Hybrid deep learning based music recommendation system

# M.Sunitha1, Dr. T. Adilakshmi2, Mehar Unissa3 1,2, 3 Vasavi College of Engineering, Hyderabad, India

1 [m.sunithareddy@staff.vce.ac.in,](mailto:m.sunithareddy@staff.vce.ac.in) 2 [t\_adilakshmi@staff.vce.ac.in,](mailto:t_adilakshmi@staff.vce.ac.in) 3 [unissamehar6@gmail.com](mailto:unissamehar6@gmail.com)

#### Abstract

#### The fundamental objective of musical recommendations is to propose songs that are appropriate to the tastes of the user. In this paper We have developed, implemented, and analyzed music recommendation systems with variations of music recommendation algorithm using MLP neural network such as content-based model using MLP neural network, collaborative model using MLP neural network and hybrid model using MLP neural network. We have also looked into the accuracy and precision of all the three algorithms with various activation function.

#### Keywords—Content-based filtering; Streaming history; Collaborative filtering; Hybrid recommendation; Deep learning; Music recommendations.

# **Introduction**

The digitization of music has had a huge effect on the music industry. This transition was also brought about by the advent of the Internet, since it used to be both the source of digital music and its distribution route. A tremendous amount of music was thereby made available. This is an issue in the current age of digital music distribution that restricts the scope of music. We already face the new music-based paradigm: listeners are already provided with instantly accessing unprecedentedly large digital music libraries. The majority of music recordings are online, with the quantity of digital music growing and counting tens of millions. Major Internet shops like the iTunes Store have up to 28 million music each month adding thousands of new music. This is not surprising as the music plays a vital role in everyday living, and more and more people use modern technologies to express and share their creativity in music. The music advisor system is a system that learns from the history of the users and recommends songs they'd want to hear probably in future. In order to construct an efficient recommending system, we have incorporated numerous algorithms.

Whilst email has early embraced the benefits of the use of advisor systems, the music domain is impacted by offline radio stations where static playlists are transmitted to every listener on the basis of track popularity and expert pre-selections. The balance has shifted with the emergence of music streaming systems, such Last.fm1 or Spotify2, and users can now build their own personal radio stations. In turn, consumers are now obliged to construct their own playlists and discover new music less likely. An elegant complement that makes use both of the knowledge of the audience and of the user's previous listening experience is a musical advisory system. As the Internet is now widely used, one or more recommendation systems have been found among the vast majority of the computer, tablet and Smartphone users. For instance, you can envision a visit to a favorite online shop to explore a specific item. After you find this product and click on the direct link, the page can contain a section named, "Customers who have also purchased this item." These objects are listed in relation to the product under review as potentially interesting items. A customized list of recommendations will be provided for registered users automatically when they log on to the website. A recommendation system is the program that provides advice. Custom suggestions require a system to get some user knowledge. In other words, a user profile including the preferences of each user must establish and retain a recommendation system. These preferences of users can be expressly gained through the user's request to rate a certain item, or by user activity tracking.

# **Motivation**

The strength of the internet has enabled many users to integrate recommendation systems into their daily lives. If someone has used Facebook, LinkedIn, or even Netflix, they have a system that promotes new stuff according to different parameters. Amazon.com uses a system that recommends products based on the browsing/buying history of the user and products bought by other users with the same taste. The prominent internet services Pandora and Last.fm choose music for users. These and other web-based music applications generate revenues not previously available and help businesses grow to other markets. Music recommendation algorithms contribute to the fuel economy of digital songs, helping consumers to discover music. In 2012 digital revenues of record labels have been estimated at 9 percent higher than in 2011, according to the International Federation of Phonographic Industries (IFPI).

# **Literature Review**

**F.O. Isinkaye et. al (2015)** Recommendation algorithms open up additional options to collect tailored Internet information. It also helps to reduce the problem of overloading information, which is a very common phenomena of information recovery systems and allows consumers access to products and services not readily accessible to users on the system. In this research, two classic recommendation methodologies were explored and their merits and limitations were highlighted by various types for improving hybridization policies. [1]

**Shuai Zhang et. al (2018)** The author has presented in this essay an exhaustive evaluation of the most remarkable work on high-level learning systems. Author have recommended the organization and grouping of current articles by proposing a range of inertial research prototypes. We also addressed how deep learning techniques could be used to advise on the benefits/disadvantages. authors also discuss some of the most important open issues and promising extensions for future years. In the last few decades, the themes of hot study have been both deep learning and recommendation systems. Each year, numerous new techniques are being developed and new models are emerging. authors believe this survey will give readers with an overview of the important components of the project, clarify the most significant developments and clarify future investigations. [2]

**Thomas Hornung et. al (2018)** Music really is an area that is subjective. That is why TRecS balances generally recognized measurements such as track and tag similarity with measurements such as time similarity, which are music specific. With our serendipity metric, new music can be discovered. We have found in our empirical study that the predictive quality improves with each user's number of recommendation lists. Since all ratings are directly returned to the predictive algorithm, the model works. We also wanted to find out, based on four groups, which orchestrate of the three sub-recommenders results best. [3]

**Zeshan Fayyaz et. al (2020)** In this work, authors presented a complete RS survey that presents many sorts of RSs such as collaborative filtering, content, demo, utility, knowledge-based and hybrid. The hybrid system is also given and characterized in a variety of combination tactics in weighted, blended, switched. Their four major obstacles included cold starts, data scarcity, scalability and variety and measuring measures utilized for the assessment of the success of a recommendation system. Furthermore, authors demonstrate how e-commerce and various fields including transport, e-health, agriculture and the media have been implemented in recommendation systems. [4]

**Markus Schedl et. al (2019)** Like in many other academic fields, in music recommendation systems, deep learning (DL) is increasingly accepted (MRS). This field uses deep neural networks, primarily for the extraction of latent characteristics from audio or metadata of music items and the learning of sequential patterns of music items (paths or artists) from playlists or hearing sessions. Latent item factors are usually included in a content-based filtering or hybrid MRS, whereas music elements sequence models, e.g., automatic playlist continuation, are utilized for sequence music recommendations. This page describes specific characteristics of the RS Research music field. [6]

**Sheela Kathavate et. al (2021)** Twenty artists are involved in the experiment. In the future, authors will aim to make the recommendation stronger by adding a higher number of artists and languages to give even better playlists for people. In order to compare the findings and seek out better results, here also use the system with various learning models. Authors wanted to provide the customers a preference for what they want to hear when there are thousands of music out there and after a step closer, we were satisfied. An emotional detector system can be built for future use, which can propose the music through recognition of our face emotion. [9]

**Markus Schedl et. al (2018)** The authors have recognized numerous major obstacles to the field of music advisory systems research (MRS) throughout this trends and survey article. Among others, these are at the heart of modern MRS research. Authors have

(1) discussed the problem of cold beginning items and users with their particular features in the area of music, (2) discussed the challenge of automatic continuation of the playlist due to the recently emerged user's demand for recommended musical experiences rather than single tracks. [10]

**Andreu Vall et. al (2019)** Music advisory systems have become a significant tool to enable consumers' interaction with ever-greater music archives, online music shops and personal gadgets, for example. The autonomous continuation of music play listing allows the suggestion of music streams that adapt to certain (maybe short) listening sessions is an important task in music recommender systems. Earlier investigations showed that collaborative filtering reveals underlying play listed co-occurrence patterns that are beneficial for predicting playlist continuations for collections of precise music playlists. However, a significant long-tailed distribution exhibits in most Music collections. [11]

**Yu Wang et. al (2020)** This work proposes to recommend higher quality song sequences with a hybrid music recommendation system based on reinforcement learning (PHRR). WMF and CNN are taught to learn audio signals for the tune. Authors also provide a model-based reinforcement learning framework, which simulates listeners and model the problem of reinforcement learning as a decision-making process based on the preferences of listeners, for both the transitions of songs. Authors innovatively improve the simulation of the interaction process to update the model more data- efficiently in order to capture modest changes in listeners' preferences sensitively. Real-world data set experiments show that PHRR works better than other comparison algorithms in the music suggestion. [12]

|  |  |  |  |
| --- | --- | --- | --- |
| **Paper Tittle** | **Authors** | **Proposed Method** | **Drawbacks** |
| Location-based Orientation Context Dependent Recommender System for Users | Mr. C. Vijesh Joe, Dr. Jennifer S. Raj. (2021) [3] | This study introduces an RS that employs the smallest bounding box for every consideration, as opposed to other typical RS that rely solely on the geographic point to establish the location of an object. | The cold state problem will occur when a person who has no knowledge of social network joins the recommendation systems. This is due to the fact that using the preference of a person that has no history would result in an empty user-matrix. |
| Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities | Zeshan Fayyaz, Mahsa Ebrahimian, Dina Nawara, Ahmed Ibrahim and Rasha Kashef. (2020) [4] | In this work, authors presented a complete RS survey that presents many sorts of RSs such as collaborative filtering, content, demo, utility, knowledge-based and hybrid. The hybrid system is also given and characterized in a variety of combination tactics in weighted, blended, switched. | Their four major obstacles included cold starts, data scarcity, scalability and variety and measuring measures utilized for the assessment of the success of a recommendation system. |
| Comparison of machine learning algorithms to classify web pages. | Arthur F. et.al [5] | EcoRec, is a perspective Rating prediction scenario for users who haven't rated purchased products, employs two or more recommender algorithms.it recommend products to clients, provides an enriched user item matrix. | Data that was previously unlabeled was used and then labelled. This is the only parameter that is specified explicitly. The measure of web engagement and temporal activities aren't taken into account. |
| Chemical sensing with familiar devices. Angewandte Chemie International Edition | Orit Raphaeli et.al [7] | Considered Web engagement measures when working on clickstream data. This strategy could assist online retailers in engaging customers on their mobile devices at any time and from any location. | Only three types of web interaction metrics are evaluated out of a total of eight. |
| Recommender system based on pairwise association rules. | Timur Osadchiy et.al [8] | Using the implicit social graph to generate the pairwise association rule. it Sort the things that take the least amount of time to get to the consumer. | The timestamp isn't regarded as significant. In terms of areas and locales, there is no recommendation for a product pair. The algorithms' costs have not been investigated. |

# **Methodology**

#### Working

The algorithm makes use of an input vector that contains collaborative, content-based filtering, and streaming history information as well. This information is loaded into a deep neural network, which trained to recognize patterns in the user’s history through a training process, eventually recommends songs it believes the user will appreciate.

#### Overview

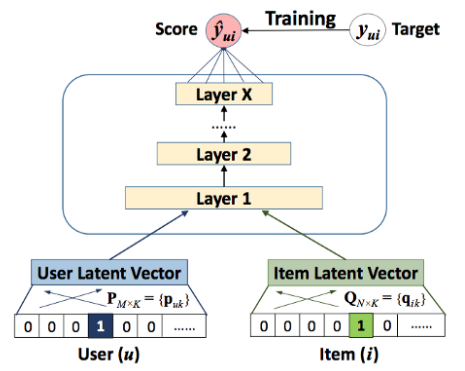
The following is a broad description of our music recommendation system: A multi-layer perceptron model takes song metadata like genre, year, playlist, tracks, and the user's streaming history other audio features as input and outputs a list of songs that the user may or may not like. It returns a score ranging from 0 to 1, indicating the likelihood that the listener would enjoy the selected song based on previous experiences. Our algorithm locates the k highest-scoring songs and suggests them to the user.

#### Content-Based Filtering

Song information is used in content-based filtering. It takes two types of metadata into account: genre and release year. This information comes from the Spotify Dataset 1922-2021, which includes the audio features of 600,000 music released between 1922 and 202. When given as input to the multi-layer perceptron neural network described below, the model can learn a content-based filtering strategy. One metadata category (genre) is unorderable, but the other (release year) can be ordered. Then, based on the numerical audio attributes of each genre, we utilized a basic K-means clustering technique to partition the genres in this dataset into ten groupings. We also use K-means from the dataset to cluster songs. The same genres will sound similar and come from similar eras, and the same may be said for the songs within those genres. We use this information to create a recommendation engine that takes the data points from the songs a user has listened to and suggests tracks that match neighboring data points. We use "Spotipy," a Python client for the Spotify Web API that makes it simple for developers to interact with it.

#### Collaborative Filtering

In order to produce a prediction, collaborative filtering focuses on user-item associations. We create a sparse one-hot encoding matrix that depicts user-item associations (tracks). We can now construct the model using the sparse user-item encoding matrix. Because there is a common in-valid playlist-track combination, we create negative training examples (in-valid playlist-track pairings). We know which music are in the playlist but not which ones aren't. The embeddings (low-dimensional) for each playlist and item are generated using the playlist (user) and item vectors. These vectors are then fed into the model as input. Our model uses multi-layer perceptron (neural net) techniques similar to those presented in Figure 1. We propose an instantiation of NCF using a multi-layer perceptron (MLP) to analyze the user–item interaction function.



#### Figure 1: Deep collaborative filtering model

* 1. **Multi-Layer Perceptron (MLP)**

As NCF uses two paths to model users and items, it's natural to concatenate their features. In multimodal deep learning research [47, 34], this design has been widely used. However, simply concatenating vectors does not account for any interactions between user and item latent features, making it unsuitable to describe collaborative filtering. To solve this problem, we propose adding hidden layers to the concatenated vector and learning the interaction between user and object latent attributes using a conventional MLP. In this method, rather than using a fixed element-wise product to learn the interactions between 𝑝𝑢and 𝑞𝑖, we can give the model a lot more flexibility and non-linearity. Our NCF framework defines the MLP model as follows:

() = ,

() = , …… (1)

() = ( + ),

= ),

where , , and are the x-th layer's perceptron's weight matrix, bias vector, and activation function, respectively. Sigmoid, hyperbolic tangent (tanh), and Rectifier (ReLU) are just a few of the activation functions available for MLP layers.We'd like to look at each function individually: 1) The sigmoid function requires each neuron to be in the (0,1) range, which may reduce the quality of the model; and it is believed to suffer with saturation, in which neurons cease to acquire when their output is around 0 or 1.2) Despite the fact that tanh is a better alternative and has been largely accepted [6, 44], it only mitigates the drawbacks of sigmoid to a limited degree, since it may be considered as a rescaled variant of sigmoid (tanh(x/2) = 2σ(x) − 1).And 3) as a result, we choose ReLU, which cease to acquire when their output is around 0 or 1.2) Despite the fact that tanh is a better alternative and has been largely accepted [6, 44], it only mitigates the drawbacks of sigmoid to a limited degree, since it may be considered as a rescaled variant of sigmoid (tanh(x/2) = 2σ(x) − 1).And 3) as a result, we choose ReLU, which may be more biologically plausible and has been shown to be non- saturated [9]; also, it favours sparse activations, making it well- suited to sparse data and reducing the likelihood of overfitting the model. Our findings suggest that ReLU outperforms tanh, which outperforms sigmoid.

#### Algorithm overview

The main aim of this study is to construct a music recommendation application. The app lets users select the songs on the device and listen to them. If a certain song is listened to by a user, a log is established. We employ many ways to create recommendation engine to offer songs to consumers. The fundamental reason for this proposed system is to enhance the capacity of the standard system of recommendations. Traditional music advice systems depend on collaborative filtering or content-based filters for recommendations to be generated. This design is to create music recommendations that are user-friendly, without significant co-use and that are useful in terms of music similarity. A large number of co-utilized music in various styles might provide diversion in the recommendations. The several random steps in both models allow various recommendations to be created, although they do not ensure innovation, given the same combinations of inputs.

***Algorithm***: Multi-Layer Perceptron for Hybrid Music Recommendation System

Input: Genre, year, playlist, tracks, streaming history, Audio features.

Dataset: Spotify Million Playlist. Output: recommendationList []; Steps:

1. Using the dataset, create a test file.
2. Using metadata as an input, create a metadata-related training file.
3. Send the multilayer perceptron the training and testing files.
4. Comparing test data for similarities and classifying
5. If they're comparable,
6. classequals to 1
7. If not,
8. class equals to 0
9. Adding songs to the matched recommendationList that have class = 1.
10. end

## Results

## Dataset

## The performance of our model is evaluated using the publicly accessible Spotify playlist dataset. To deploy the data, we employ the layered approach shown in Fig.2. This dataset is based on the selection of individuals in the #nowplaying dataset who broadcast their #nowplaying twitter via Spotify. In principle, the database comprises subscribers, tracks, and the songs that appear in those playlists. "User id," "artistname," "trackname," and "playlist name" are the necessary fields in the csv format containing the dataset, where user id is a hashed user's Spotify user name.[10]

## Performance evaluation

## The model is implemented using the Python language. The MLP model is implemented using the TensorFlow framework. We create a sparse one-hot encoding matrix that links users (playlist) and item associations (tracks). Then, by using numeric audio properties of every category, we calculated the cosine similarity. Finally, we used a standard K-means clustering approach to divide the genres in this database into ten groups. The embeddings (low-dimensional) for every playlist and item are created using playlists (more commonly termed u for user) item I vector and numeric characteristics. The dot product is used to integrate the three embeddings in Generalized Matrix Factorization (GMF) (this is the classic matrix factorization).

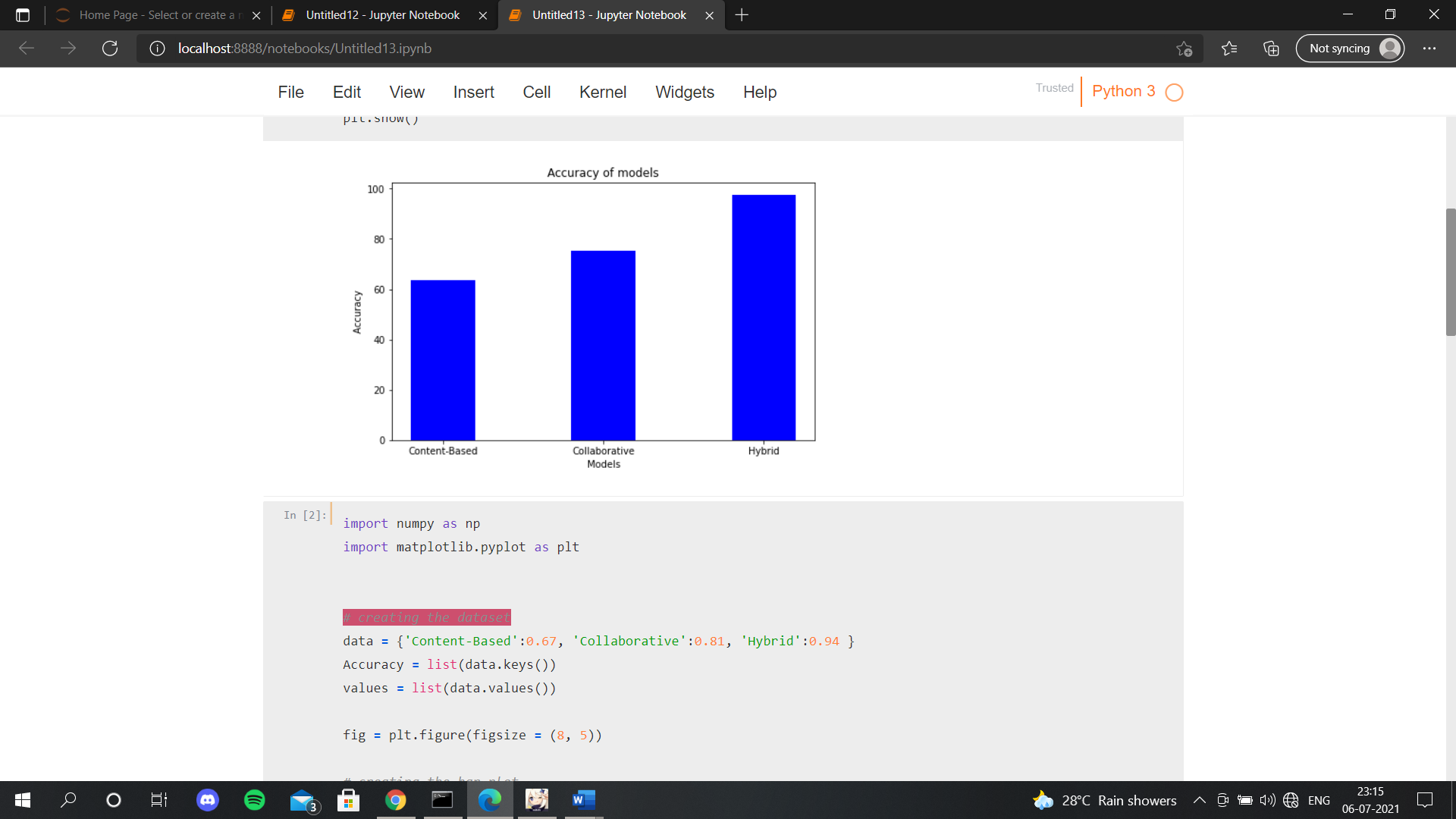
**Table 1: Sample Results**

|  |  |  |  |
| --- | --- | --- | --- |
| **User\_Id** | **Probability** | **Track\_Name** | **Track\_Artist** |
| 4962 | 0.999999 | Runaway | Timeflies |
| 3177 | 0.999999 | Ocean Away | Ellon |
| 6228 | 0.999999 | Daisy Cutter | 311 |
| 715 | 0.999995 | Elder Scroll | JAY Z |
| 9901 | 0.999995 | Little Queen | Heart |
| 10950 | 0.999994 | Pure | David Keller |
| 6091 | 0.999993 | Lazy Summer | Journey |

Above table shows what results would look like after performing hybrid recommendations. It gives the song along with the probability that the particular song will be liked by user. We compare various methods using MLP Network such as content-based model using mlp neural network, collaborative model using mlp neural network and hybrid model using mlp neural network.

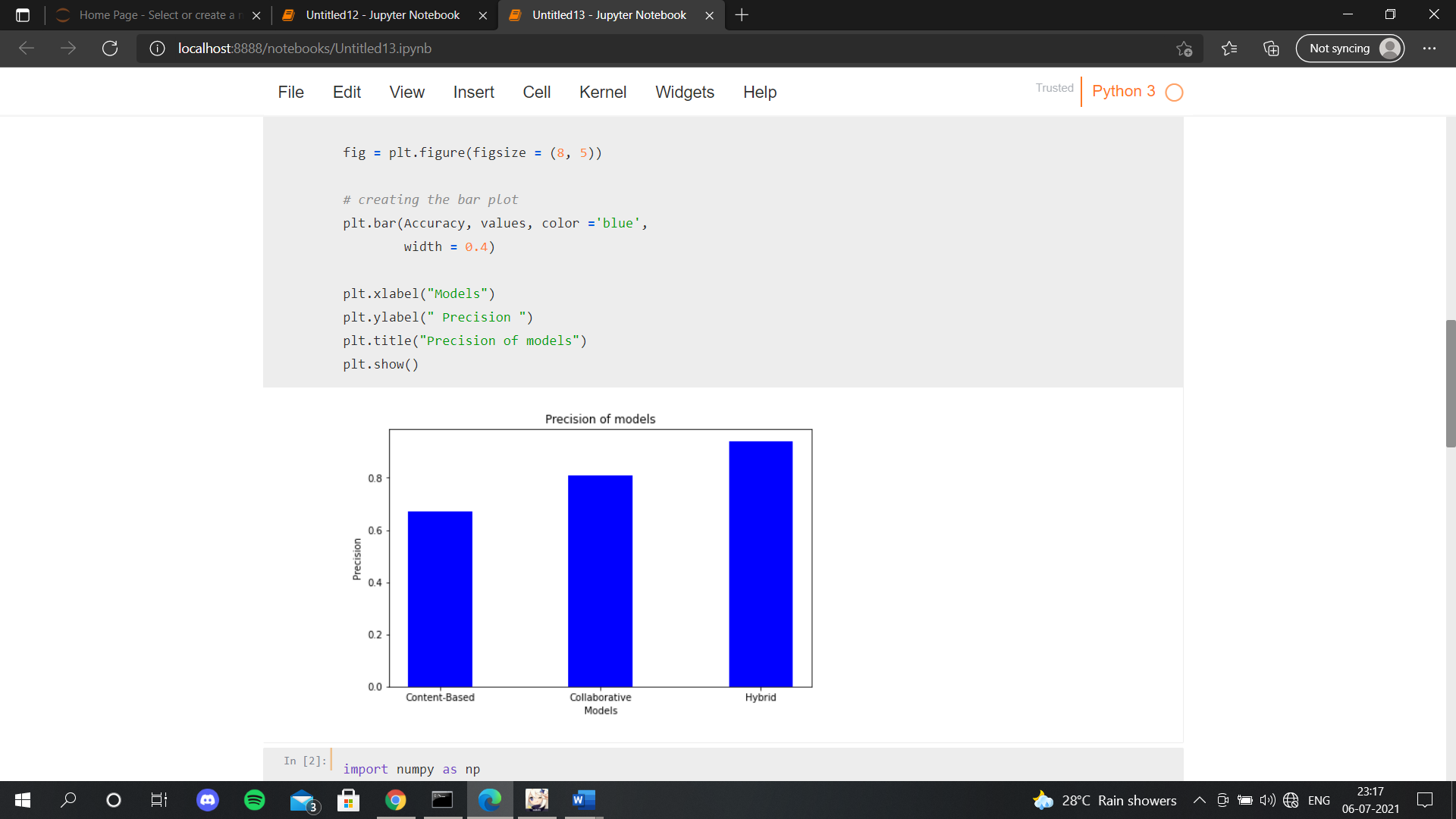
**Table 2: Comparison of Models**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | | |  | |  | |  | |  | |  | |
|  | **Content-Based Filtering using MLP** | | | | **Collaborative Filtering using MLP** | | | | | | **Hybrid Model using MLP** | | | | | |
| **Accuracy** |  | 63.50% |  |  |  | 75% | |  | |  | |  | | 97.50% | |  | |
| **Precision** |  | 0.67 |  |  |  | 0.81 | |  | |  | |  | | 0.94 | |  | |
|  |  |  |  |  |  |  | |  | |  | |  | |  | |  | |
|  |  |  |  |  |  |  | |  | |  | |  | |  | |  | |



**Figure 2: Accuracy of Models**

As we can see accuracy of hybrid model is more at 97% when compared to Collaborative and content-based model. It tends to give more accurate results then the other two. Whereas content-based model is the lowest with 64% accuracy.

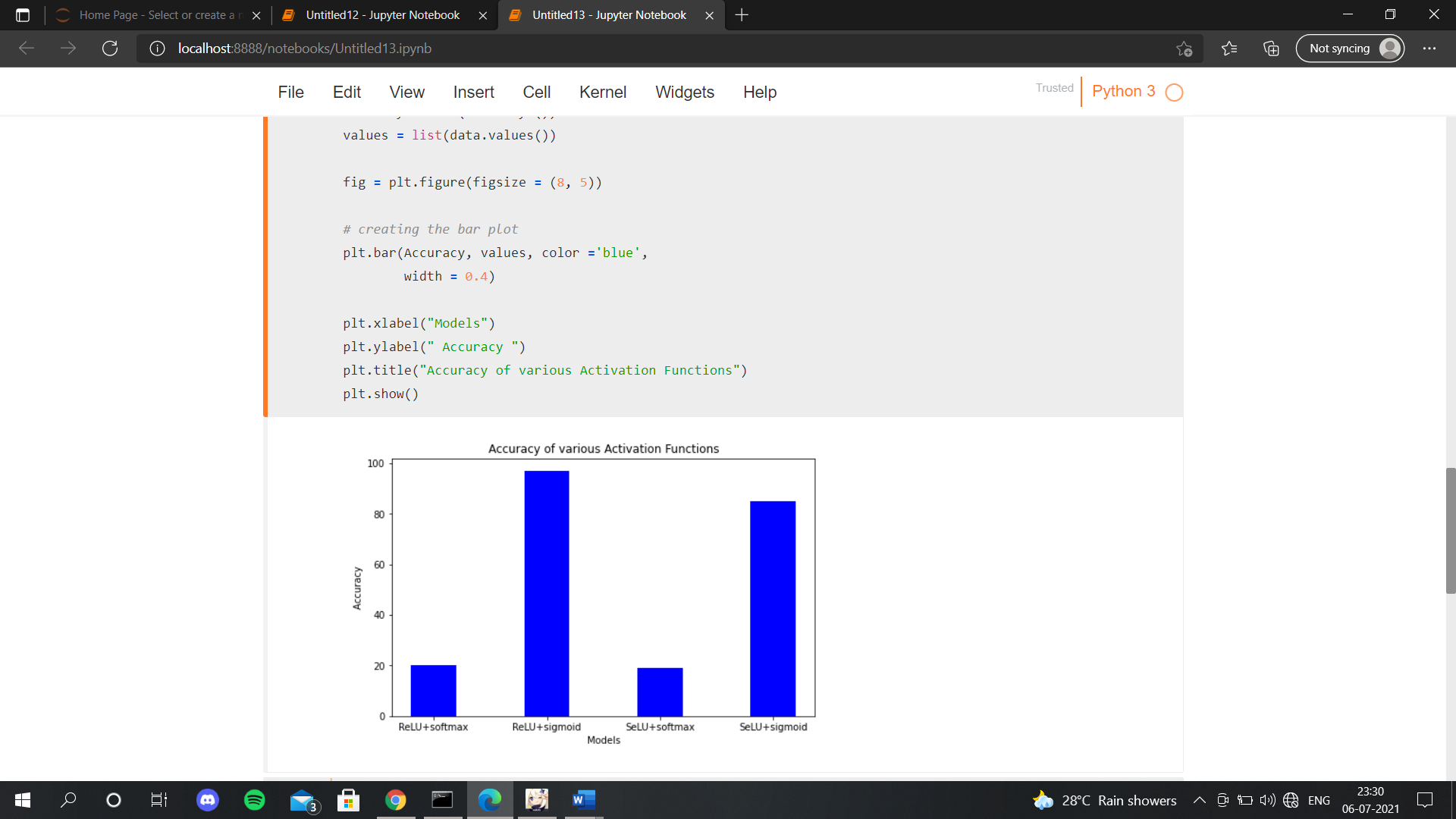


**Figure 3: Precision of Models**

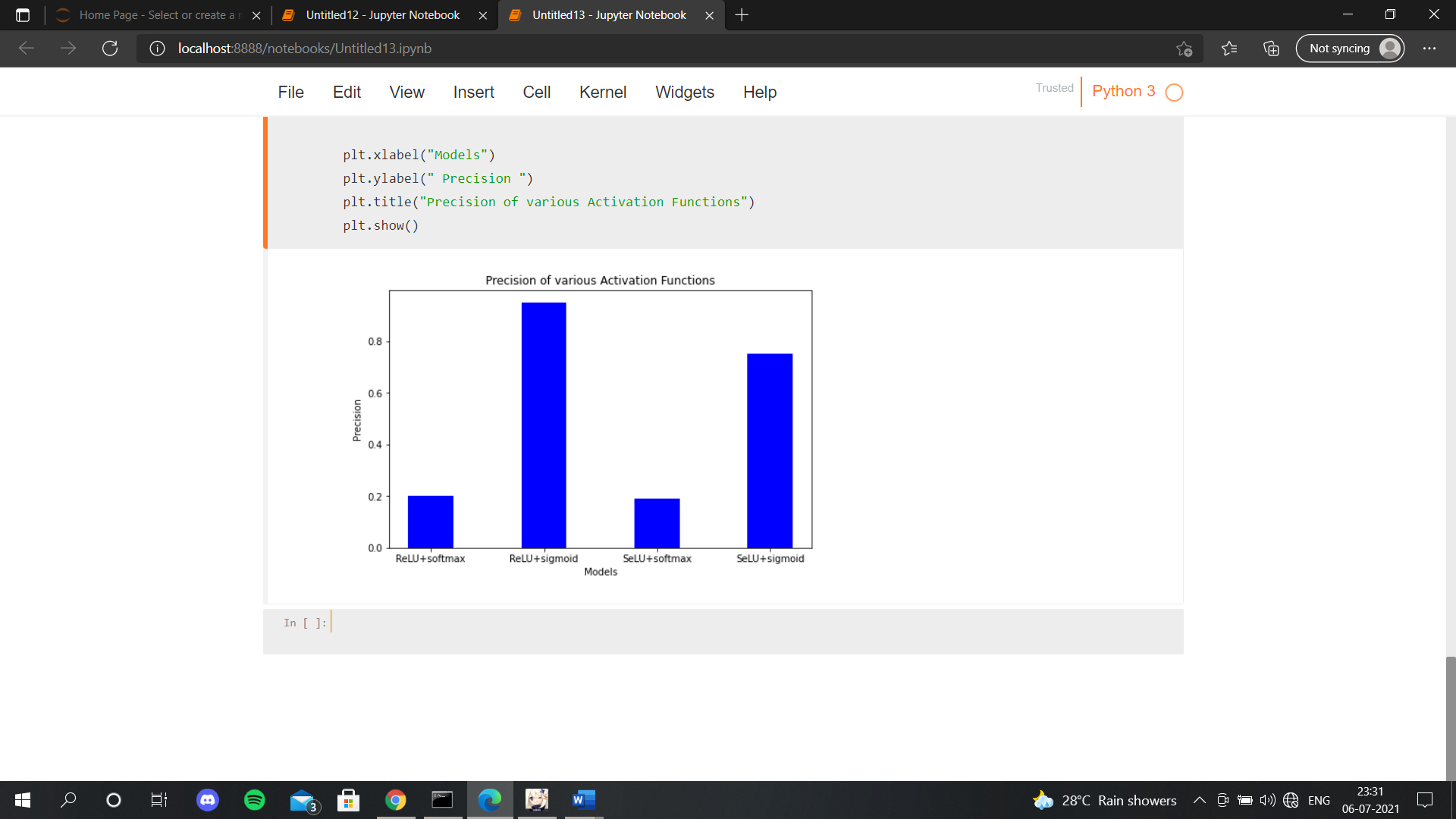
From the graph above we can understand that the hybrid model has much more precision then the other two models. And Content based model has the lowest precision when compared to other models. We have also looked into the accuracy and precision of all the three algorithms with a set of activation function such as relu+sigmoid where relu is used as the hidden layer and sigmoid is used as the output layer similarly we used relu+Softmax, selu+sigmoid, selu+softmax and have selected the one with the highest accuracy and precision. The music advice is a highly complicated subject since it is necessary to structure music so that the favorite songs are recommended to users that will not be defined. Practical tests by real users assessed the offered algorithms and framework satisfactorily.

**Table 3 : Analysis with different activation functions**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Hidden Layer+Output Layer | | | ReLU+Softmax | | ReLU+Sigmoid | | SeLU+Softmax | | SeLU+Sigmoid | |
|  | Accuracy |  | 20.50% |  | 98% |  | 19.00% |  | 85% |  |
|  | Precision |  | 0.2 |  | 0.94 |  | 0.19 |  | 0.82 |  |



**Figure 4: Accuracy of various activation funcations**



**Figure 5: Precision of various activation functions**

We have analysed our model with different activation Such as ReLu, sigmoid, softmax and selu. We choose ReLu as the hidden layer because it carries out complex calculation faster and easy way; selu gives slope larger than one than one for the positive input. These two activation function are taken as an input/hidden layer. Whereas softmax and sigmoid activation function are used as the output layer. Our observation is that ReLU+ sigmoid has higher accuracy and precision as in comparison to other pair of activation function So, we take it into consideration for building our model.

## Conclusion

Recommendation algorithms are an underappreciated component of our daily lives, dictating what we listen to on Spotify and view on YouTube. The precision of these systems is still being improved via research. We discuss such a recommendation engine in our research, which takes both content-based & collaborative filtering into consideration as data to a deep neural network. Similarly, to the methods used by Spotify, Pandora, and other music streaming services, this app analyzes customer musical interests to make music predictions. We aimed to solve frequent issues in existing algorithms in the literature, such as the lack of genuine updates and many iterations, in the creation of our method. Input types that are varied as a conclusion, we have a system that makes high-precision recommendations and seems to be easily expandable. Amazon or Netflix are examples of such services.

## Future Scope

### We will continue to work on the model to improve prediction accuracy. An emotional detector system can be built for other applications that recommends the music by recognizing our face emotion. We wish to grasp better, also, the consequences and the serendipity function of the rating explanation feature. More human behavioral features will be incorporated into the model in the future. For suggestion, we wish to assess the role of these features. As the system that stimulates users' mood becomes more and more relevant user context data available, more improvisational, optimal selection of songs may be produced to recommend.

**References**

1. F.O. Isinkaye a, Y.O. Folajimi, B.A. Ojokoh “Recommendation systems: Principles, methods and evaluation” Egyptian Informatics Journal 2015.
2. Shuai Zhang, Lina Yao, Aixin Sun, Yi Tay “Deep Learning based Recommender System: A Survey and New Perspectives” ACM Computing Surveys 2018.
3. Joe, Mr C. Vijesh, and Jennifer S. Raj “Location-based Orientation Context Dependent Recommender System for Users” Journal of trends in Computer Science and Smart technology (TCSST) 3, no. 01 (2021): 14-23.
4. Zeshan Fayyaz, Mahsa Ebrahimian, Dina Nawara, Ahmed Ibrahim and Rasha Kashef “Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities” Appl. Sci. 2020.
5. AbdulHussien, A. A. (2017). Comparison of machine learning algorithms to classify web pages. International Journal of Advanced Computer Science and Applications (ijacsa), 8(11)
6. Markus Schedl “Deep Learning in Music Recommendation Systems” Frontiers in Applied Mathematics and Statistics 2019.
7. Filippini, D., Alimelli, A., Di Natale, C., Paolesse, R., D'Amico, A., & Lundström, I. (2006). Chemical sensing with familiar devices. Angewandte Chemie International Edition, 45(23), 3800-3803.
8. Osadchiy, T., Poliakov, I., Olivier, P., Rowland, M., & Foster, E. (2019). Recommender system based on pairwise association rules. Expert Systems with Applications, 115, 535-542.
9. A. Joe, Mr C. Vijesh, and Jennifer S. Raj “Location-based Orientation Context Dependent Recommender System for Users” Journal of trends in Computer Science and Smart technology (TCSST) 3, no. 01 (2021): 14-23.
10. C.-W. Chen, C. D. Boom, J. Garcia-Gathright, P. Lamere, J. McInerney, V. Murali, H. Rawlinson, S. Reddy, and R. Yon, “The million playlist dataset,” Spotify - RecSys Challenge 2018, 2018. [Online]. Available: https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge/dataset\_files#