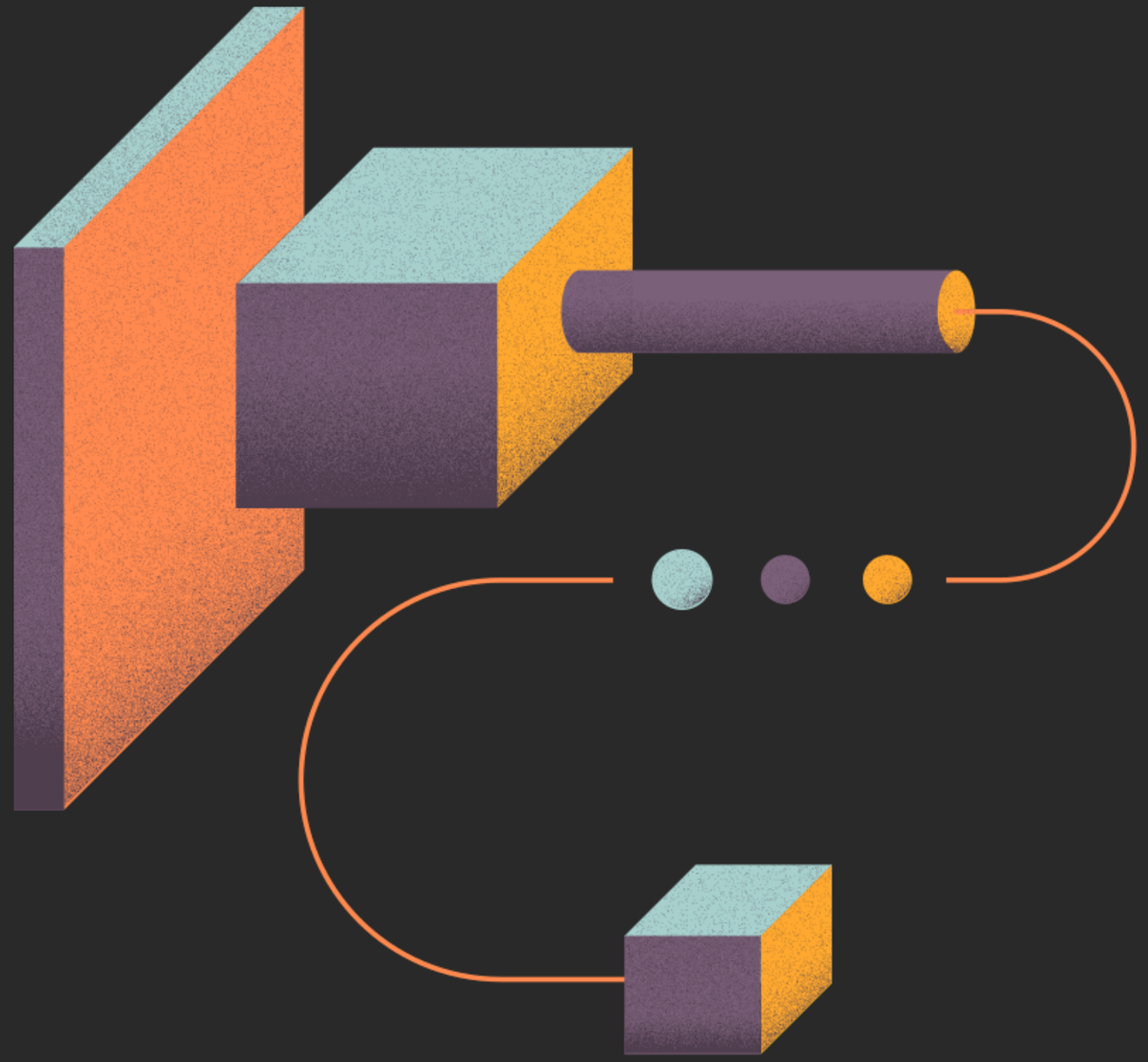
MUSIC RECOMMENDATION BY ETL, ML AND POWER BI ANALYSIS

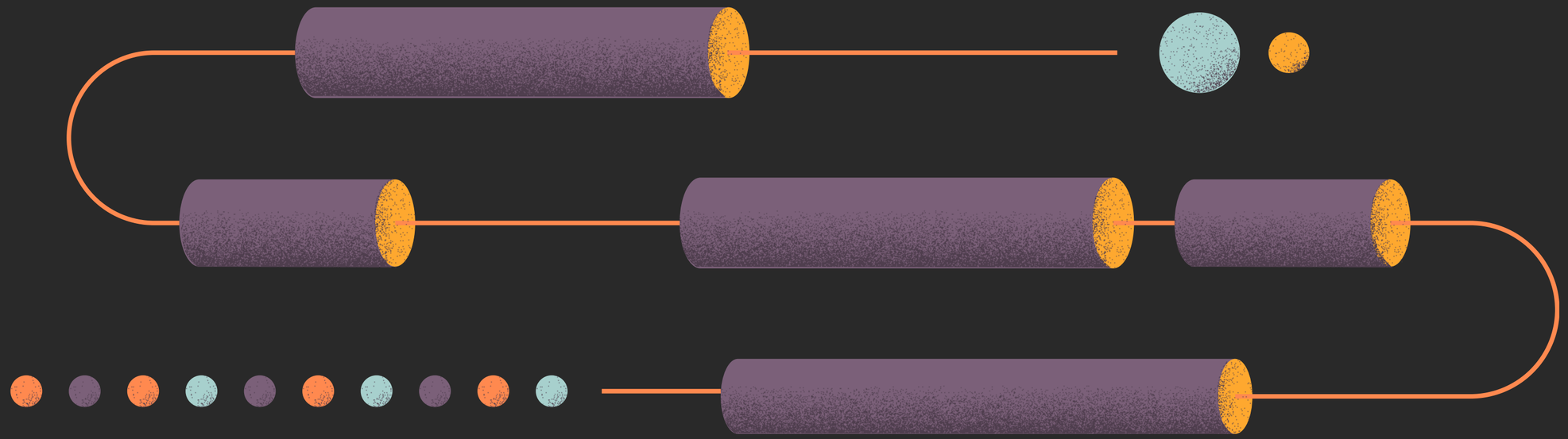


WELCOME EVERYONE

Welcome Message

Presenting to you a recommendation system using ETL, ML and Power BI analytics and visualization

GET STARTED



LIST OF CONTENT

ALL ABOUT THE TEAM

Muhammad Hamza
CT-19057

Fashad Ahmed Siddique
CT-19043

Why Are We Here ?

To present our Music Recommendation System using ETL (extract, transform, load), Power BI visualization and machine learning algorithms.

PROBLEM STATEMENT

Our Vision

Improve user engagement, increase revenue, and differentiate our music streaming service from the competition by providing a competent music recommendation system.

A problem well stated is a problem half solved.

John Dewey

SYSTEM FLOW

ETL PIPELINE with SPOTIFY API

Built an ETL pipeline that extract data directly from the spotify user.

Data Visualization with POWER BI

Using the loaded data from the pipeline to present useful analytics to the audience.

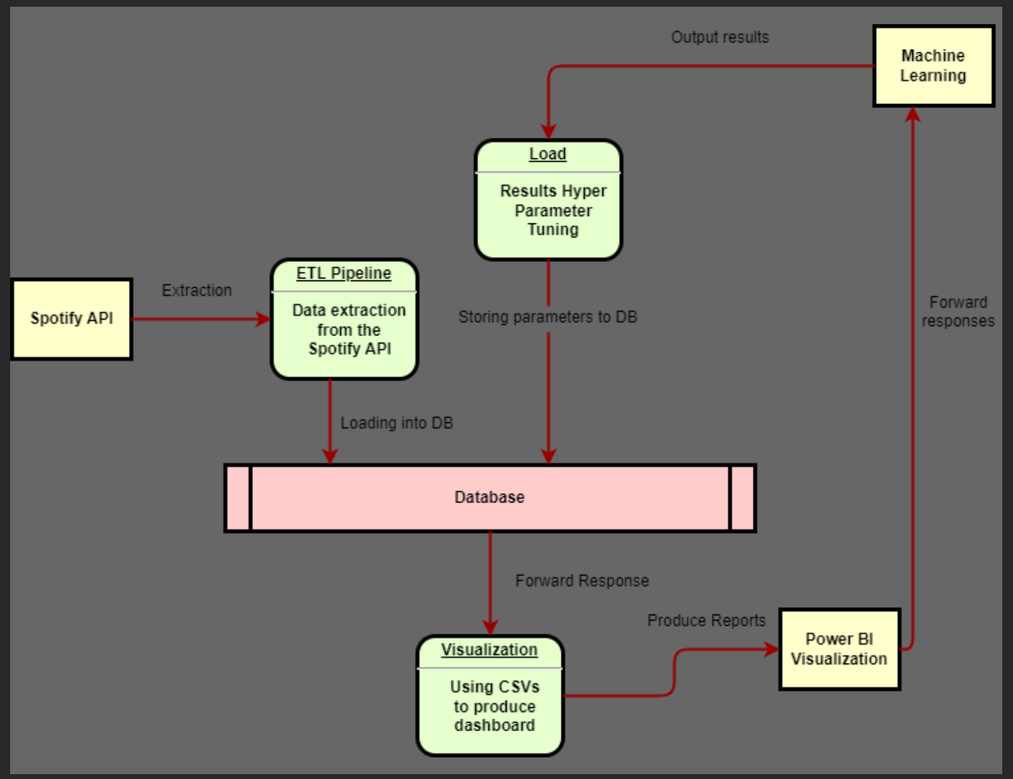
Recommendations using various ML algos

Using the pipeline data to infer useful features and then process them separately to produce computations to suggest similar song choices with respect to genres they listen to.

METHODOLOGY DIAGRAM

Step One

Using ETL pipleline to extract all the data from the source and pulling other data from secondary repositories



Step Two

Loading the data into the dataabase

Step Four

Using the loaded data in the machine learning algorithms to compute the features and produce valuable results as a recommendation for the user

Step Thre

Using the loaded data in a visualization tool to infer useful analytics and producing useful. insights

SOLUTION & CODE

ETL Code

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

# Convert time to Unix timestamp in milliseconds
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
}
```

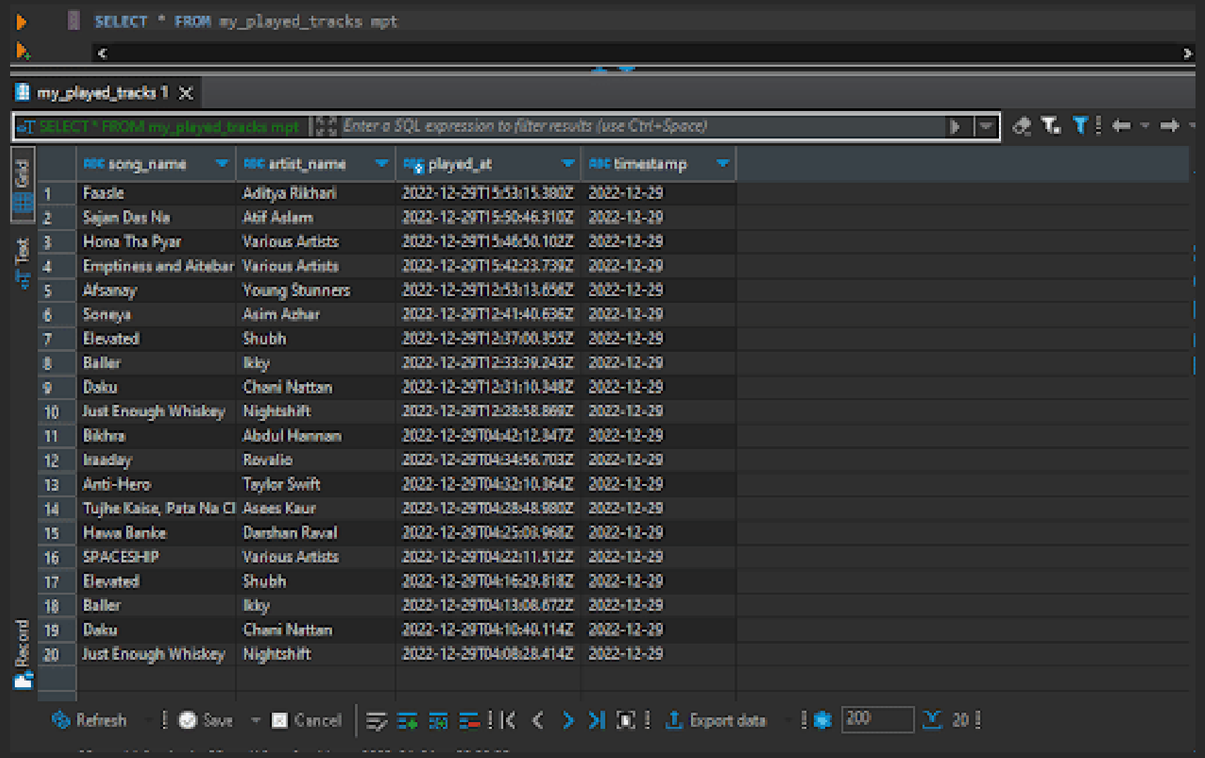
ML code

```
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)
```

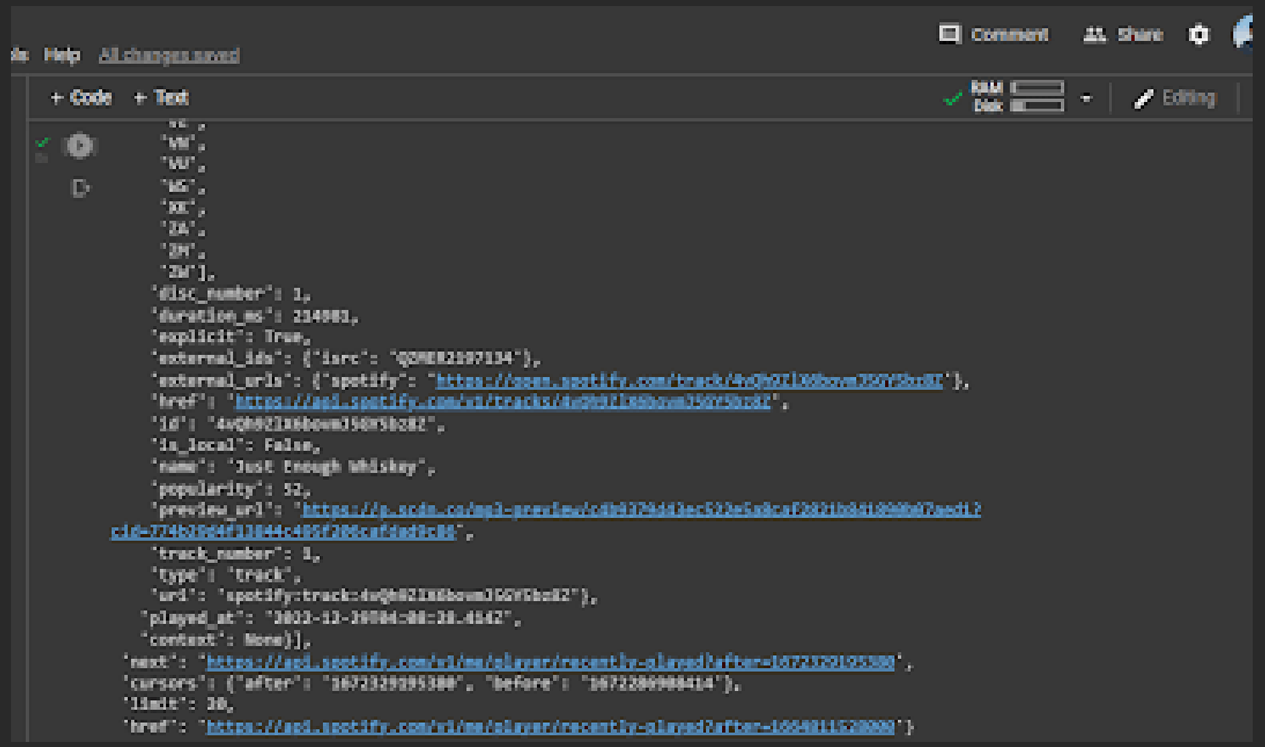
```
# Visualizing the Clusters with t-SNE

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()
```

SOLUTION & CODE

Database view



Colab view

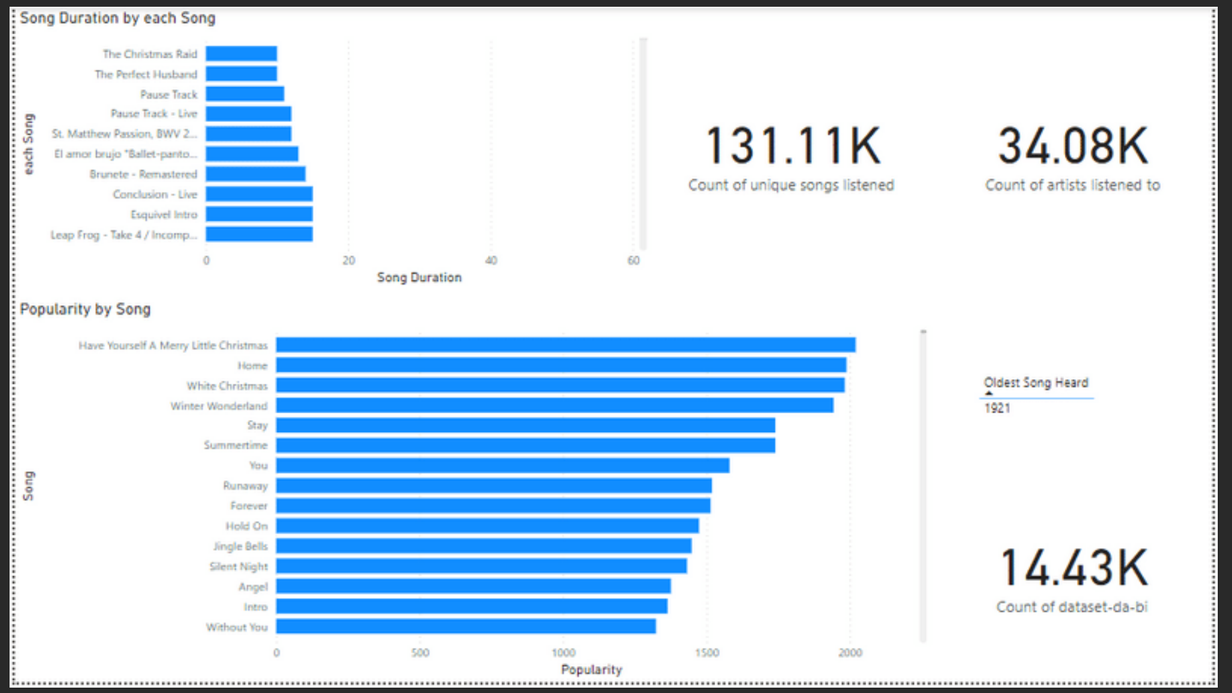


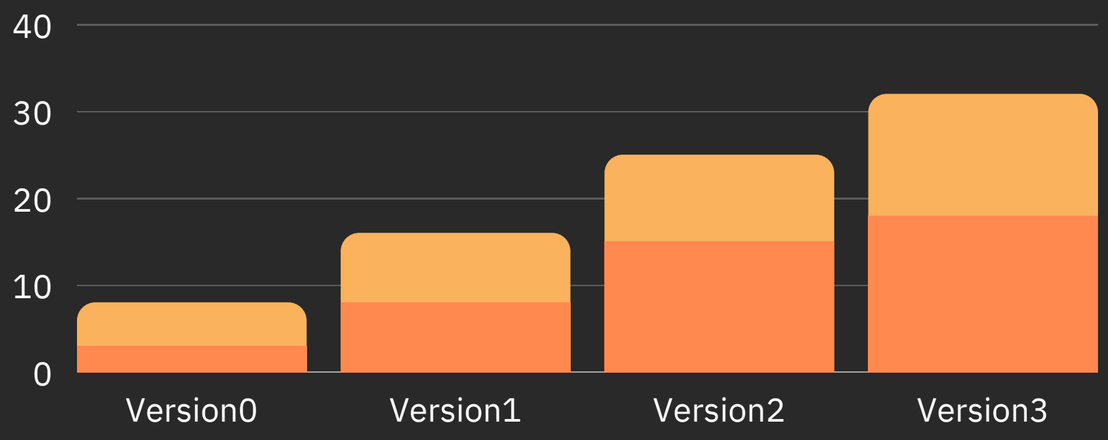
SOLUTION & CODE

Raw API view


```
{
  "items":[
    {
      "track":{
        "album":{
          "album_type":"single",
          "artists":[
            {
              "external_urls":{
                "spotify":"https://open.spotify.com/artist/3ozYqVCLohfpXIhalkhM8D"
              },
              "href":"https://api.spotify.com/v1/artists/3ozYqVCLohfpXIhalkhM8D",
              "id":"3ozYqVCLohfpXIhalkhM8D",
              "name":"Aditya Rikhari",
              "type":"artist",
              "uri":"spotify:artist:3ozYqVCLohfpXIhalkhM8D"
            }
          ],
          "available_markets":[ ],
          "external_urls":{
            "spotify":"https://open.spotify.com/album/2xfAmgKBQroCYbDnl18me3"
```


Power BI Dashboard view





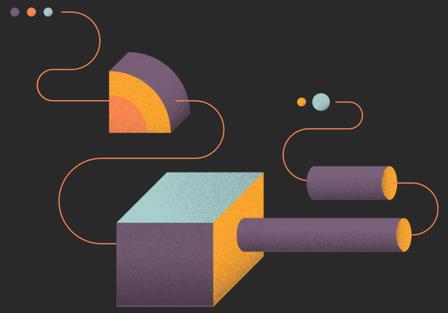
FUTURE WORK

Producing Better Versions

Coming up with better versions with the best methodlogies to keep the system upto date and produce better results.

Enhancing Accuracy

As the usage increases, more and more data will be aded into the datasets ultimately helping increase the accuracy of the system.



THANKS FOR YOUR TIME