the date of matching with the trend for items that had a maximum trend-score of 30. The max score was limited to

```
top_trending_nfts_df = trends_based_recommendations()
top_trending_nfts_df.head()
```

reference_id	All_key_words_list	All_key_words_str	trend_score	matched_trends
0x495f947276749ce646f68ac8c248420045cb7b5e-30041571019036570279696425714306755064417936087066583674697121488131734896641	[jesus, simmons, cryptoguysnft, preparing, nft...	jesus simmons cryptoguysnft preparing nft game...	6.817898	[jesus]
0x495f947276749ce646f68ac8c248420045cb7b5e-94916783746142258917689322291875902080657954662840061252830161925292293095425	[triangles, 109, rihanna, mesmerizer, theme, t...	triangles 109 rihanna mesmerizer theme traits ...	4.530678	[rihanna]
0x495f947276749ce646f68ac8c248420045cb7b5e-110044771230368441551571992070040963530623495929530335175865263817154854125569	[map, art, 135, elon, musk, item, 135, officia...	map art 135 elon musk item 135 official map ar...	2.270546	[elon]
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263433004574310401	[midnight, art, 2, elon, musk, mesmerizer, the...	midnight art 2 elon musk mesmerizer theme trai...	2.270546	[elon]
0x495f947276749ce646f68ac8c248420045cb7b5e-107628504515518689987342262222228454948876366026517717055271263434104085938177	[bobblehead, 3, elon, musk, mesmerizer, theme,...	bobblehead 3 elon musk mesmerizer theme traits...	2.270546	[elon]

Fig. 3. Sample output of Generated Recommendations *(self-composed)*

generate this heatmap, to make the changes in scores clearly visible for as much items as possible. Additional heatmaps that were generated have been placed in the Appendix of this paper under *Extended Testing & Evaluation of the Model*

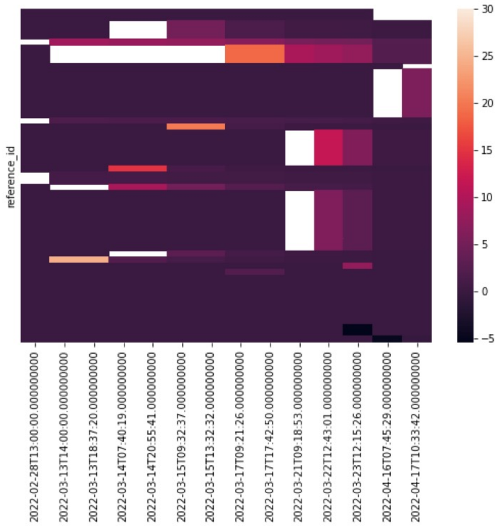


Fig. 4. Trends based Recommender - Trend Score Heatmap *(self-composed)*

7

The matrix that was generated by calculating each trend-score for each NFT on each datetime of the collected trends can be seen in Figure 6.

When taking a look at the heatmap in Figure 4, several observations can be made related to the expected output. It clearly shows how the trend-score gradually decreases with time, as the trend gets older. The decrease in scores in time can be understood better by the annotated heatmap in Figure 7. High-impact trends stay relevant for a longer period of time. Sometimes even better than those matched on the same day. This makes sense since a highly impactful topic is expected to be in peoples' minds for a longer period of time.

Although the trends data matches may be low as it depends on the kind of descriptions of items used for recommendations, another trends recommender could help identify possible interests that a marketplace admin/ creator/ seller could identify for future item additions to an e-commerce platform. This could show the impact of the trend. Even though new matching items could be added over the course of several days, that wouldn't affect the usability of the trend-score calculation algorithm as the number of days since the trend happened is considered for the final trend-score.

6 CONCLUSION

In this research, the author had identified the lack of Recommendation Systems for NFTs. Social media trends data appeared to be a valuable source of getting real-time global trends. An attempt to infuse these trends data into generating valid feature recommendations was attempted.

The trends-based recommender, *NFT-Trends-RecSys* explored in this research is expected to enhance Content-based Recommendation Systems with Collaborative-filtering-like capabilities, while preserving user anonymity & without collecting user click-data. The suggested model architecture was tested to identify if it was possible to recommend trending, timely items to users without collecting user-specific information. The results shown are very promising since all the data collected & used for the research was entirely random & arbitrary.

The data extraction methods explored for recommending NFTs, integration of social trends into recommendations & the aggregation algorithm of recommendations utilizing ensembled models are novel results yielded by this research.

Many possibilities appeared after the conclusion of this research as mentioned in the Future Work section that could be used as a stepping stone to creating even more interesting & utilitarian Recommendation Architectures in the future. With the NFTs expected to be embraced by digital systems & the internet of the next decade, the outcome of this research & the invented algorithm could be built-upon to make recommendations as good as those of the last decade.

7 FUTURE ENHANCEMENTS & NOVEL POSSIBILITIES THAT EMERGED

The current solution does a string match with keywords of each item. This may cause some matches to be skipped due to appearing in different forms. The NLP technique, lemmatization could be a possible solution for this. Name Entity Recognition is another NLP technique that could enhance the quality of trends data used. The significance of introducing such techniques will have to be tested since they may not have a significant impact on the output as most trends appear to be nouns.

Using multiple sources of trends data would be the first thing the author suggests as this would easily add to the quality and quantity of the generated recommendations. In the case of NFTs, Reddit & Discord could be identified as the next 2 best options. Google Trends data & possibly Search data could be value-addition as well. Furthermore, this can be applied to a localized forum or feedback received in the form of comments on e-commerce sites.

One of the short-comings to help match trends for this purpose that the author of this research noticed is that Twitter trends contain hashtags as trends names at times. Either the developers from the end of Twitter could give a possible solution to it or hashtags may have to be pre-processed and separated.

Due to the lack of NFT data, a DL based approach could not be attempted in this research. As a substitute or addition to recently released movies, Amazon's DL Neural Network Model could make use of trends, maybe to bolster recommendations for movies as well as e-commerce items.

The trends could be categorized to identify similar trends that users seem to show interest in. It would be almost impossible to attempt this level of personalization without collecting user data. Therefore, the value of such an attempt may have to be justified.

The trends-based recommender could act as a Decentralized Recommendations System to provide trending recommendations of NFT assets since the trends and items can come from two different sources. It would be interesting to build a peer-to-peer Recommendation Network that could support this.

The suggested solution to integrate social media trends into recommendations could also help address the cold-start problem in a distributed computing environment or when used as a SASS product where a store gives its items with descriptions and requests for recommendations from a third-party that has social trends data.

ACKNOWLEDGMENTS

To Twitter & OpenSea development teams for providing API-keys for free API access & data.

REFERENCES

[1] Smriti Ayushi and Badri Prasad. 2018. Cross-Domain Recommendation Model based on Hybrid Approach. *International Journal of Modern Education and Computer Science*, 10, 11, (Nov. 2018), 36–42. DOI: 10.5815/ijmecs.2018.11.05.

[2] Silvia Bartolucci and Andrei Kirilenko. 2020. A model of the optimal selection of crypto assets. *Royal Society Open Science*, 7, 8, (Aug. 2020), 191863. DOI: 10.1098/rsos.191863.

[3] [n. d.] Cardiffnlp/twitter-roberta-base-sentiment · Hugging Face. https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment. ().

[4] Rung-Ching Chen and Hendry. 2019. User Rating Classification via Deep Belief Network Learning and Sentiment Analysis. *IEEE Transactions on Computational Social Systems*, 6, 3, (June 2019), 535–546. DOI: 10.1109/TCSS.2019.2915543.

[5] Li Chen Cheng and Ming-Chan Lin. 2020. A hybrid recommender system for the mining of consumer preferences from their reviews. *Journal of Information Science*, 46, 5, (Oct. 2020), 664–682. DOI: 10.1177/0165551519849510.

[6] Sylve Chevet. 2018. Blockchain Technology and Non-Fungible Tokens: Reshaping Value Chains in Creative Industries. *SSRN Electronic Journal*. DOI: 10.2139/ssrn.3212662.

[7] dfinzer. 2021. Announcing our $100M raise, led by a16z. (July 2021).

[8] Michael Dowling. 2021. Is non-fungible token pricing driven by cryptocurrencies? | Elsevier Enhanced Reader. https://www.sciencedirect.com/science/article/pii/S1544612321001781?via%3D (Apr. 2021). DOI: 10.1016/j.frl.2021.102097.

[9] Robert Hackett. 2021. This crypto marketplace just raised $23 million to be the 'Amazon of NFTs'. https://fortune.com/2021/03/18/nft-art-crypto-marketplace-opensea-amazon/. (Mar. 2021).

[10] Shigang Hu, Akshi Kumar, Fadi Al-Turjman, Shivam Gupta, Simran Seth, and Shubham. 2020. Reviewer Credibility and Sentiment Analysis Based User Profile Modelling for Online Product Recommendation. *IEEE Access*, 8, 26172–26189. DOI: 10.1109/ACCESS.2020.2971087.

[11] [n. d.] Introduction to smart contracts. https://ethereum.org. ().

[12] Hardesty Larry. 2019. The history of Amazon's recommendation algorithm. https://www.amazon.science/the-history-of-amazons-recommendation-algorithm. (Nov. 2019).

[13] G. Linden, B. Smith, and J. York. 2003. Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing*, 7, 1, (Jan. 2003), 76–80. DOI: 10.1109/MIC.2003.1167344.

[14] Maxim Naumov et al. 2019. Deep Learning Recommendation Model for Personalization and Recommendation Systems. *arXiv:1906.00091 [cs]*, (May 2019). arXiv: 1906.00091 [cs].

[15] [n. d.] Recommendations: What and Why? | Recommendation Systems. https://developers.google.com/machine-learning/recommendation/overview. ().

[16] Brent Smith and Greg Linden. 2017. Two Decades of Recommender Systems at Amazon.com. *IEEE Internet Computing*, 21, 3, (May 2017), 12–18. DOI: 10.1109/MIC.2017.72.

[17] Tom Vanderbilt. [n. d.] The Science Behind the Netflix Algorithms That Decide What You'll Watch Next. *Wired*.

[18] 2020. What are you missing? Using basic machine learning to predict and recommend NFTs with OpenSea data - OpenSea Blog. https://opensea.io/blog/analysis/predict-and-recommend-nfts/. (Jan. 2020).

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

[20] Wenping Zhang, Mengna Xu, and Qiqi Jiang. 2018. Opinion Mining and Sentiment Analysis in Social Media: Challenges and Applications. In *HCI in Business, Government, and Organizations*. Vol. 10923. Fiona Fui-Hoon Nah and Bo Sophia Xiao, (Eds.) Springer International Publishing, Cham, 536–548. ISBN: 978-3-319-91716-0. DOI: 10.1007/978-3-319-91716-0_43.

A EXTENDED TESTING & EVALUATION OF THE MODEL

	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

Fig. 5. Trends based Recommendations Trends Matches Data *(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-110044771230368441551571992070040963530623495929530335175865263817154854125569	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-107628504515518668998734226222222845494887636602651771705527126343330045743104 01	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-107628504515518668998734226222222845494887636602651771705527126343410408593817 7	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-107628504515518668998734226222222845494887636602651771705527126343740262082150 5	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-107628504515518668998734226222222845494887636602651771705527126343630310919372 9	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-107628504515518668998734226222222845494887636602651771705527126341761141152153 7	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]

Fig. 6. Trends based Recommendations Heatmap Data Matrix *(self-composed)*

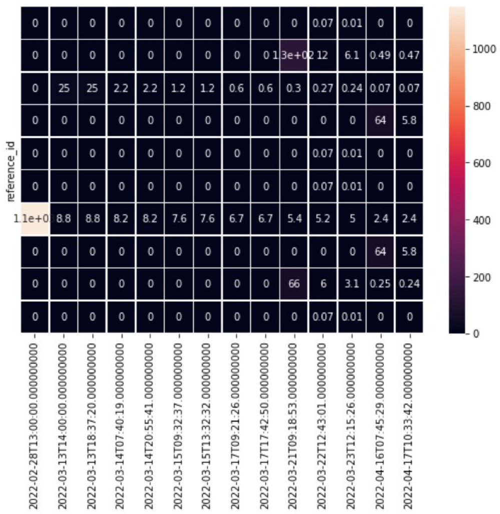


Fig. 7. Trends based Recommender Testing Annotated Heatmap - 10 random items *(self-composed)*

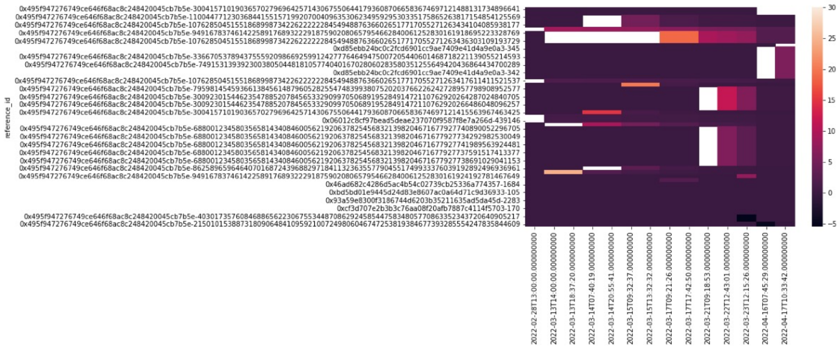


Fig. 8. Trends based Recommender Testing Heatmap - max score 30 *(self-composed)*

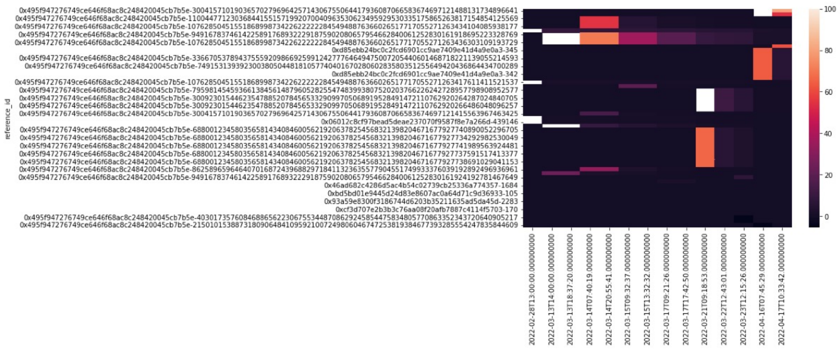
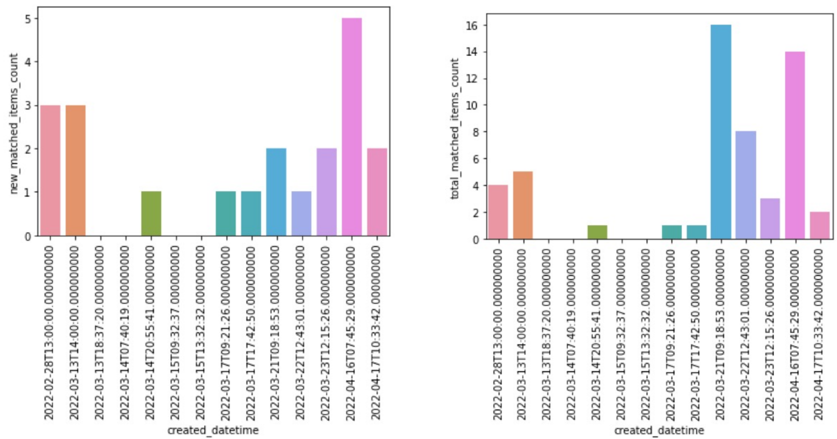


Fig. 9. Trends based Recommender Testing Heatmap - max score 100 *(self-composed)*

Fig. 10. Trends based Recommender Testing Heatmap - All items *(self-composed)*



(a) Trends based Recommendations Newly Matched Items (b) Trends based Recommendations Total Matched Items

Fig. 11. Trends based Recommendations Matched Item Counts *(self-composed)*

APPENDIX - REVIEW PAPER

A Review on Pushing the Limits of Baseline Recommendation Systems with the integration of Opinion Mining & Information Retrieval Techniques

DINUKA RAVIJAYA PIYADIGAMA, University of Westminster, UK
GUHANATHAN PORAVI, Informatics Institute of Technology, Sri Lanka

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 differs, the required type of recommendations also differs for 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 researchers to achieve better quality recommendations.

Even though each of these methods has proven to perform well individually, there have been attempts to push the boundaries of their limitations. Following a wide range of methods, researchers have tried to expand on the capabilities of standard recommendation systems 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.

This paper is a review of the novel models & architectures of hybrid Recommendation Systems. The author identifies possibilities of expanding the capabilities of baseline models & the advantages and drawbacks of each model with selected use cases in this review.

CCS Concepts: • **Information systems → Recommender systems**; *Retrieval models and ranking*; *Data mining*; • **Human-centered computing → Social recommendation**; • **Applied computing** → Online shopping.

Additional Key Words and Phrases: Recommendation Systems, Collaborative Filtering, Content based Filtering, Hybrid Recommendation Systems, Opinion Mining

ACM Reference Format:
Dinuka Ravijaya Piyadigama and Guhanathan Poravi. 2022. A Review on Pushing the Limits of Baseline Recommendation Systems with the integration of Opinion Mining & Information Retrieval Techniques. 1, 1 (May 2022), 10 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In the modern-day age, Recommendation Systems play a vital role in almost every B2C and B2B system. These systems aid in the resolution of the problem of information overload.

Recommending items for purchase, displaying personalized recommendations for users to watch videos/ movies, displaying advertisements to users, displaying personalized recommendations for online profiles and content on social networks, displaying the most likely-to-use tools/ software in a system are all done using Recommendation Systems. In 2018 it was estimated that 35% of Amazon's revenue [17] is driven by Recommendation Systems. 75% of Netlfix

Authors' addresses: Dinuka Ravijaya Piyadigama, drpiyadigama@gmail.com, University of Westminster, 309 Regent St., London, UK; Guhanathan Poravi, Informatics Institute of Technology, 57 Ramakrishna Rd, Colombo 06, Sri Lanka, guhanathan.p@iit.ac.lk.

viewer activity [20] was also said to come from recommendations back in 2013. There are many types of standard recommendation algorithms & systems made to cater to these use cases.

Among the many types of recommendation systems, item-to-item Collaborative filtering has been the most successful technique, 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 were introduced. More recently; Deep Learning methods have been brought forward by researchers to achieve better quality recommendations.

Even though each of these methods has proven to perform well, there have been attempts to push the boundaries of their limitations. Following a wide range of methods, researchers 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 ensembling models or taking hybrid approaches when building models and architectures for Recommendation Systems.

2 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** [16] has been the most successful technique for an extended period of time [19], 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 [13] were introduced.

3 DEEP LEARNING-BASED RECOMMENDATION TECHNIQUES

In 2019, **Facebook** open-sourced a new categorical data-driven **Deep learning-based recommendation engine** [17, 21]. This recommendation model was developed from the two perspectives of recommendation systems and predictive analytics. It made use of embeddings, two Multilayer Perceptrons (MLPs), one sigmoid function, [12] and a parallelization scheme to support large scales of data.

More recently, after many attempts to go beyond the gold standard of recommendation systems [16, 19] with the use of deep learning techniques, Amazon finally has achieved to use an *"Auto-Encoder"* Deep Neural Network to give better movie recommendations [15].

4 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 (NLP) domains [6]. 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 [8] 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 an impactful improvement in many cases. As a result, much of the work related to Recommendation Systems using Deep learning techniques has 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 [8]. In a more recent study, researchers tried to understand the similarities and advantages of using **MLP** versus **dot product** [18]. Similar to many Deep learning approaches, it was understood that MLP wasn'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.

5 HOW TO CHOOSE THE IDEAL ALGORITHM FOR A RECOMMENDATIONS SYSTEM?

Generally, an 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.

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

$$y = \varphi(\sum_{i=1}^{n} w_i x_i + b) = \varphi(w^T x + b) \tag{2}$$

The equation 1 is used for the calculation of the dot product between two items' similarities, while the equation 2 is used in a single-layer perceptron, where w denotes the vector of weights, x is the vector of inputs, b is the bias and φ is the non-linear activation function.

A variation that combines the MLP with a weighted dot product model, named ***Neural Matrix Factorization (NeuMF)*** has also been explored. But, that too is deemed to be outperformed by the dot product method.

One of the major limitations identified related to dot products 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.

6 USER OPINION & SENTIMENT AWARE RECOMMENDATION SYSTEMS

"Users usually transmit their decisions together with emotions."[4]

User emotions are an important factor to be considered when trying to get a better understanding of the probable decisions of a user. Sentiment analysis of user-generated content can be ideal to provide users with better recommendations. Opinion mining is a process to identify another person's viewpoint on something. Sentiment analysis is to extract someone's attitude or feeling [23]. Both these measures can be considered to be important in understanding a user's opinion related to an item.

One of the easiest ways to capture the opinions of users is to use content generated on social media. It's a cheap, fast, and effective way of capturing user opinions.

6.1 Extracting User Review Sentiment for Recommendations

There have been studies to understand the influence of user sentiment on the use of user reviews. Sentiment Analysis techniques have been applied for the purpose of understanding users' opinions related to movies that they have previously watched, in order to understand the user's preference profile. In previous research, [5] a framework that is capable of summarizing an aggregated list of historical reviews given by a user has been introduced. Later, these results are combined together with a Collaborative Filtering algorithm. This overcomes the problem of 'data sparsity' which occurs as a result of depending on user ratings. Another advantage that this method provides is that it enables the system to help creators identify the preferences of the movie consumers.

One of the major concerns that this method seems to have is that while it will be able to give appropriate recommendations to a user, the recommendations will most likely contain old movies that have been watched by other users as well. The system will have a difficulty in identifying new movies that are highly trending.

The strategy of the framework that has been adapted here has not considered different aspects of reviews that users may place. A mentioned example is *the user might focus on the quality of the sound effects in action movies, but the storyline in dramas.* The semantic strategy of opinion extraction is noted as an area to be worked on. Another limitation that has been mentioned in relation to this framework is that it does not consider slang, irony, or sarcasm, although these styles of semantics have the ability to completely change the person's opinion.

Because the proposed method is highly reliant on the technique of text-mining user reviews, the final recommendations could have a positive influence if greater attention is paid to research on text-mining models and relevant procedures.

6.2 Cross-Domain Recommendations with Decision-making Support based on Twitter Sentiment

6.2.1 A Cross-Domain Hybrid Recommendation System. When it comes to recommending items from multiple domains, it becomes challenging to use the same recommendation model for recommendations. Each of these domains may have distinctive features as well as a varied bias in weights of each feature towards recommendations. Taking into consideration of the 3 domains music, movies & books a group of researchers has been able to produce a cross-domain recommendations system [1]. This work has focused on using the domain knowledge gained from movies to generate recommendations for books and music.

In order to build the required system, the authors have tested various supervised classification algorithms together with a hybrid approach for recommendations with the combination of content-based recommendations, user-to-user Collaborative filtering, and personalized recommendation techniques. Out of the tested classifiers, the Decision tree classifier was found to give the highest accuracy.

The system that was produced in this research was able to address the limitation of data sparsity and the cold start problem that occurs with single domain analysis. With the integration of several domains, the system has shown the capability of generating a higher accuracy in suggestions.

6.2.2 Using Twitter Sentiment for Validation of Recommendations. The authors have then taken the extra step of using Twitter sentiment analysis on the generated recommended entities. One of the points that can be taken out of this is that public sentiment on social media is a consideration that users show interest in when choosing an item to consume, even after getting it as a recommendation. This drills down to the natural human desire to get validation on consumables by the people around them and from those that they look up to. Taking this into consideration for a recommendation system is, therefore, a positive aspect, especially for systems that don't have the luxury of integrating directly with a large social network such as Facebook/ Twitter in order to generate direct recommendations for its users.

The overall system uses this as a decision-support system to provide the user with decision making by visualizing the positive, negative & neutral polarity percentages given by people, on Twitter. While it is clear that such decision support is valuable to the user, it feels ironic to recommend an item to a user and then say that it's not well-accepted by the community. While helping the user identify the recommended items based on other users' online activity, it may also say that it's not recommended, or deemed with negative sentiment by the majority of the community. This may leave the user in the confusion about what to take. It also leaves the part of the work the system can do for the user. A better method to integrate this public sentiment into recommendations would be to utilize the sentiment scores on social media to harvest the ideal recommendations and then show them to the user. Items with only positive sentiment can be considered.

6.3 Identifying Possible Classification Techniques of User Reviews

While procedures relevant to text-mining play an important role in understanding users' opinions, the classification of these reviews also has to be done appropriately, in order to be used for recommendations.

The emotional information that users provide with their comments has the potential to influence the correctness and precision of recommendations. In the work done [4], a deep learning model has been used to process user comments to generate a possible rating for recommendations.

Sentiment analysis is applied to the reviews to create a feature vector. Then, a noise reduction procedure is implemented on the data set to delete short comments, comments with no expression, and false rating comments. This has been done to improve the classification of ratings, as it has been pointed out in previous literature[5] as being highly reliant on generating recommendations based on sentiment analysis of user-generated content. Finally, a Deep Belief Network has been used to achieve data learning for the recommendations.

This **Deep Belief Network and Sentiment Analysis (DBNSA)** has been said to outperform baseline models, especially training loss value, precision, and recall on Yelp and Amazon data sets. Furthermore, it is said to save more time than other baseline methods. The biggest drawback that this method seems to have is that the algorithm is not suitable for real-time testing. Furthermore, social relationships and subsequent timeline comments have also been identified as a possible extension of this work since they can help address the cold-start problem by using timeline comments from a user's close social relationships. This might be a little difficult to implement by integrating externally into social networks since a user's close group of friends who think alike will have to be known to get the ideal information that will affect a particular user's decision-making process.

There has been previous research focused on devising a robust recommendation methodology by identifying the credibility of reviewers and the quality of reviews when taking into consideration of all item reviews [14]. Sentiment analysis captured from reviews is an additional enhancement that this model uses apart from identifying the factors that affect a user's fondness for a certain product. This has been identified as the first work that integrates credibility-driven feature-based fine-grained sentiment analysis with user modeling for online product recommendations.

The entire system which is named credibility, interests, and sentiment enhanced recommendation (CISER) has five sub-modules. Candidate feature extraction is done using the *spaCy* library, while sentiment confidence is given by *fastText*. Thereafter, reviewer credibility analysis, user interest mining, candidate feature sentiment assignment, and recommendation module follow. When the proposed system was tested with Amazon's camera review data-set, it managed to outperform baseline models. This shows that the more specifics we look into when considering items for recommendations, the better the Recommendations System is able to perform.

The paper suggests devising sophisticated measures for representative rating and expertise as possible future enhancements that can be done to increase the specifics of identifying reviewer credibility. Furthermore, social network information and online activity logs can be considered for the user credibility model.

7 BREAKDOWN OF RECOMMENDATION SYSTEM ARCHITECTURES 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 figure1, the author tries to elaborate on the possible technical contribution brought forward in this research.

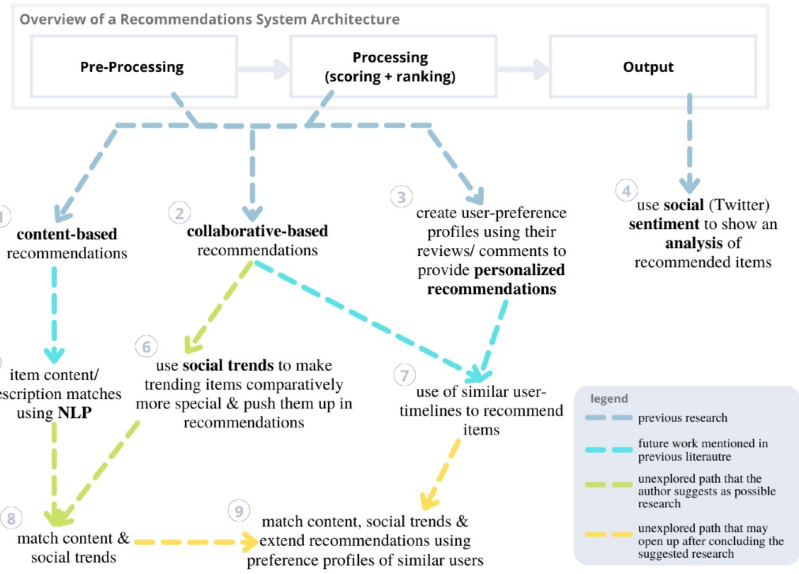


Fig. 1. Enhancements done to Recommendation Systems using opinion mining techniques *(self-composed)*

The figure 1 shows the identified possible points of integration of opinion mining techniques into a Recommendations System. 1, 2 [16, 15], 3 [5] & 4 [1] techniques have been already applied as identified in past literature, while the 7th technique has been mentioned as possible future work from the 3rd technique [4]. Method 5 hasn't been explicitly attempted in recent literature concerning 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 taken care of.

Method 6 has not been identified in previous literature and is expected to align better with the desires circulating market places that don't directly track & collect user input such as user clicks. 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.

8 NLP TECHNIQUES THAT CAN BE APPLIED TO SUPPORT THE 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 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), point 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 [22]. Transformer models particularly trained to analyze Twitter Sentiment can be identified to be highly accurate & fast with classifying opinion mining data from social media [3, 2].

9 COMMON CHALLENGES FACED BY SENTIMENT-AWARE RECOMMENDATIONS SYSTEMS

One of the most common challenges faced by Recommendation Systems that wish to integrate sentiment analysis into any part of their architecture is the inability to filter out and classify sarcastic comments. This is an area of NLP that needs to be researched further together with the ability to classify the relevant comments.

Extracting all this useful information in real-time in large volumes can be very challenging. This is why several of these systems appear to fail to provide recommendations in real-time.

10 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 [23]. When doing so, it would be good to have an impact score for each document to make sure that recommendations are biased appropriately towards the opinions of the majority with consideration of the users' opinions.

11 EVALUATION APPROACHES FOR RECOMMENDATION SYSTEMS

As highlighted in past reviews, evaluating & benchmarking Recommendation Systems has been a major concern due to the lack of available datasets and questions related to domain-specific approaches/ algorithms used for recommendations.

For the convenience of future work, the following breakdown can be used to evaluate future Recommendations Systems. Especially, those that integrate novel algorithmic, hybrid approaches for recommendations.

When evaluating Recommendation Systems, we may examine the outcomes produced by the system in two ways. The first way would be to identify 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 evaluate the outcome can be done by 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, *Precision@K (P@K)*, also identified as the *Top-N strategy* in several pieces of 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 1. 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.	

Mean Absolute Error (MAE) & Root Mean Squared Error (RMSE) is 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 [10].

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 produced 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, [8] the author mentions that the lack of used datasets and code bases hinders the ability to properly benchmark and evaluate new research related to Recommendation Systems. The importance of reproducibility of research related to Recommendations Systems has future been elaborated on in the reviews that follow [7, 11, 9].

12 CONCLUSION

In this review, the author has pointed out the reasons to as why baseline Recommendation Systems have been attempted to be improved with the use of ensemble techniques & hybrid models. Several attempts that past researchers have taken towards implementing such Recommendation models were also discussed and critically evaluated.

Since this review has covered many opinion mining-related work, the NLP techniques that can be applied to these, the common challenges faced and practices that could be followed to optimize the use of gathered opinions for recommendations purposes have been noted. Finally, since evaluating Recommendation Systems has also not been clearly handled in the past, the author has included a summary of several evaluation metrics that can be adopted together with each of their objected orientations.

REFERENCES

[1] Smriti Ayushi and Badri Prasad. 2018. Cross-domain recommendation model based on hybrid approach. *International Journal of Modern Education and Computer Science*, 10, 11, (Nov. 8, 2018), 36–42. DOI: 10.5815/ijmecs.2018.11.05.

[2] Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa-Anke. 2020. TweetEval: unified benchmark and comparative evaluation for tweet classification. *arXiv:2010.12421 [cs]*, (Oct. 26, 2020). Retrieved May 1, 2022 from http://arxiv.org/abs/2010.12421 arXiv: 2010.12421.

[3] [n. d.] Cardiffnlp/twitter-roberta-base-sentiment · hugging face. Retrieved May 1, 2022 from https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment.

[4] Rung-Ching Chen and Hendry. 2019. User rating classification via deep belief network learning and sentiment analysis. *IEEE Transactions on Computational Social Systems*, 6, 3, (June 2019), 535–546. Conference Name: IEEE Transactions on Computational Social Systems. DOI: 10.1109/TCSS.2019.2915543.

[5] Li Chen Cheng and Ming-Chan Lin. 2020. A hybrid recommender system for the mining of consumer preferences from their reviews. *Journal of Information Science*, 46, 5, (Oct. 2020), 664–682. DOI: 10.1177/0165551519849510.

[6] Minjin Choi, Yoonki Jeong, Joonseok Lee, and Jongwuk Lee. 2021. Local collaborative autoencoders. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining* (WSDM '21). Association for Computing Machinery, New York, NY, USA, (Mar. 8, 2021), 734–742. ISBN: 978-1-4503-8297-7. DOI: 10.1145/3437963.3441808.

[7] Maurizio Ferrari Dacrema, Simone Boglio, Paolo Cremonesi, and Dietmar Jannach. 2021. A troubling analysis of reproducibility and progress in recommender systems research. *ACM Transactions on Information Systems*, 39, 2, (Jan. 6, 2021), 20:1–20:49. DOI: 10.1145/3434185.

[8] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. 2019. Are we really making much progress? a worrying analysis of recent neural recommendation approaches. *Proceedings of the 13th ACM Conference on Recommender Systems*, (Sept. 10, 2019), 101–109. arXiv: 1907.06902. DOI: 10.1145/3298689.3347058.

[9] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. 2020. Methodological issues in recommender systems research (extended abstract). In Twenty-Ninth International Joint Conference on Artificial Intelligence. Vol. 5. ISSN: 1045-0823. (July 9, 2020), 4706–4710. DOI: 10.24963/ijcai.2020/650.

[10] Aviram Dayan, Guy Katz, Naseem Biasdi, Lior Rokach, Bracha Shapira, Aykan Aydin, Roland Schwaiger, and Radmila Fishel. 2011. Recommenders benchmark framework. In *Proceedings of the fifth ACM conference on Recommender systems - RecSys '11*. the fifth ACM conference. ACM Press, Chicago, Illinois, USA, 353. ISBN: 978-1-4503-0683-6. DOI: 10.1145/2043932.2044003.

[11] Maurizio Ferrari Dacrema, Federico Parroni, Paolo Cremonesi, and Dietmar Jannach. 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). Association for Computing Machinery, New York, NY, USA, (Oct. 19, 2020), 355–363. ISBN: 978-1-4503-6859-9. DOI: 10.1145/3340531.3411901.

[12] C. Freudenthaler, L. Schmidt-Thieme, and Steffen Rendle. 2011. Factorization machines factorized polynomial regression models. Retrieved Aug. 27, 2021 from https://www.semanticscholar.org/paper/Factorization-Machines-Factorized-Polynomial-Models-Freudenthaler-Schmidt-Thieme/10e8617b6c58d599b8d17b3a60047a599bfe5e72.

[13] G. Geetha, M. Safa, C. Fancy, and D. Saranya. 2018. A hybrid approach using collaborative filtering and content based filtering for recommender system. *Journal of Physics: Conference Series*, 1000, (Apr. 2018), 012101. Publisher: IOP Publishing. DOI: 10.1088/1742-6596/1000/1/012101.

[14] Shigang Hu, Akshi Kumar, Fadi Al-Turjman, Shivam Gupta, Simran Seth, and Shubham. 2020. Reviewer credibility and sentiment analysis based user profile modelling for online product recommendation. *IEEE Access*, 8, 26172–26189. Conference Name: IEEE Access. DOI: 10.1109/ACCESS.2020.2971087.

[15] Hardesty Larry. 2019. The history of amazon's recommendation algorithm. Amazon Science. Section: Latest news. (Nov. 22, 2019). Retrieved May 25, 2021 from https://www.amazon.science/the-history-of-amazons-recommendation-algorithm.

[16] G. Linden, B. Smith, and J. York. 2003. Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing*, 7, 1, (Jan. 2003), 76–80. DOI: 10.1109/MIC.2003.1167344.

[17] Maxim Naumov et al. 2019. Deep learning recommendation model for personalization and recommendation systems. *arXiv:1906.00091 [cs]*, (May 31, 2019). Retrieved Apr. 29, 2021 from http://arxiv.org/abs/1906.00091 arXiv: 1906.00091.

[18] Steffen Rendle, Walid Krichene, Li Zhang, and John Anderson. 2020. Neural collaborative filtering vs. matrix factorization revisited. In *Fourteenth ACM Conference on Recommender Systems* (RecSys '20). Association for Computing Machinery, New York, NY, USA, (Sept. 22, 2020), 240–248. ISBN: 978-1-4503-7583-2. DOI: 10.1145/3383313.3412488.

[19] Brent Smith and Greg Linden. 2017. Two decades of recommender systems at amazon.com. *IEEE Internet Computing*, 21, 3, (May 2017), 12–18. Conference Name: IEEE Internet Computing. DOI: 10.1109/MIC.2017.72.

[20] Tom Vanderbilt. [n. d.] The science behind the netflix algorithms that decide what you'll watch next. *Wired*. Section: tags. Retrieved Aug. 27, 2021 from https://www.wired.com/2013/08/qq-netflix-algorithm/.

[21] 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. (Feb. 7, 2019). Retrieved May 15, 2021 from https://ai.facebook.com/blog/dlrm-an-advanced-open-source-deep-learning-recommendation-model/.

[22] Thomas Wolf et al. 2020. Transformers: state-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Online, (Oct. 2020), 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6.

[23] Wenping Zhang, Mengna Xu, and Qiqi Jiang. 2018. Opinion mining and sentiment analysis in social media: challenges and applications. In *HCI in Business, Government, and Organizations*. Vol. 10923. Fiona Fui-Hoon Nah and Bo Sophia Xiao, (Eds.) Series Title: Lecture Notes in Computer Science. Springer International Publishing, Cham, 536–548. ISBN: 978-3-319-91716-0. DOI: 10.1007/978-3-319-91716-0_43.