Comparing self-extracted to third-party audio features for music genre classification

Bradley Erickson BErickson15@winona.edu Winona State University

Check out the project on GitHub.

Background

Song genres are primarily relative to the listener and there is no clear-cut way to classify which genre a song belongs to. With the power of machine learning, researchers have taken a crack at automating this process using artificial neural networks. Conducting this and other audio analysis can prove useful to music companies that wish to understand what customers enjoy the most. Two mirror models will be created. One trained on the features extracted directly from the audio and the other trained using third-party song metrics. The two models will be compared to see which method has the better prediction accuracy.

Data

Our experiment will analyze 10 **genres:**

1. Blues
2. Classical
3. Country
4. Disco
5. Hip-hop
6. Jazz
7. Metal
8. Pop
9. Reggae
10. Rock

Self-extracted feature data is gathered using the GTZAN dataset. This dataset contains 1,000 30 second song clips spanning across the 10 genres. Features:

1. Zero crossing rate
2. Chroma shift
3. Spectral Centroid
4. Spectral bandwidth
5. Spectral roll-off
6. Tempo
7. Room mean square
8. Mel-Frequency coef.

Third-party data is scraped from the Spotify API. From the top 100 songs for each of the 10 genres, we metrics Spotify engineers created. Metrics:

1. Key
2. Mode
3. Time signature
4. Acousticness
5. Danceability
6. Energy
7. Instrumentalness
8. Liveness
9. Loudness
10. Valence
11. Tempo

Methods

We need to transform the inputs to represent a normal distribution and scale the inputs to a (0 – 1) range. We split the data into training and test sets using an 80-20 split, stratifying on genre. To compare how the datasets performance against one another, we train a dense convolution neural network. Using Monte-Carlo cross-validation, we train 20 models and average the validation accuracy. The final model we produced has an input, 64-node, batch normalization 32-node, 16-node, and output layer. We will choose to train with 20 epochs and a 64-unit batch size. These parameters yielded the best performance without overfitting the training data.

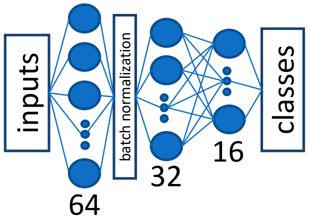

Figure 1: Model of neural network

Results

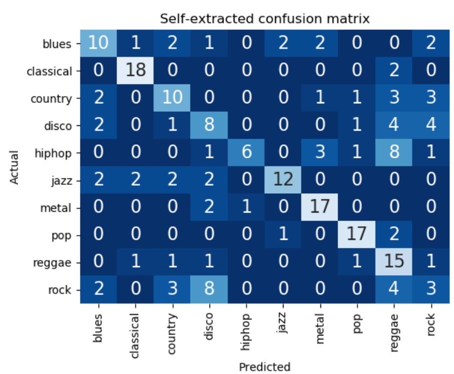

Figure 2: Self-extracted model confusion matrix

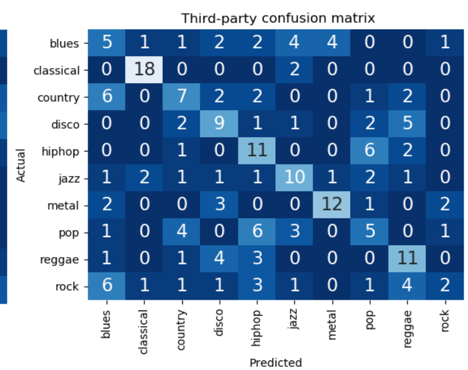

Figure 3: Third-party model confusion matrix

	Avg. Validation Accuracy	Test Accuracy
Self-extracted	52.97%	58%
Third-party	46.65%	45%

Figure 4: Final accuracy percentages

Figures 2 and 3 show the training data's actual versus predicted results. The diagonals show our true positives. We see that both datasets do an excellent job predicting classical and a poor job predicting rock. Additionally, the self-extracted data model does a good job predicting metal, pop, and reggae. The non-diagonals can reflect some similarities between genres. For example, in the self-extracted data, rock music is often classified as disco. On the third-party data, we see hip-hop and pop being classified as one another. These genres most likely share similar properties. Figure 4 shows percentage results between the models. The self-extract model performs better than the third-party model To expand on this research, I would do the following:

1. Look at other classification methods
2. Expand amount of data
3. Use the same songs in each dataset

References

1. Jeong, Il-Young, and Kyogu Lee. "Learning Temporal Features Using a Deep Neural Network and its Application to Music Genre Classification." Ismir. 2016.
2. Li, Tom LH, Antoni B. Chan, and Andy HW Chun. "Automatic musical pattern feature extraction using convolutional neural network." Genre 10 (2010): 1x1.
3. Tzanetakis, George, and Perry Cook. "Musical genre classification of audio signals." IEEE Transactions on speech and audio processing 10.5 (2002): 293-302.