NED University of Engineering & Technology

Department of Computer Science & Information Technology

MUSIC RECOMMENDATION BY ETL, ML & POWER BI ANALYSIS

GROUP MEMBERS: - FASHAD AHMED SIDDIQUE - MUHAMMAD HAMZA

ROLL NUMBER: - CT-19043 - CT-19057

DEPARTMENT: BCIT

COURSE: DWH & MINING

ABSTRACT:

Once the ETL pipeline is set up, the recommendation system can use the transformed and loaded data to generate personalized music recommendations for each user. The motivation for a music recommendation system based on ETL is to provide personalized music recommendations to users based on their listening history, preferences, and activity. We will use an ETL pipeline to extract data from various sources, transform it into a form that can be used by the recommendation system, and load it into a data storage system for fast querying. The recommendation system should be able to handle structured and unstructured data and should be able to perform data transformations and enrichment as needed.

In conclusion, a music recommendation system based on ETL and ML can be a powerful and flexible solution for generating personalized music recommendations for users. It may also be helpful to compare the performance of the recommendation system with a baseline or a control group, such as a group of users who do not receive recommendations or a group of users who receive recommendations from a different system.

INTRODUCTION:

A music recommendation system is a software application that suggests music tracks to users based on their listening history, preferences, and activity. One way to build such a system is to use an ETL (extract, transform, load) pipeline.

The ETL process involves three steps:

- **Extract:** The first step is to extract data from various sources, such as streaming logs, user profiles, and music metadata. These data sources may be structured or unstructured and may be stored in different formats, such as CSV, JSON, or XML.

- **Transform:** The second step is to transform the data into a form that can be used by the recommendation system. This may involve cleaning and normalizing the data, as well as enriching it with additional information, such as music genre, artist popularity, or lyrics.

- **Load:** The third step is to load the transformed data into a data storage system, such as a database or a data lake, which can be queried by the recommendation system. The data

storage system should be optimized for fast querying and scalability, as the recommendation system may need to process large volumes of data in real time.

Once the ETL pipeline is set up, the recommendation system can use the transformed and loaded data to generate personalized music recommendations for each user. The system may use various algorithms and techniques, such as collaborative filtering, content-based filtering, or matrix factorization, to generate the recommendations.

There are several benefits to using an ETL pipeline for building a music recommendation system:

- It enables the recommendation system to ingest data from multiple sources, which can provide a more comprehensive view of the user's music preferences and activity.
- It allows the recommendation system to handle structured and unstructured data, which can be useful if the data sources have different formats or schemas.
- It enables the recommendation system to perform data transformations and enrichment, which can improve the accuracy and relevance of the recommendations.
- It enables the recommendation system to scale and handle large volumes of data, as the ETL process can be optimized for performance and parallelization.

Overall, an ETL-based music recommendation system can provide a powerful and flexible solution for generating personalized music recommendations for users.

MOTIVATION & SCOPE:

The motivation for a music recommendation system based on ETL (extract, transform, load) is to provide personalized music recommendations to users based on their listening history, preferences, and activity. The scope of such a system would be to suggest music tracks to users that they are likely to enjoy, in order to improve their listening experience and keep them engaged with the music streaming service.

There are several potential benefits to using an ETL-based music recommendation system:

- Improved user engagement: By providing personalized recommendations, the recommendation system can help users discover new music that they may not have found otherwise. This can improve their overall listening experience and encourage them to continue using the music streaming service.

- Increased revenue: By keeping users engaged and encouraging them to listen to more music, the recommendation system can help the music streaming service increase its revenue from subscriptions and advertisements.

- Competitive advantage: A high-quality recommendation system can differentiate the music streaming service from its competitors and help it attract and retain users.

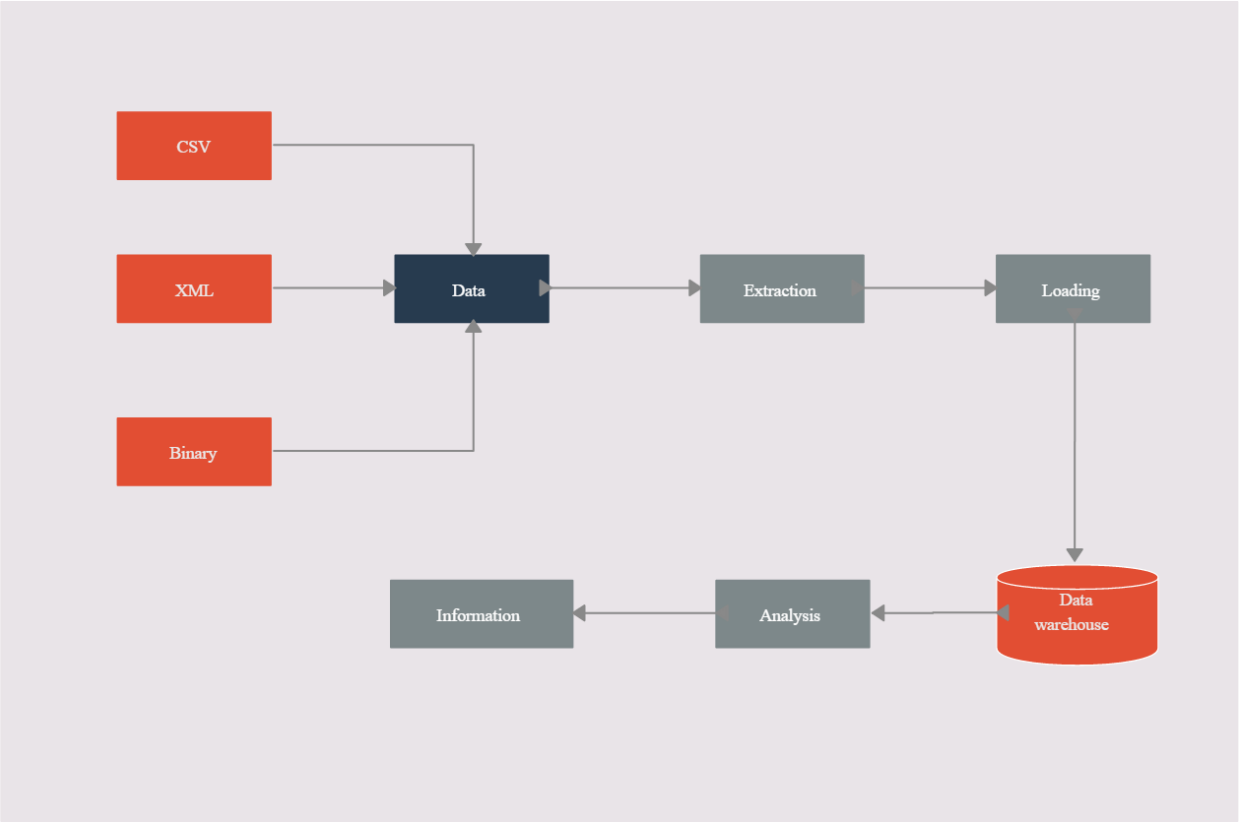
- Improved data quality: By using an ETL pipeline, the recommendation system can ensure that the data it uses is clean, normalized, and enriched, which can improve the accuracy and relevance of the recommendations.

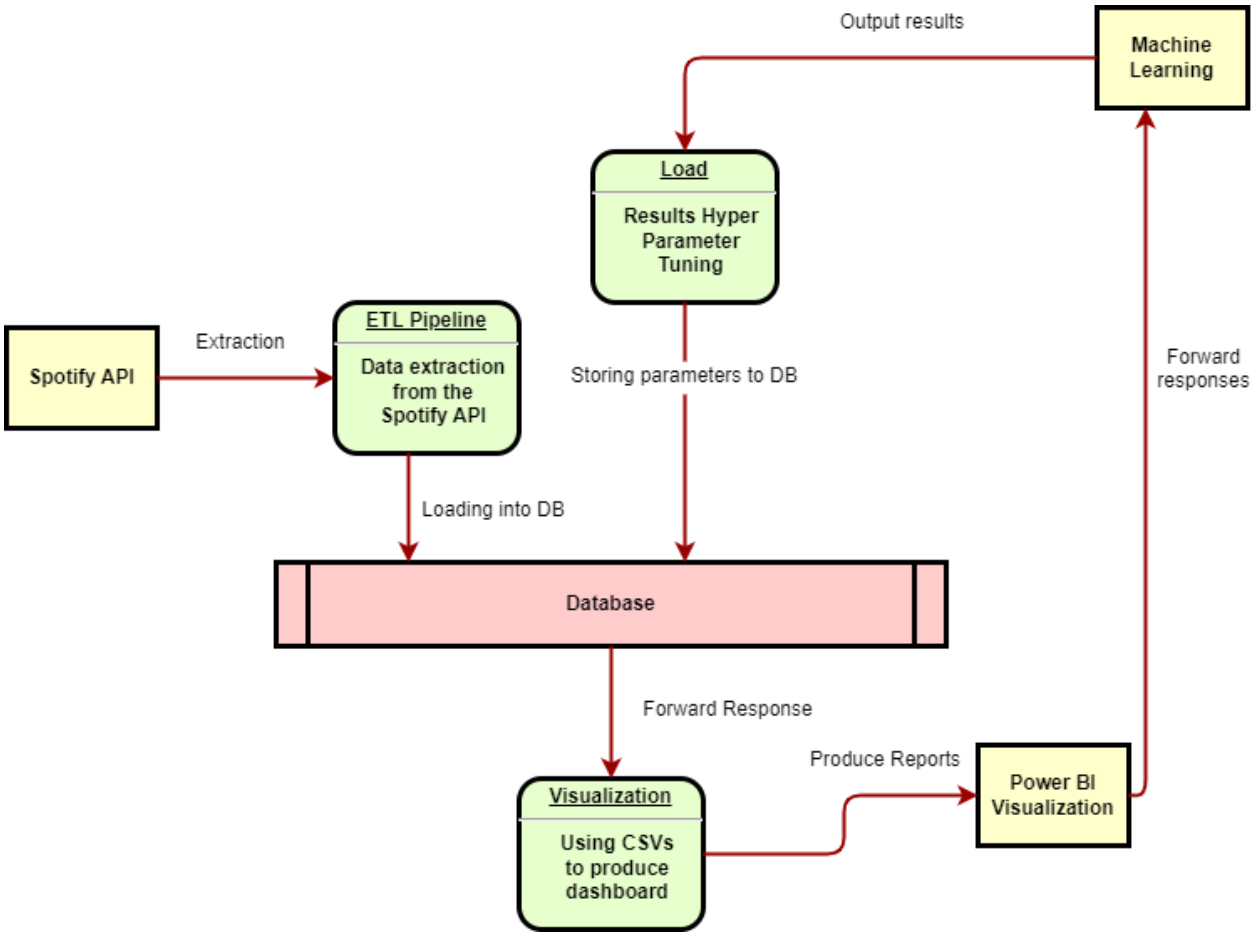
The scope of an ETL-based music recommendation system may vary depending on the specific requirements and goals of the music streaming service. For example, the system may be designed to focus on a particular music genre or to support specific features, such as playlist generation or artist recommendation.

PROBLEM STATEMENT:

Our music streaming service has a large and diverse catalog of music tracks, but many of our users have a hard time finding new music that they will enjoy. We want to build a recommendation system that can suggest music tracks to our users based on their listening history, preferences, and activity. The recommendation system should be able to handle large volumes of data in real time and should be able to generate personalized recommendations for each user. We will use an ETL pipeline to extract data from various sources, transform it into a form that can be used by the recommendation system, and load it into a data storage system for fast querying. The recommendation system should be able to handle structured and unstructured data and should be able to perform data transformations and enrichment as needed. The goal of the recommendation system is to improve user engagement, increase revenue, and differentiate our music streaming service from the competition.

METHODOLOGY DIAGRAM:





SOLUTION/CODING:

GitHub Repository Link:

https://github.com/Fashad-Ahmed/Spotify-ETL-Music-Recommendation

ETL Implementation:

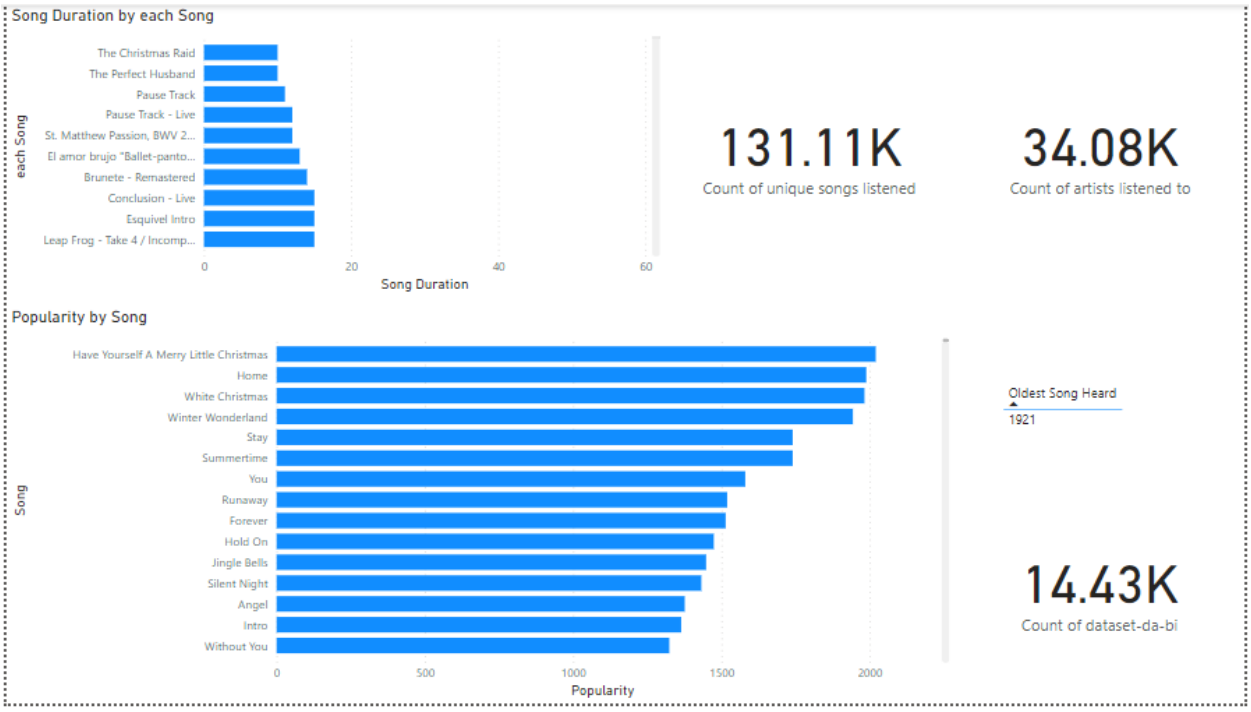
```
headers = {
    "Accept": "application/json",
    "Content-Type": "application/json",
    "Authorization": "Bearer {token}".format(token=TOKEN)
}

# Convert time to Unix timestamp in miliseconds
today = datetime.datetime.now()
yesterday = today - datetime.timedelta(days=90)
yesterday_unix_timestamp = int(yesterday.timestamp()) * 1000

req = requests.get("https://api.spotify.com/v1/me/player/recently-played?after={time}".format(time=yesterday_unix_timestamp), headers = headers)

data = req.json()

song_names = []
artist_names = []
played_at_list = []
timestamps = []

# Extracting only the relevant bits of data from json
for song in data["items"]:
    song_names.append(song["track"]["name"])
    artist_names.append(song["track"]["album"]["artists"][0]["name"])
    played_at_list.append(song["played_at"])
    timestamps.append(song["played_at"][0:10])

song_dict = {
    "song_name": song_names,
    "artist_name": artist_names,
    "played_at": played_at_list,
    "timestamp": timestamps
}
```

Machine Learning Implementation:

```
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline

cluster_pipeline = Pipeline([('scaler', StandardScaler()), ('kmeans', KMeans(n_clusters=10))])
X = genre_data.select_dtypes(np.number)
cluster_pipeline.fit(X)
genre_data['cluster'] = cluster_pipeline.predict(X)
```

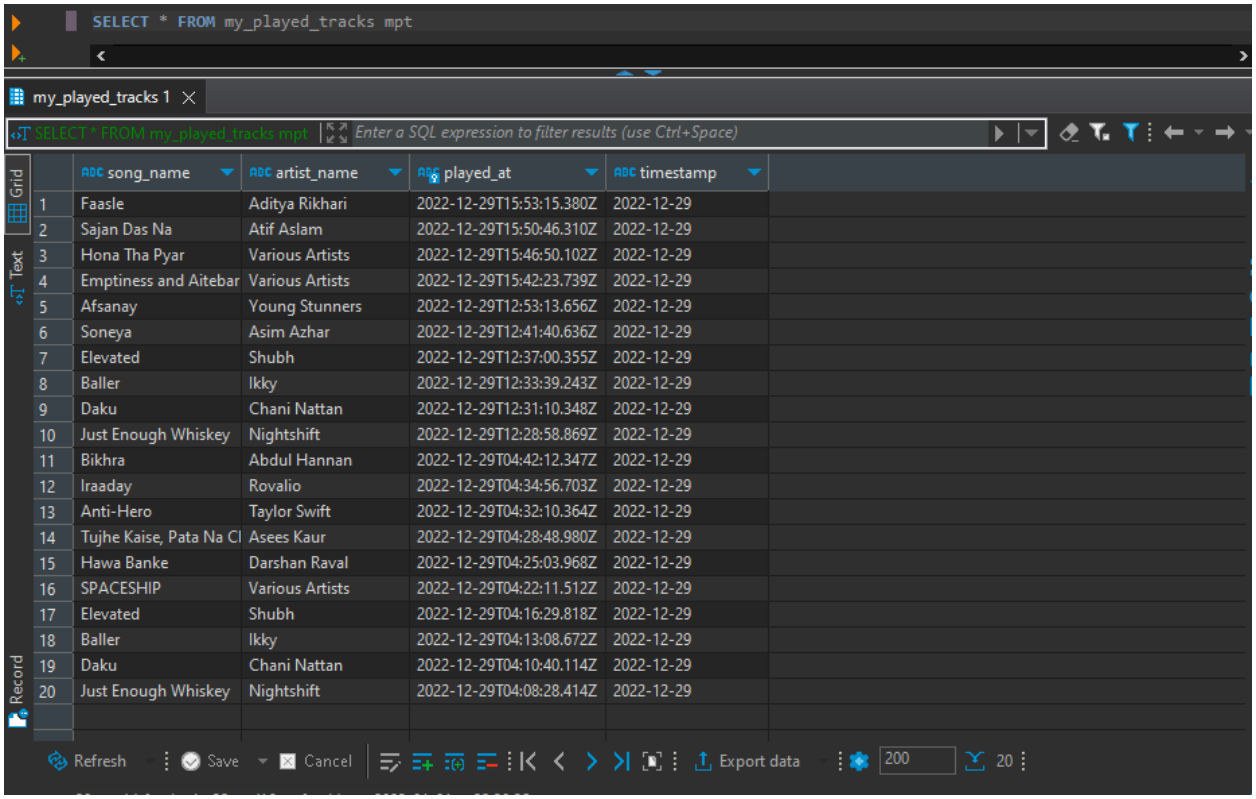
```
# Visualizing the Clusters with t-SNE

from sklearn.manifold import TSNE

tsne_pipeline = Pipeline([('scaler', StandardScaler()), ('tsne', TSNE(n_components=2, verbose=1))])
genre_embedding = tsne_pipeline.fit_transform(X)
projection = pd.DataFrame(columns=['x', 'y'], data=genre_embedding)
projection['genres'] = genre_data['genres']
projection['cluster'] = genre_data['cluster']

fig = px.scatter(
    projection, x='x', y='y', color='cluster', hover_data=['x', 'y', 'genres'])
fig.show()
```

Power BI Dashboard:



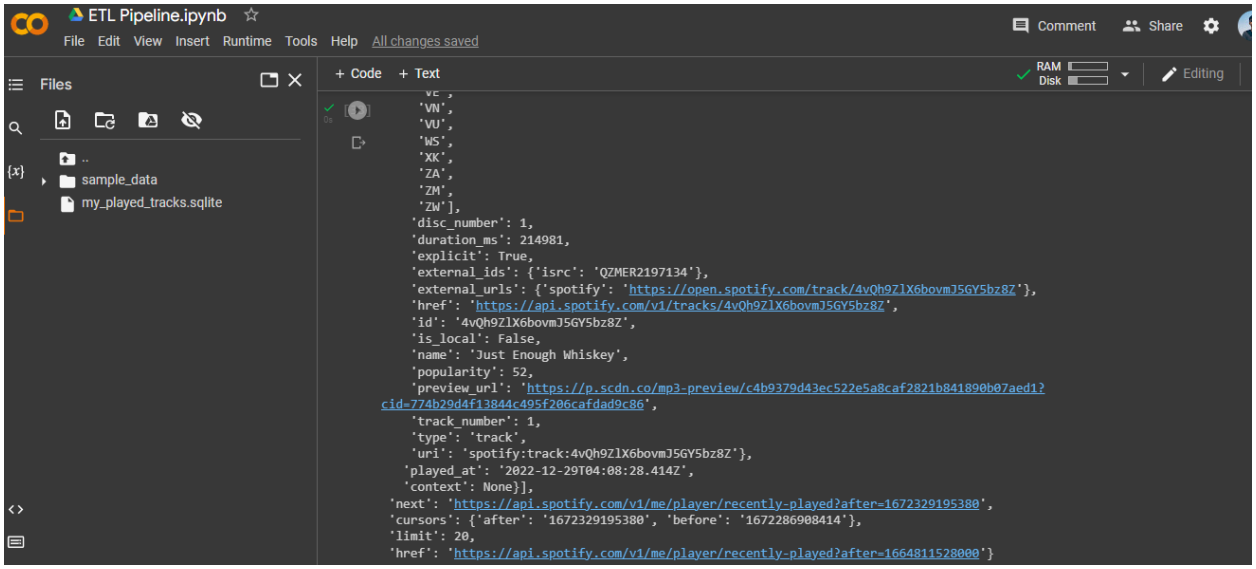
RESULTS & COMPARISON:

ETL Results:



	song_name	artist_name	played_at	timestamp
1	Faasle	Aditya Rikhari	2022-12-29T15:53:15.380Z	2022-12-29
2	Sajan Das Na	Atif Aslam	2022-12-29T15:50:46.310Z	2022-12-29
3	Hona Tha Pyar	Various Artists	2022-12-29T15:46:50.102Z	2022-12-29
4	Emptiness and Aitebar	Various Artists	2022-12-29T15:42:23.739Z	2022-12-29
5	Afsanay	Young Stunners	2022-12-29T12:53:13.656Z	2022-12-29
6	Soneya	Asim Azhar	2022-12-29T12:41:40.636Z	2022-12-29
7	Elevated	Shubh	2022-12-29T12:37:00.355Z	2022-12-29
8	Baller	Ikky	2022-12-29T12:33:39.243Z	2022-12-29
9	Daku	Chani Nattan	2022-12-29T12:31:10.348Z	2022-12-29
10	Just Enough Whiskey	Nightshift	2022-12-29T12:28:58.869Z	2022-12-29
11	Bikhra	Abdul Hannan	2022-12-29T04:42:12.347Z	2022-12-29
12	Iraaday	Rovalio	2022-12-29T04:34:56.703Z	2022-12-29
13	Anti-Hero	Taylor Swift	2022-12-29T04:32:10.364Z	2022-12-29
14	Tujhe Kaise, Pata Na Cl	Asees Kaur	2022-12-29T04:28:48.980Z	2022-12-29
15	Hawa Banke	Darshan Raval	2022-12-29T04:25:03.968Z	2022-12-29
16	SPACESHIP	Various Artists	2022-12-29T04:22:11.512Z	2022-12-29
17	Elevated	Shubh	2022-12-29T04:16:29.818Z	2022-12-29
18	Baller	Ikky	2022-12-29T04:13:08.672Z	2022-12-29
19	Daku	Chani Nattan	2022-12-29T04:10:40.114Z	2022-12-29
20	Just Enough Whiskey	Nightshift	2022-12-29T04:08:28.414Z	2022-12-29

Machine Learning Results:



CONCLUSION:

In conclusion, a music recommendation system based on ETL (extract, transform, load) and ML (machine learning) can be a powerful and flexible solution for generating personalized music recommendations for users. By using an ETL pipeline to extract, transform, and load data from various sources, the recommendation system can handle structured and unstructured data and can perform data transformations and enrichment as needed. The recommendation system can then use ML algorithms to generate personalized recommendations based on the transformed data.

There are several benefits to using an ETL-based recommendation system, including improved user engagement, increased revenue, and competitive advantage. However, there are also limitations to consider, such as data quality and availability, data volume and complexity, algorithmic performance, user behavior, and legal and ethical issues.

To ensure the success of the recommendation system, it is important to carefully design and implement the ETL pipeline and the ML algorithms and to regularly evaluate and optimize the system based on the performance and feedback of the users.

EVALUATION:

There are several ways to evaluate a music recommendation system based on ETL (extract, transform, load). Here are some potential metrics and methods that could be used:

- Accuracy: One way to evaluate the performance of the recommendation system is to measure the accuracy of the recommendations it generates. This can be done by comparing the recommended tracks with the tracks that the user actually listens to, and calculating the percentage of correct recommendations.

- Coverage: Another important metric is coverage, which refers to the percentage of the catalog that is being recommended. A recommendation system with high coverage is able to suggest a wide range of tracks to the users, while a system with low coverage may only recommend a narrow set of tracks.

- Diversity: Diversity refers to the variety of tracks being recommended to the users. A recommendation system with high diversity is able to suggest a wide range of tracks from different genres, artists, and eras, while a system with low diversity may only recommend a narrow set of tracks.

- Novelty: Novelty refers to the degree to which the recommendations are new and unexpected to the user. A recommendation system with high novelty is able to suggest tracks that the user has not heard before, while a system with low novelty may only recommend familiar tracks.

- User satisfaction: Another way to evaluate the recommendation system is to measure the satisfaction of the users who receive the recommendations. This can be done through surveys, focus groups, or online reviews.

To evaluate the recommendation system, it is important to collect data from a representative sample of users and to use statistical methods to analyze the data and draw conclusions. It may also be helpful to compare the performance of the recommendation system with a baseline or a control group, such as a group of users who do not receive recommendations or a group of users who receive recommendations from a different system.

LIMITATIONS:

There are several limitations that can be faced when making a music recommendation system based on ETL (extract, transform, load):

- Data quality and availability: The quality and availability of the data can affect the performance of the recommendation system. If the data is incomplete, inconsistent, or noisy, it can be difficult to generate accurate and relevant recommendations. In addition, if the data sources are not available or are not accessible, it can be challenging to extract and transform the data.

- Data volume and complexity: A music recommendation system may need to process large volumes of data in real-time, which can be challenging if the data is complex or varied. For example, if the data includes multiple types of music metadata, such as lyrics, genre, and artist popularity, it may be difficult to transform and enrich the data in a meaningful way.

- Algorithmic performance: The accuracy and efficiency of the recommendation algorithms can also be a limitation. If the algorithms are not able to handle large volumes of data or are not able to generate accurate recommendations, the performance of the recommendation system may suffer.

- User behavior: The behavior of the users can also be a limitation. For example, if the users do not provide sufficient feedback or do not listen to the recommended tracks, it can be difficult to improve the recommendation system based on their preferences.

Legal and ethical issues: There may be legal and ethical issues to consider when building a recommendation system, such as privacy, copyright, and discrimination. It is important to ensure that the recommendation system complies with relevant laws and regulations and respects the rights and interests of the users.

FUTURE WORK:

There are several areas for potential future work on a music recommendation system based on ETL (extract, transform, load) and ML (machine learning):

- Improved data quality and availability: One area of focus could be on improving the quality and availability of the data used by the recommendation system. This could involve sourcing additional data sources, cleaning and normalizing the data, and enriching it with additional information.

- Enhanced recommendation algorithms: Another area of focus could be on improving the accuracy and efficiency of the recommendation algorithms. This could involve using more advanced ML techniques, such as deep learning, or experimenting with different algorithms and parameter settings.

- Personalization and customization: Another direction for future work could be on providing more personalized and customized recommendations to the users. This could involve using additional data sources or features to better understand the users' preferences and needs, or allowing the users to customize their recommendations through explicit feedback or preferences.

- Integration with other features: A music recommendation system could also be integrated with other features or services, such as playlist generation, artist recommendation, or music discovery. This could provide a more seamless and comprehensive experience for the users.

- Evaluation and optimization: It is important to regularly evaluate and optimize the recommendation system to ensure that it meets the needs and expectations of the users and the business. This could involve collecting data from the users, analyzing the data, and making adjustments to the ETL pipeline and the recommendation algorithms as needed.

Overall, there are many opportunities for future work on a music recommendation system, which could help to improve the accuracy, relevance, and personalization of the recommendations, and to enhance the user experience and the business value of the system.