

PROJECT DELIVERABLE 3

Results

As stated in my previous deliverables, since music taste is inherently subjective, I would be evaluating the model primarily based on my own musical tastes and knowledge along with the opinions of my friends on the model's output. For the previous deliverable, I did some preliminary tests by inputting a song and judging the results, as seen below.

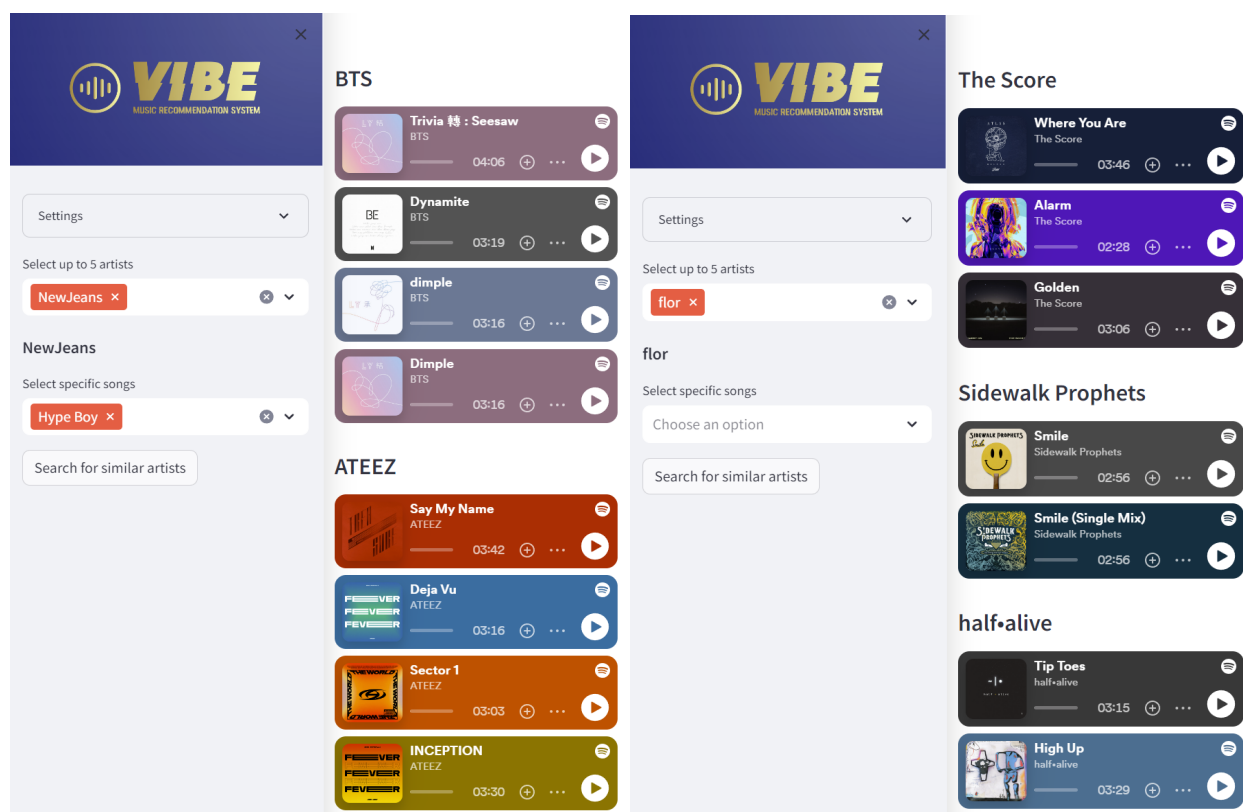
```
... ['Hype Boy', 'NewJeans']  
    ['GOT THE THRILLS', 'TWICE']  
    ['Talk that Talk', 'TWICE']  
    ['Love War (Feat. BE'0)", 'YENA']  
    ['SET ME FREE (ENG)', 'TWICE']  
    ['Silent Cry', 'Stray Kids']  
    ['LILAC', 'IU']  
    ['Kill This Love', 'BLACKPINK']  
    ['AYA', 'MAMAMOO']  
    ['Nxde', '(G)I-DLE']  
    ['SET ME FREE', 'TWICE']  
    ['_WORLD', 'SEVENTEEN']  
    ['Yummy Yummy Love', 'MOMOLAND']  
    ['dimple', 'BTS']  
    ['PLAY (feat. Changmo)', 'CHUNG HA']  
    ['Dimple', 'BTS']  
    ['Walk With You', 'NCT DREAM']  
    ['Not Shy', 'ITZY']  
    ['Ring The Alarm', 'KARD']  
    ['Oh My!', 'SEVENTEEN']  
    ['INCEPTION', 'ATEEZ']  
    ['HOT', 'SEVENTEEN']  
    ['Secret Story of the Swan', 'IZ*ONE']
```

```
... flor  
    The Wrecks  
    Sea Girls  
    ROMES  
    Bad Suns  
    Circa Waves  
    The Faim  
    Fontaines D.C.  
    Friday Pilots Club  
    The Night Game  
    Castlecomer  
    Citizen Way  
    Arrested Youth  
    DREAMERS  
    Declan McKenna  
    The Struts  
    Frank Carter & The Rattlesnakes  
    THE DRIVER ERA  
    Smallpools  
    Liily  
    Unspoken  
    Pale Waves  
    Saint Motel
```

I have since added additional options and steps to the algorithm. The system can now take both artists and songs as inputs at the same time, which allows for a greater degree of recommendation customizability. I also added a variance setting that ranges from 1 to 5, which increases the run-to-run variance of recommended songs. By optimizing the dataframe and saving it as a parquet file instead of a csv, I was able to use almost the entire dataset of over a million songs.

I am still using the same general algorithm as the previous deliverable, calculating the Euclidean distance between songs, but with a few additional steps. After the list of 100 similar songs is generated, each song is given a score between 1 and 100 based on their similarity ranking. The system then outputs a list of artists and songs from the list, ordered by the cumulative score of their top 4 similar songs. The default value of 1 always gives the same output for a given input, while higher values give more diverse recommendations.

Two example outputs can be seen below for different inputs.



I am quite pleased with the performance of the algorithm. I have tested the system with a variety of artists from different genres, and have found the recommendations to be in line with expectations. Since the core algorithm was largely the same as the previous deliverable, results were very similar to my preliminary results.

Demonstration

I have implemented the system into a Streamlit web app (seen above) that I am hosting both at vibe-music.streamlit.app and <https://huggingface.co/spaces/AI3x-T/Vibe>. I used Streamlit to create the app, as it was an easy and seamless way to integrate my algorithm into a website that looked good and was user-friendly. Since Streamlit Community Cloud only offers a guarantee of 1GB of RAM to each app, I decided to also host it on Hugging Face, which offers 16GB of RAM in its free tier.

The app allows the user to input a list of artists and optionally songs, along with It then generates a list of recommendations and creates a Spotify embed for each recommended song, allowing users to listen to the songs directly on the page without having to go to another website. I plan to continue optimizing the app and potentially add some more settings to let users tweak the system more to their liking. I also aim to add integration with a database so that I can save user input. This could allow me to train a model in the future and implement collaborative filtering alongside the current content-based recommendation algorithm.