Spotifinder

Song Recommendations

Finding & Recommending

- Finding song entry - given part or whole of song and/or singer name
  - Best match for ‘Fast Chapman’ - ['Fast Car', 'Tracy Chapman', 79]
- Recommending songs - given song entry
  - Recommendations based on choice of ['Fast Car', 'Tracy Chapman', 79]
      - ['Fast Car', 'Tracy Chapman', 79]
      - ['Pope Is a Rockstar', 'SALES', 69]
      - ['Cheer Up, My Brother', 'HNNY', 64]
      - ['Aphrodite', 'RINI', 62]
      - ['Love You Down - Extended Remix', 'Ready For The World,Louil Silas Jr.', 48]

Finding Song Entry

- Based on song title and artist
- Components
    - TfIDF Vectorizer trained on song title and artist in dataset
    - AutoEncoder - trained on full DTM(12433 features/columns) generated by TfIDF Vectorizer for dataset
    - NearestNeighbor trained on encoded DTM generated by passing full DTM through the Encoder portion of the trained AutoEncoder

Recommending Songs

- Based on song genres plus numerical properties(‘popularity', 'duration_ms', 'explicit', ‘danceability', 'energy', 'key', 'loudness', 'mode', 'speechiness', 'acousticness', 'instrumentalness', 'liveness', 'valence', 'tempo', ‘time_signature’)

- Components

  - TfIDF Vectorizer trained on genres column in dataset

  - StandardScaler trained on numerical features in dataset

  - AutoEncoder - trained on full DTM generated by TfIDF Vectorizer(10887) + scaled numerical features(15) from dataset - 10902 features/columns in total.

  - NearestNeighbor trained on encoded feature set generated by passing full DTM + scaled numerical features through the Encoder portion of the trained AutoEncoder

Notebooks and Code

- All notebooks and code located in https://github.com/Spotify-TT-204/Spotifinder
- Notebooks - located in *…/notebooks*
    - *data_cleanup.ipynb* - work associated with data analysis, cleanup and engineering.
    - *data_modeling.ipynb* - work associated with finding song entry
    - *recommendation_modeling.ipynb* - work associated with song recommendation

- Python class wrapper around find song entry and song recommendation functionality - *…/app/data_model/find_songs.py*
- Sample code using class wrapper - *…/app/data_model_usage.py*
    - Output from sample code - *…/app/data_model_usage.out*