SoundSensei

Customizable Playlist Curation and Safe Listening for All

Team Data Detectives: Arya Mohan, Jake Michiels, Koushika Kesavan, Maitreyee Talnikar, Rajvi Parekh

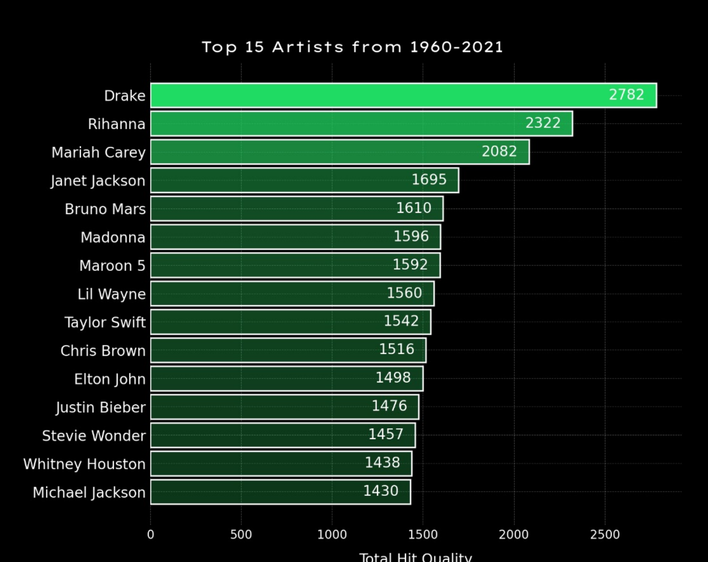
PROBLEM STATEMENT

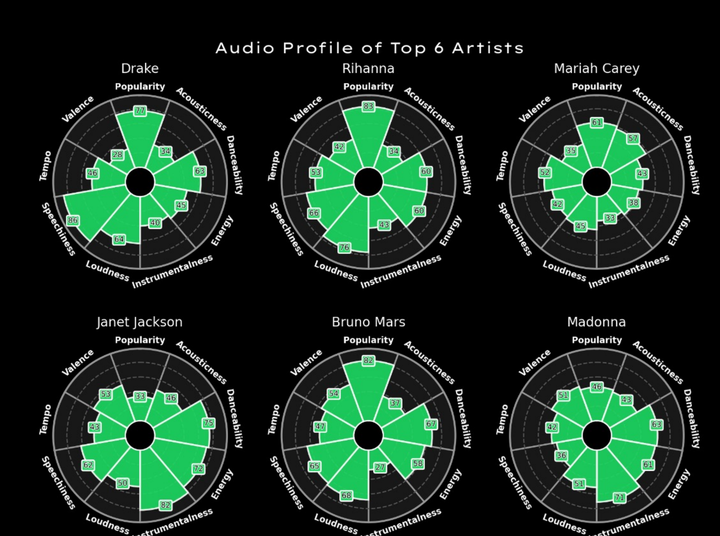
SoundSensei is a sophisticated music recommendation system centered on transparency, customization, and control. Tailored for music enthusiasts and artists, our system not only analyzes your playlist for insights but also provides customizable parameters for personalized recommendations. Users can refine playlists through intuitive controls, ensuring a user-centric musical journey. Safety is paramount, empowering parents to curate kid-friendly content. Our target audience includes avid music listeners and those seeking a better understanding of content tones. Successful implementation promises a substantial impact on comprehensive music analysis and customizable recommendations, gauged through professional user studies and feedback.

METHODOLOGY

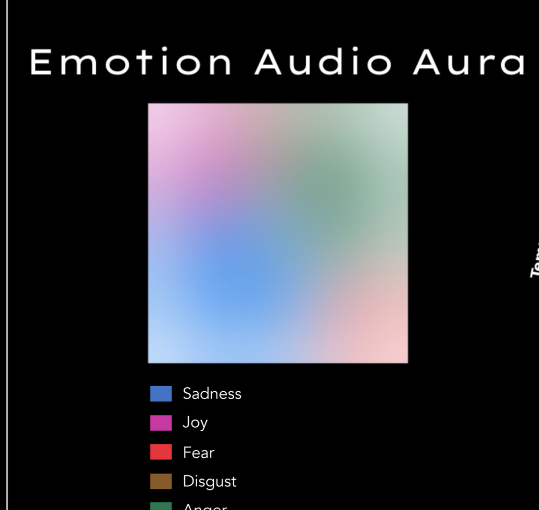
Our project leverages an integrated approach, focusing on automatic playlist continuation and playlist summarization using data from songs and their lyrics, bringing forth unique innovations.

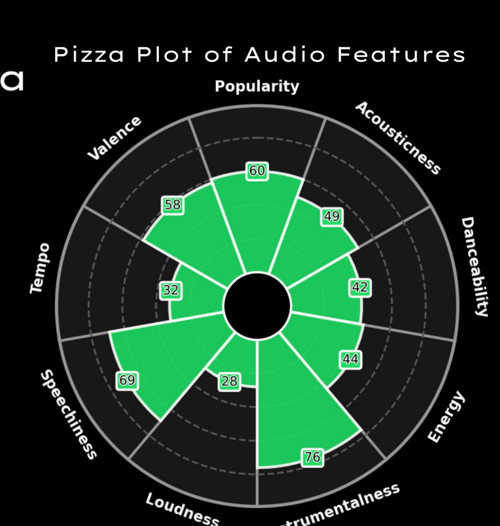
- **Data Collection and Exploratory Data Analysis (EDA):** We employ the Spotify API for acquiring song data and the Genius API for lyrics data. We performed extensive EDA to gain a deep understanding of the dataset, which is crucial for effective modeling.

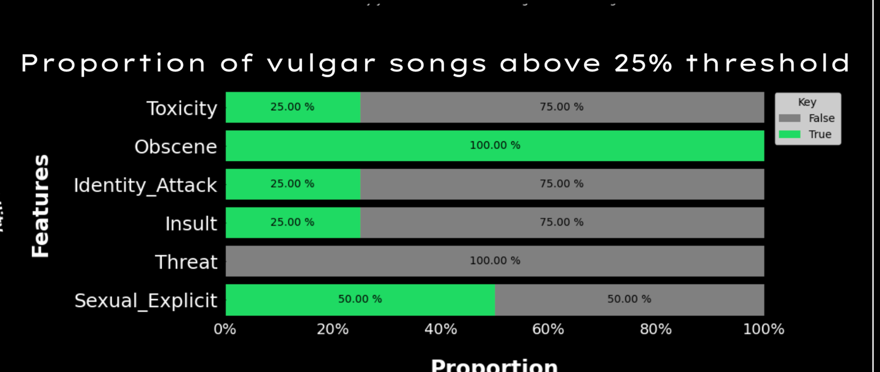




- **Playlist Summarization:** Our system performs a thorough EDA of user playlists, focusing on audio features and sentiment analysis via IBM Cloud's Natural Language Understanding. We also incorporate vulgarity analysis using the Perspective API. These steps help users understand their musical tastes. The summarization provides insights into the playlists' overall mood and themes, thereby enhancing user engagement.

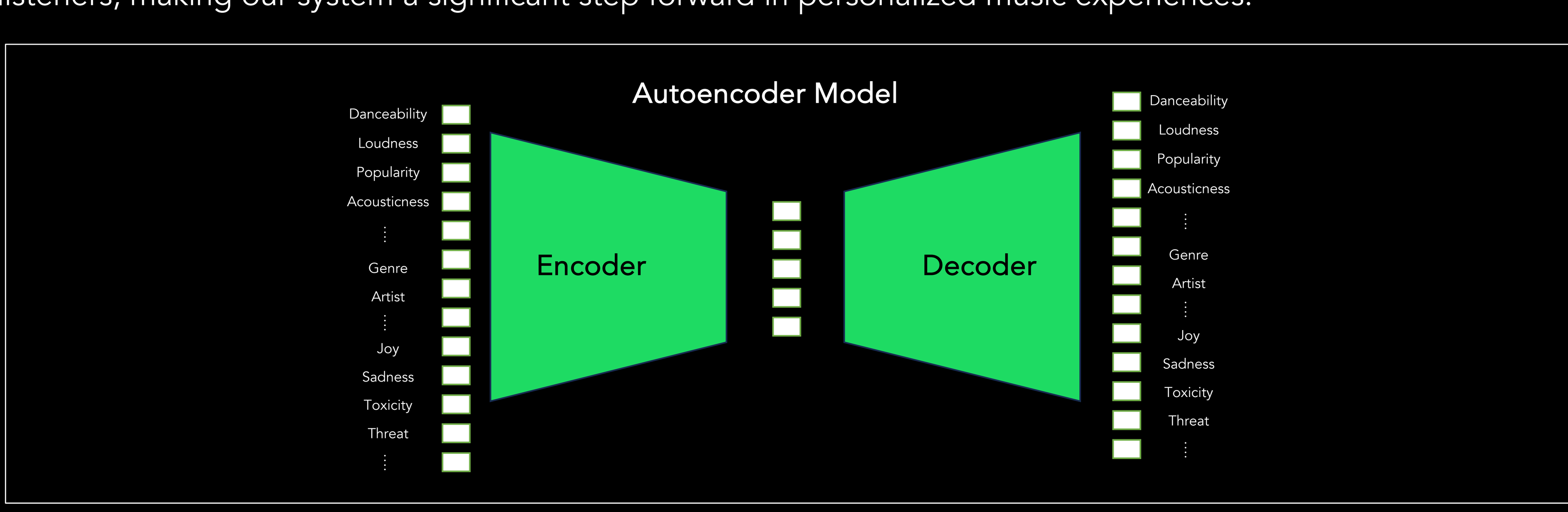


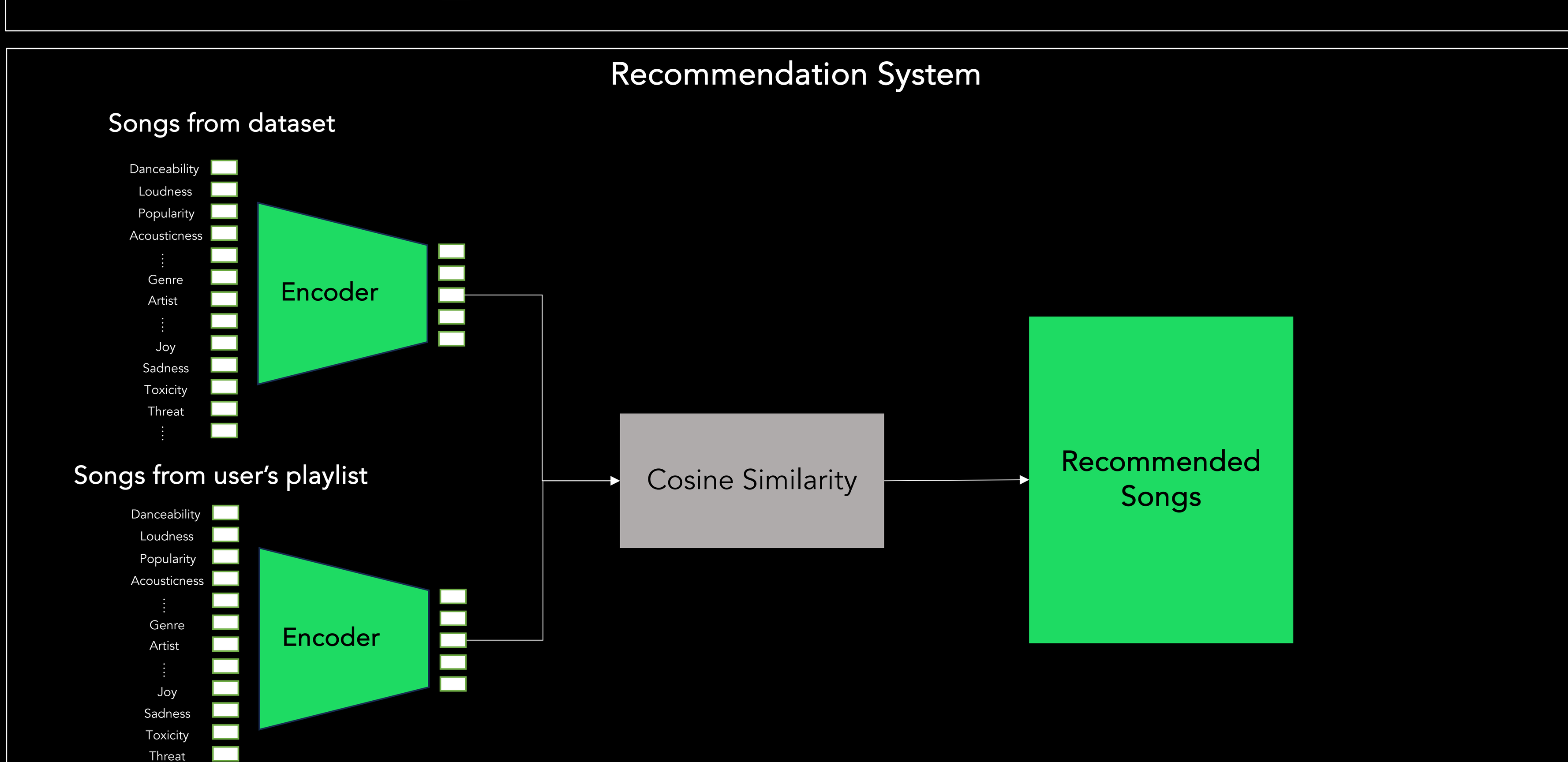




- **Automatic Playlist Continuation:** Our automatic playlist continuation model uses an autoencoder architecture in conjunction with cosine similarity to generate song recommendations that complement the user's existing playlist. The user can control different audio features they want in their recommendations providing insight into why they were recommended particular songs. Another notable addition is the 'kid-friendly' mode, which filters explicit lyrics, making the playlists appropriate for younger listeners.

Our project stands out for its user-centric design and enhances traditional music recommendation systems by innovatively integrating sentiment analysis, vulgarity detection, and an advanced autoencoder-based model for playlist continuation. Additionally, our system equips users to control the recommendation process through various input factors, a feature often lacking in contemporary recommender systems. The system's configurability and the inclusion of a kid-friendly mode cater to a diverse audience, ensuring safety and an enriching experience for all users, including children. This approach meets the evolving demands of modern listeners, making our system a significant step forward in personalized music experiences.





DATA

SoundSensei uses several Python packages to collect and process the data, including BeautifulSoup, Spotipy, GeniusLyrics and spaCy. Key steps in the data collection pipeline include:

1. Web scraping the Wikipedia page for the Billboard Year-End Top 100 charts for each year to collect the song name, artist name, and chart position. We collected data from 1960 to 2022.

2. Using the Spotipy library to access the Spotify API and collect audio features for each song, like danceability, energy, acousticness, duration, explicit content, etc.

3. Accessing the Genius Lyrics API via the lyrics-genius package to collect lyrics for each song and passing it to Perceptive API to identify different types of explicit content such as obscenity, toxicity, or violence.

The final preprocessed dataset contained close to 6000 rows, with columns for audio features, artists and genres, release year, duration, explicit content, etc.

EXPERIMENTS

The experimental phase involves performing detailed hyperparameter tuning. This yields several key observations:

Autoencoder Architecture: We experimented with varying the number of hidden layers and neurons in the autoencoder. It was observed that increasing the depth of the model improved feature extraction but also led to a risk of overfitting. To mitigate this, we implemented dropout layers, which helped in regularizing the model.

Learning Rate Optimization: Accurate learning rate is crucial for the autoencoder. A gradual decrease in the learning rate, implemented via learning rate schedulers, provided more stability and a consistent decrease in the validation loss.

Distance Metric Selection: We experimented with various distance metrics such as Euclidean, Manhattan, and Cosine similarity, we ultimately chose Cosine similarity for its effectiveness in high-dimensional spaces. It had a more nuanced understanding of song similarity by comparing the orientation rather than the magnitude of vectors in the embedding space.

Distance Metric	Precision	Recall
Euclidean Distance	0.88	0.93
Manhattan Distance	0.73	0.65
Cosine Distance	0.91	0.87

Profanity Filter Sensitivity: The profanity filter's sensitivity was fine-tuned to strike a balance between filtering explicit content and maintaining a diverse range of song recommendations.

RESULTS

The testbed consists of three primary models:

Sentiment Analysis Model: Utilizing the IBM Cloud Natural Language Understanding API, our sentiment analysis model achieved an accuracy of approximately 92% on a test set. This high level of accuracy, indicates the model's effectiveness in accurately classifying the emotional content of lyrics, which is crucial for the recommendation system's efficacy.

Vulgarity Model: This model plays a crucial role in identifying songs containing explicit content, flagging them for potential exclusion in our recommendations. To evaluate this model, a small set of songs from a variety of genres containing explicit or clean lyrics were labeled for each category. We found that the model is very effective at detecting obscene language. It consistently returns scores close to 1 when obscene language is present, and scores close to 0 when not present. Using a threshold of 50%, the model correctly identified obscene songs 98% of the time. The model struggles to identify the other categories with high confidence and returns roughly .5 on explicit examples and roughly 0 on clean examples. This is likely due to the figurative nature of song lyrics. Fortunately, setting a low threshold (~25%) for these categories allows the model to classify songs correctly 89% of the time. Future work in this area should be focused on improving the confidence of highly figurative examples, and expanding the types of explicit content the model is able to detect (i.e. drug references, self-harm, gang violence, etc).

Automatic Playlist Continuation Model: The autoencoder exhibited a remarkably low mean squared error (MSE) of ~0.0052 in its loss function on the validation set. This low MSE indicates a high degree of accuracy in feature reconstruction, indicating that the embedding space can be reliably used for our recommendation system. We evaluated our final recommendation system using precision and recall metrics. It showed a balanced performance in providing relevant (high precision of ~91%) and diverse (high recall of ~87%) song suggestions. This balance enhances the user experience by aligning closely with user preferences and introducing new musical elements.

CONCLUSION & FUTURE WORK

In summary, SoundSensei provides customizable and safe recommendations. Its core strengths lie in its ability to provide transparent and personalized music recommendations, coupled with robust safety features for family-friendly content. The utilization of advanced techniques, such as sentiment analysis and vulgarity detection, distinctly positions SoundSensei in the realm of music curation and recommendation systems.

Looking forward, SoundSensei could evolve to encompass a broader range of music genres and languages, making it more inclusive and globally applicable. Incorporating user feedback loops and machine learning models that adapt to individual user preferences over time could further enhance the personalization aspect. Furthermore, the exploration of more advanced natural language processing techniques to further enhance sentiment and vulgarity analysis could also provide more nuanced and accurate recommendations.