Predicting Next Music Genre for Twitter Users

Parya Abadeh
University of Guelph
pabadeh@uoguelph.ca

Abstract—This report explores the development of a novel deep learning model to predict the next music genre a Twitter user might be interested in. The model leverages a unique dataset encompassing users' historical music listening habits gleaned from links shared on Twitter to music platforms, along with their corresponding tweet timelines and associated stress levels. This rich dataset allows for the investigation of the relationship between music preferences and factors like emotions expressed in tweets. I propose an LSTM (Long Short-Term Memory) and Gated Recurrent Unit (GRU) based Seq2Seq model to capture the sequential nature of user behavior. The model considers sequences of past music genres, tweet information such as time of the day of posting, stress levels, and temporal details to predict the next genre of interest. Attention mechanisms are incorporated to focus on crucial elements within the input data, enhancing prediction accuracy. The evaluation phase utilizes metrics like accuracy, alongside visualizations to comprehensively assess the model's capabilities and limitations. This research aims to bridge the gap in existing literature by analyzing the sequential patterns of music genres alongside potential influences from user emotions expressed on social media. The findings hold promise for advancements in personalized music recommendation systems and contribute to a deeper understanding of the correlation between music preferences and user behavior.

Keywords—Music Recommendation, Genre Prediction, LSTM, Attention

I. Introduction

Music is everywhere. Whether you're walking down a crowded street or chilling in a coffee shop, chances are you'll hear someone jamming out with headphones on. It's a constant companion, and its effects go way beyond just being a fun way to pass the time. Music therapy, for example, uses the power of music to help people feel better mentally and physically.

With the rise of the internet, the way we listen to music has totally changed. Streaming services like Spotify and Apple Music offer millions of songs at your fingertips. No more rummaging through dusty CD collections – now you have access to pretty much any genre or artist you can imagine. This explosion of music choices has led researchers to ask some interesting questions about how people pick the tunes they listen to. Existing recommendation systems [1] are pretty good at figuring out what kind of music you like based on what you've listened to before within a particular app. They basically learn your taste from your past listening habits.

But what if there's more to the story? What if the things you post on social media, like your tweets, could also give clues about the kind of music you're into? This project takes a fresh approach by looking not just at your past listening habits, but also at your tweets and even your stress levels associated with your tweets, to try and predict what kind of music you might be interested in listening to next. It's kind of like trying to understand the connection between your mood and the music you choose. By exploring this new area, this project could not only improve music recommendation systems, but also help us understand the fascinating link between music and our everyday lives.

During rest of this research report, I will talk about the research methodology and results.

II. Related works

Music recommendation systems (MRS) have become a cornerstone of music streaming platforms like Spotify. These systems personalize user experiences by suggesting music tailored to their individual tastes. Researchers have delved into various methodologies and algorithms employed by these platforms to understand how they work.

While the inner workings of these commercial systems remain largely a secret, scholarly investigations and data-driven analyses are shedding light on their possible functionalities. For instance, Gulmatico et al. (2022) [5] explored the potential hierarchical structure of the algorithms employed by Spotify, offering insights into the intricate processes behind its recommendation engine. Similarly, Kaminski and Ricci (2009) [6] investigated the relationship between music attributes and user preferences, laying the groundwork for predicting song success based on audio factors. Music recommendation systems are crucial for navigating the vast amount of music available on digital platforms. Collaborative filtering, a popular approach, analyzes user listening history to recommend similar songs enjoyed by users with similar tastes [12]. However, this method suffers from the "cold start" problem, where it struggles to recommend new or unpopular songs due to a lack of usage data [13]. This paper [12] proposes a latent factor model that addresses this limitation. It leverages music audio to predict latent factors that influence user preference, even when usage data is unavailable. The paper compares two methods for predicting these factors: a traditional bag-of-words approach and deep convolutional neural networks. Their findings demonstrate the effectiveness of deep learning in music recommendation, paving the way for more accurate recommendations, especially for new and unpopular music.

Traditionally, research in music recommendation has primarily focused on machine learning approaches hosted on platforms like Kaggle. These endeavors have spanned supervised and semi-supervised learning algorithms, including logistic regression, naive Bayes, k-nearest neighbors, and support vector machines [7]. However, the emergence of deep learning has opened a new chapter in MRS. Deep learning's ability to capture complex patterns within vast datasets makes it a powerful tool for music

recommendation. Deep learning models can not only learn intricate relationships between music attributes and user preferences but can also handle hierarchical representation learning, leading to more accurate and personalized recommendations.

One recent study by Elbin [8] introduced a novel approach called MusicRecNet, which combines signal processing techniques with a convolutional neural network (CNN) model. MusicRecNet aims to surpass previous classifiers in music genre classification and recommendation tasks, addressing limitations outlined in earlier works [9]. This model goes beyond genre classification, encompassing both classification and recommendation functionalities. It can also detect music plagiarism, catering to the needs of both users and the music industry. Evaluated on the GTZAN dataset, MusicRecNet demonstrated promising results in music genre classification, similarity assessment, and recommendation tasks.

Another exciting area of research focuses on personalized music recommendation based on emotions. Music has a profound impact on our emotional state, and leveraging this connection can significantly enhance user experiences. A study by [9] explored integrating deep learning models, such as Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), and hybrid architectures (CNN-LSTM and LSTM-CNN), for emotion detection in music recommendation systems. The study compared these models to identify the most effective algorithm for recommending songs based on a user's emotional state. Furthermore, the application incorporated a CNN model for emotion detection through facial expressions, offering a comprehensive approach to personalized music recommendation. By fetching songs and playlists from third-party APIs and detecting emotions through text analysis or facial expressions, the application recommends music tailored to the user's current mood. The study demonstrated the effectiveness of the LSTM-CNN model in emotion detection for music recommendation applications.

In the domain of music information retrieval (MIR), music genre classification remains a significant challenge. A study by [10] compared two distinct methodologies for genre classification. The first approach employed deep learning techniques, specifically CNNs trained on spectrograms (visual representations of audio signals). The second approach utilized hand-crafted features extracted from audio data and fed into traditional machine learning classifiers. This comparative analysis aimed to identify the most influential features contributing to genre classification tasks. Experiments conducted on the Audio Set dataset yielded promising results, with an ensemble classifier combining the CNN and traditional machine learning approaches achieving the best performance.

These studies showcase the ongoing advancements in music recommendation systems and genre classification. Deep learning is emerging as a powerful tool for capturing complex relationships within music data, leading to more accurate and personalized music recommendations. Future research directions include incorporating user emotions and exploring techniques to address noisy data for improved performance.

While user listening, history offers valuable insights for music recommendation systems, it represents just one facet of user behavior. Social media platforms provide a rich tapestry of user activity that can further enhance recommendation accuracy. By incorporating social media data into the recommendation process, we can gain a more comprehensive understanding of user preferences and interests. For instance, a user who frequently shares posts about a particular genre or artist on social media likely harbors a strong interest in that genre. Leveraging such social media cues, alongside listening history, can enable the recommendation system to provide more relevant and personalized music suggestions. Future research directions in music recommendation could explore techniques for effectively integrating social media data with traditional listening history data. This combined approach holds the potential to significantly improve the quality of music recommendations, offering users a more tailored and engaging listening experience.

III. Methodology

In this section, I aim to discuss the methodologies that have been implemented, shedding light on their underlying mechanisms, and providing a comprehensive understanding of their application.

A. Long Short Term Memory (LSTM)

Unveiling user preferences in the ever-expanding world of music streaming is a challenge that deep learning techniques like Long Short-Term Memory (LSTM) networks are uniquely suited to tackle. Unlike traditional neural networks that process information one data point at a time, LSTMs hold a distinct advantage: their feedback loop architecture. This allows them to analyze entire sequences of data, such as a user's music listening history, and identify long-term patterns within that data. This exceptional ability at handling sequential information makes LSTMs ideal for tasks like sequence prediction, a capability I leverage in this study.

Here, I utilize the power of LSTMs to analyze user listening patterns and predict the next music genre a user might enjoy. Crucially, LSTMs excel at overcoming the "vanishing gradient" problem, a common hurdle in traditional neural networks that can hinder their ability to learn from long sequences. This makes LSTMs particularly well-suited for analyzing user listening history, where understanding long-term preferences is key to accurate predictions. In the next section, we'll delve deeper into the specific architecture of LSTM models and how it empowers them to unlock valuable insights from sequential data.

B. LSTM Architecture

Traditional recurrent neural networks (RNNs) struggle with capturing long-term dependencies in sequential data due to the vanishing gradient problem. This limitation hinders their ability to learn from long sequences. Long Short-Term Memory (LSTM) networks address this challenge. LSTMs are a special type of RNN architecture equipped with internal mechanisms that effectively deal with the vanishing gradient problem. By understanding how LSTMs are structured, we can gain insight into how they overcome this limitation and excel at tasks involving sequential data, such as music

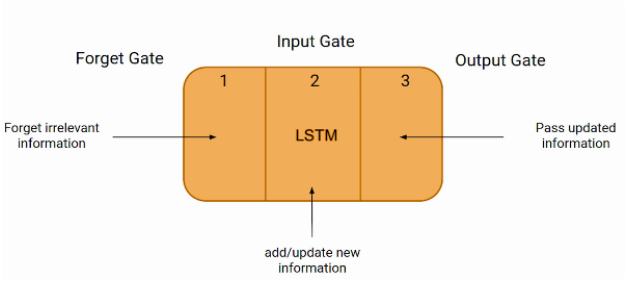
recommendation. The LSTM network architecture consists of three parts, as shown in Figure 1, and each part performs an individual function.

Instead of passively processing information in a sequence, LSTMs excel at managing long-term dependencies through a series of "gates." These gates act like valves, controlling the flow of information within the LSTM cell.

- Forget Gate: This gate decides what information from the previous time step is no longer relevant and can be discarded. It essentially cleans up the cell's memory, focusing on the most crucial details.
- Input Gate: This gate determines what new information from the current input should be remembered. It acts like a filter, selecting valuable data to be added to the cell's internal state.
- Output Gate: Finally, the output gate controls what information from the current cell state is passed on to the next time step in the sequence. It essentially determines what the LSTM "remembers" to use for future predictions.

By working together, these gates enable LSTMs to effectively learn from long sequences and overcome the vanishing gradient problem that hinders traditional RNNs. Similarly to a layer of neurons in a traditional neural network, an LSTM unit combines these gates with a memory cell, creating a powerful tool for processing sequential data.

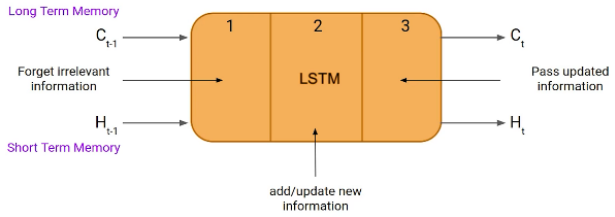
Fig. 1. LSTM network architecture



Just like a simple RNN, an LSTM also has a hidden state where H(t-1) represents the hidden state of the previous timestamp and Ht is the hidden state of the current timestamp. In addition to that, LSTM also has a cell state represented by C(t-1) and C(t) for the previous and current timestamps, respectively.

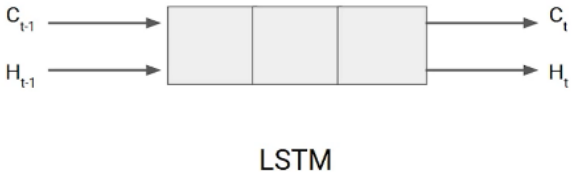
Here the hidden state is known as Short term memory, and the cell state is known as Long term memory. Refer to the following image.

Fig. 2. LSTM network architecture (hidden states)



It is interesting to note that the cell state carries the information along with all the timestamps.

Fig. 3. LSTM network architecture (overview)



C. GRU Architecture

GRUs, or Gated Recurrent Units, are a type of recurrent neural network (RNN) architecture like Long Short-Term Memory (LSTMs). Both are designed to tackle sequential data, where information unfolds over time. However, GRUs offer a simpler and more computationally efficient alternative to LSTMs.

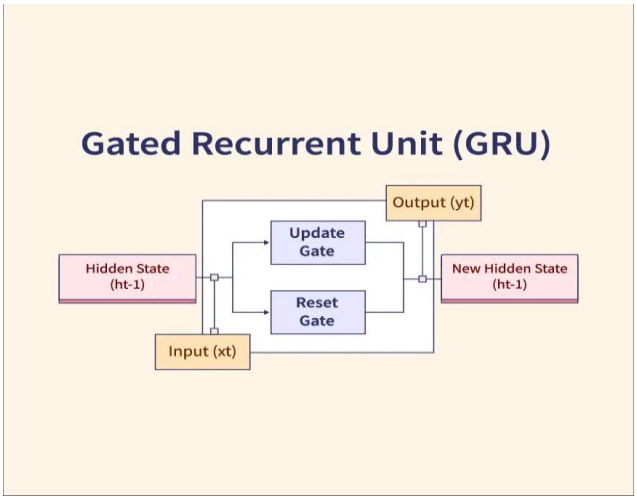
The core distinction between GRUs and LSTMs lies in how they handle the memory state. LSTMs utilize a separate memory cell state updated by three gates: forget, input, and output. In contrast, GRUs replace the memory cell with a "candidate activation vector" and manage it using just two gates: reset and update.

- Reset Gate: This gate acts as a filter, determining how much of the previous hidden state (the network's memory) is no longer relevant and can be discarded.

- Update Gate: This gate controls how much of the newly created candidate activation vector, incorporating the current input and filtered past information, should be incorporated into the new hidden state.

By effectively managing information flow through these gates, GRUs can learn from long sequences and avoid the vanishing gradient problem that hinders traditional RNNs.

Fig. 4. GRU network architecture (overview)



Like other RNN architectures, GRUs process data sequentially, updating their hidden state (internal memory) based on the current input and past information. At each step, the GRU computes a candidate activation vector that blends the current input with the previous hidden state. This candidate vector is then used to create a new hidden state for the next time step.

In essence, GRU cells maintain essential information throughout the network with the help of these two gates. The GRU architecture takes two inputs at each time step:

The previous hidden state (network's memory)
The current input data

These inputs are processed through the update and reset gates to derive the output for the current time step. Finally, a dense layer with SoftMax activation is applied to generate the final output and a new hidden state that is passed on to the next time step.

While like LSTMs in functionality, GRUs offer advantages in terms of simplicity and computational efficiency due to their fewer parameters. This makes them a compelling choice for tasks where resources are limited, or a simpler architecture is preferred.

I will use both LSTM and GRU in my dataset and report the results in the results and conclusion section.

IV. Experimental Setup and Results

A. Dataset

This project leverages a rich dataset constructed from Twitter users. I have conducted an analysis on Twitter users diagnosed with any of the following six mental disorders: Depression (characterized by persistent feelings of sadness and loss of interest in activities), Anxiety (involving excessive worry or fear), Bipolar disorder (marked by shifts in mood, energy, and activity levels), PTSD (Post-Traumatic Stress Disorder, resulting from experiencing or witnessing traumatic events), Borderline personality disorder (involving unstable relationships, emotions, and self-image), and Panic disorder (characterized by sudden and repeated episodes of intense fear). I gathered tweets from these users using a specific pattern: "*I am diagnosed with…*". For instance, if a user posted, "*I am diagnosed with panic since I had a bad car accident*," I categorized them as having panic disorder. Additionally, I included another set of users known as control users, who haven't shared any tweets indicating a diagnosis of a mental disorder. Table I displays the distribution of users within my dataset.

TABLE I. Distribution of Users

	Anxiety	Depression	Control	Borderline	Bipolar	Panic
Number of users	583	1270	4482	100	277	40

Once we identified the user groups, I focused on collecting information about their music preferences. We achieved this by analyzing their tweet timelines for the presence of music links. These links typically point to popular music platforms like Spotify, Apple Music, or SoundCloud. For instance, if a user tweeted "*Currently jamming to Let's All Chant (link to music)*", this tweet would be identified as a music session for that user.

Through this process, I constructed a dataset that captures the music listening habits of users within each group. This dataset includes the music genres these users listened to and shared on Twitter. Figure 5 provides a visualization of the average number of music sessions per user group.

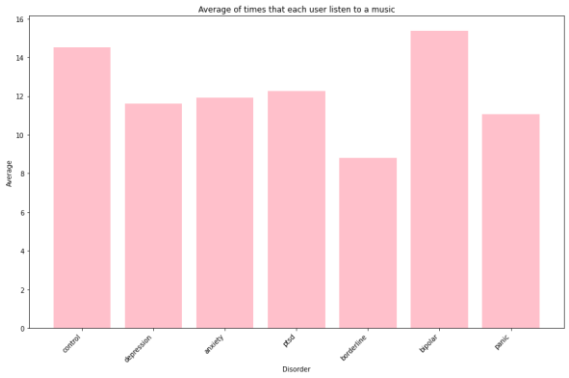


Fig. 5. Average number of music sessions per user group

For the subsequent phase of preparing my dataset, as the focus of my project is predicting the next genre of music that Twitter users might be interested in, I needed to ascertain the genres of the music. To accomplish this, I utilized APIs provided by Spotify, Apple Music, and SoundCloud to retrieve music genres. However, some music tracks were not found by these platforms' APIs. In such cases, I employed the OneMusicAPI, a cost-effective music metadata API, which aggregates data from various online music databases, providing an extensive and accurate range of music data comprising approximately 8,000,000 albums and 4,000,000 artists. Additionally, I utilized the Last.fm API to retrieve music genres.

As the final task involves predicting a single genre, it was imperative to ensure that the genres obtained were singular. However, the aforementioned APIs sometimes returned multiple genres for a single track, such as "Hip Hop" and "R&B." To address this, I attempted to categorize subgenres and identify parent genres using the website: https://www.chosic.com/list-of-music-genres/. Subsequently, I employed genre taxonomies found in literature [2] to further refine the genre categories. As a final step, if a music track

remained associated with two genres and no method existed to reduce them to one, I excluded that track from the analysis. In this study, only the five most prevalent genres from the dataset were considered: Jazz, Hip Hop, Rock, Pop, and Electronic. Any music files categorized under a genre outside of these five were grouped into an "Others" category. Figure 6 illustrates a simplified genre taxonomy utilized in this process.

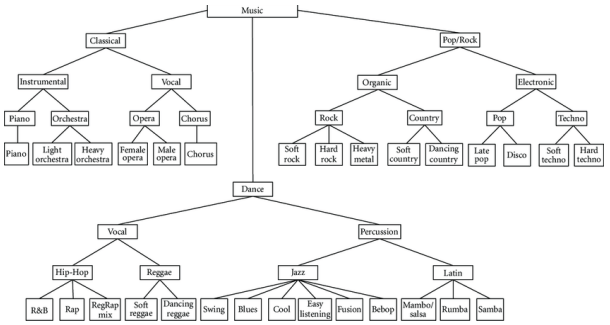


Fig. 6. Music Genre Taxonomy.

This project distinguishes itself from prior research in two key ways. First, while previous studies have explored past music listening habits and some platforms attempt to capture user preferences comprehensively, this project focuses solely on music explicitly shared on Twitter. For example, a user might listen to music throughout the day but only share two specific tracks on Twitter. This project interprets these shared tracks as a deliberate expression of preference, even if they listened to other music between those selections.

Second, in addition to music listening history, we aim to incorporate the emotional state of users into our prediction model. To understand user emotions, we leverage the stress scores associated with their tweets following a music-sharing event. Existing research [3] suggests that music can influence a listener's emotional state for up to 70 minutes.

Therefore, we analyze user tweets posted within three specific time intervals after each music session: 30 minutes, 60 minutes, and 90 minutes. For each tweet during these intervals, we calculate a stress score using the TensiStrength [4] tool. This tool assigns a score between -1 (no stress) and -5 (stressful) based on the text content.

In essence, our current dataset reflects a user with a particular mental state who listens to music and shares it on Twitter (music session). Additionally, for each music session, we have an average stress score calculated from the user's tweets within the three defined time intervals following the music share. Table II shows the number of music sessions after the three defined time intervals. Table III shows the number of each music genres per disorder in different intervals.

TABLE II. NUMBER OF MUSIC SESSIONS IN DIFFERENT INTERVALS

	30m	60 minutes	90 minutes
Music sessions	14469	20531	25796

TABLE III. NUMBER OF MUSIC SESSIONS IN DIFFERENT INTERVALS

30 minutes	Jazz	Electronic	Hip Hop	Pop	Rock	Others
depression	22	60	281	370	334	226
borderline	1	2	33	57	34	20
bipolar	7	24	277	159	130	105
anxiety	24	59	142	324	484	160
ptsd	38	63	476	168	233	181
control	136	412	3419	2833	1382	1843
60 minutes						
depression	28	87	432	579	470	353
borderline	1	6	52	81	59	38
bipolar	11	38	322	251	194	154
anxiety	34	84	232	483	676	234
ptsd	55	91	720	262	347	281
control	181	605	4764	3830	1912	2584
90 minutes						
depression	38	122	545	741	578	470
borderline	3	13	70	100	77	48
bipolar	18	49	405	330	262	193
anxiety	39	103	305	593	760	304
ptsd	65	116	884	331	421	366
control	236	751	6084	4709	2387	3280

B. Preprocessing Dataset

To prepare the user sequence data for LSTM processing, we employ a multi-pronged approach to create informative embedding vectors:

1. **Roberta Text Embeddings:** We leverage the power of Roberta, a large language model (LLM) pre-trained on a massive text corpus [14]. Roberta excels at capturing semantic meaning from textual data. By feeding the cleaned lyrics (after stop word removal, stemming, and lemmatization) into Roberta, we obtain a 768-dimensional vector representation for each lyric snippet. This vector encapsulates the core meaning and relationships within the lyrics.

2. **Mental Disorder Label Embedding:** A binary embedding ([1] for mental disorder users, [0] for control users) is incorporated to account for potential differences in music preferences related to mental health.
3. **Genre One-Hot Encoding:** We encode the music genre using a one-hot encoding scheme. This assigns a unique vector with a "1" in the position corresponding to the genre and "0" elsewhere. This allows the model to differentiate between genres effectively.
4. **Time of Day One-Hot Encoding:** Finally, we capture the time of day the music was listened to using another one-hot encoding scheme. This encoding assigns a vector with a "1" in the position corresponding to the time (morning, afternoon, evening, night) and "0" elsewhere. This allows the model to potentially identify listening patterns based on the time of day.

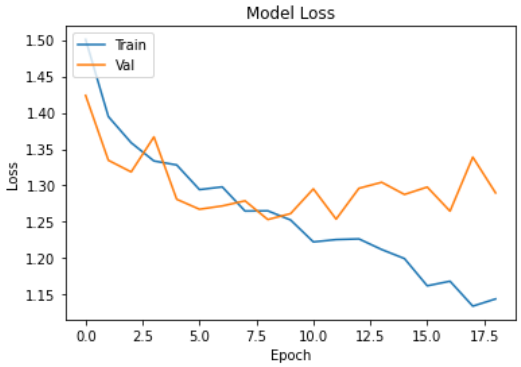
By combining these elements, we create a comprehensive embedding vector with a size of 779 for each user sequence. This vector captures not only the semantic meaning of the lyrics but also additional contextual information about the user and the music itself.

C. LSTM And GRU Models Training

With these enriched user sequence embeddings, we can now train an LSTM model and a GRU model. The model will be designed to predict the genre of the last sequence in a user's listening history, given the sequence of embedding vectors representing all the user's past listening sessions. This approach allows the model to learn from a user's listening patterns and preferences over time. The setup is 2 LSTM layers with 128 units with 2 dense layers at the end. Table IV shows the distribution of the classes. (To prevent overfitting and improve model generalization, we employed an early stopping technique. This technique utilizes a Early Stopping callback function from a deep learning library (Keras). The Early Stopping function monitors the validation loss (val_loss) during training. If the validation loss fails to improve for a specified number of epochs (patience=10 and I have also tried with 5 but the accuracy decreases so the best parameter here is 10), the training process is halted. This helps to regulate the training and prevents the model from memorizing irrelevant patterns in the training data, potentially leading to poor performance on unseen data.)

TABLE IV. NUMBER OF CLEASSES

Genre	Number of samples
Pop	18278
Rock	15476
Electronic	3274
Hip Hop	25780
Jazz	1156
Others	11099



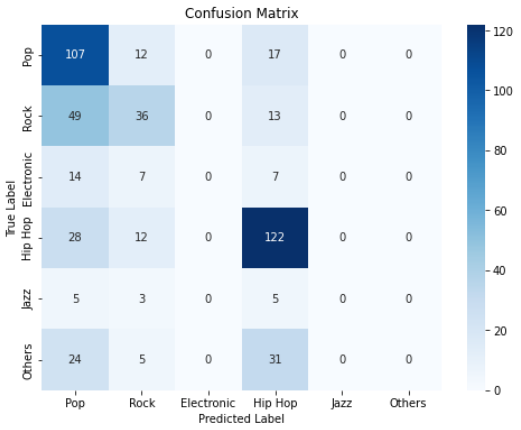


Fig. 7. LSTM results with 6 classifcation

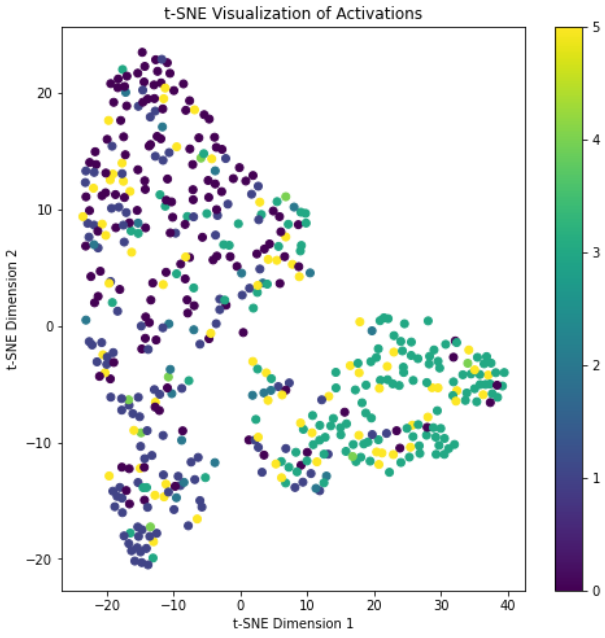


Fig. 8. T-SNE visualization of the last layer in LSTM

The test accuracy for this experiment is 53% for 6 class classifications task which is an acceptable result. **I just**

wanted to try the oversampling method to overcome the imbalance dataset and here you can see the results for overs sampling and the accuracy of model is 48%.

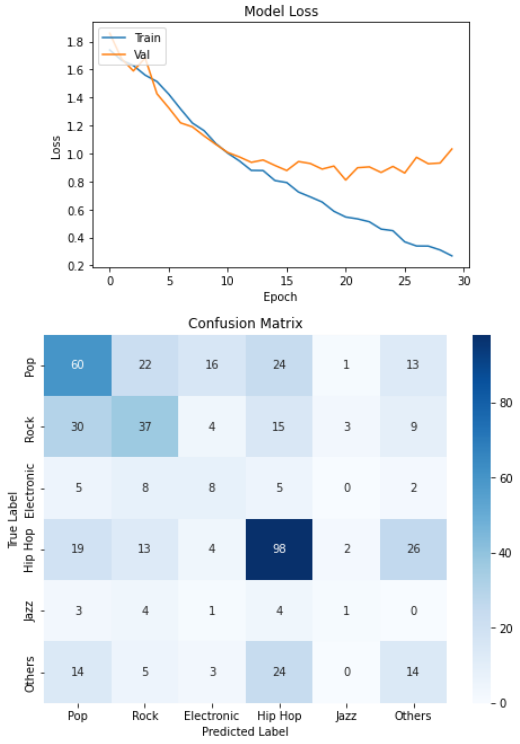


Fig. 9. LSTM results with 6 classifications and oversampling

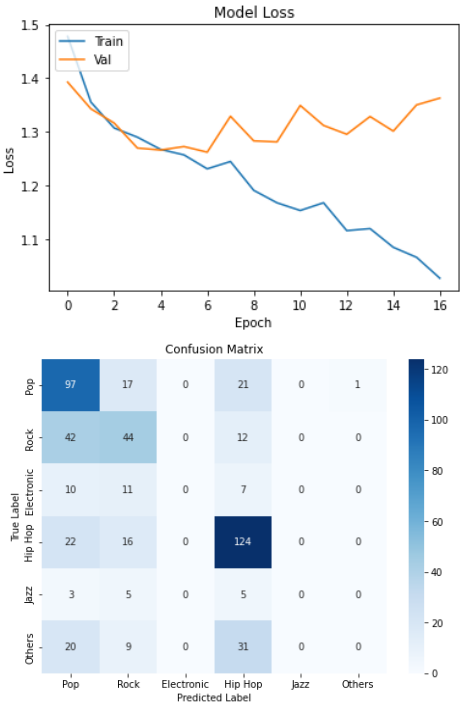


Fig. 10. GRU results with 6 classifications

As shown in the figures 9 and 7, the test acuracy for both models are approximately the same. Since the number of samples in Jazz and Eletronic genres are lower than other, as a next step I want to move these two genres also in Others category.

TABLE V. NUMBER OF CLEASSES

Genre	Number of samples
Pop	18278
Rock	15476
Hip Hop	25780
Others	15529

Here you can see the number of genres in table V that are more similar to each other and more balanced. After training LSTM and GRU on this setup you can see the results below:

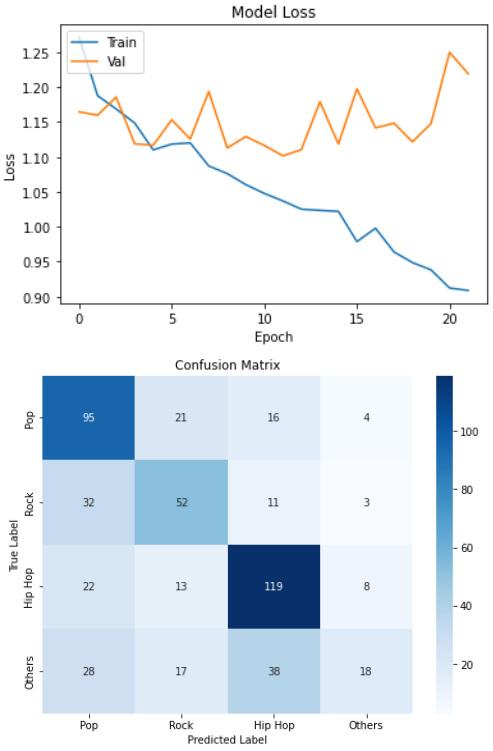


Fig. 11. LSTM results with 4 different genres

The test accuracy is increased to 57% since the model can capture the patterns in each history and genre better. The results for GRU model is almost the same at accuracy 56%.

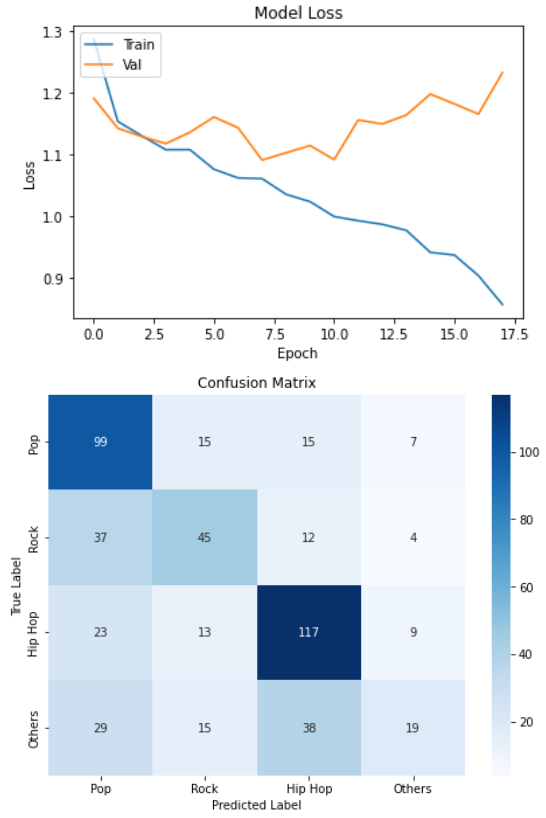


Fig. 12. GRU results with 4 different genres

This bias is reflected in the confusion matrix, where a significant portion of the errors fall under the "Other" category. In essence, when the model lacks confidence in its prediction, it defaults to assigning the user's next genre as "Other."

To address this curiosity and explore potential improvements, I took the following steps:

1. **Removed "Other" Genre:** I removed the "Other" genre from the classification task to see if excluding this large category would impact accuracy.
2. **Attention Layer and Stress Score:** I investigated the effect of adding an attention layer [15] to the LSTM model. Attention layers can help the model focus on the most relevant parts of the sequence data during prediction. Additionally, I explored incorporating the user's stress score associated with the 30-minute interval following the music session by concatenating it with the existing embedding vector. This additional data point might provide insights into the user's emotional state, potentially influencing their music selection.

Through these modifications, I aimed to evaluate whether the model's accuracy for predicting specific genres (excluding "Other") could be improved.

Genre	Number of samples
Pop	18278
Rock	15476
Hip Hop	25780

TABLE VI. NUMBER OF CLEASSES

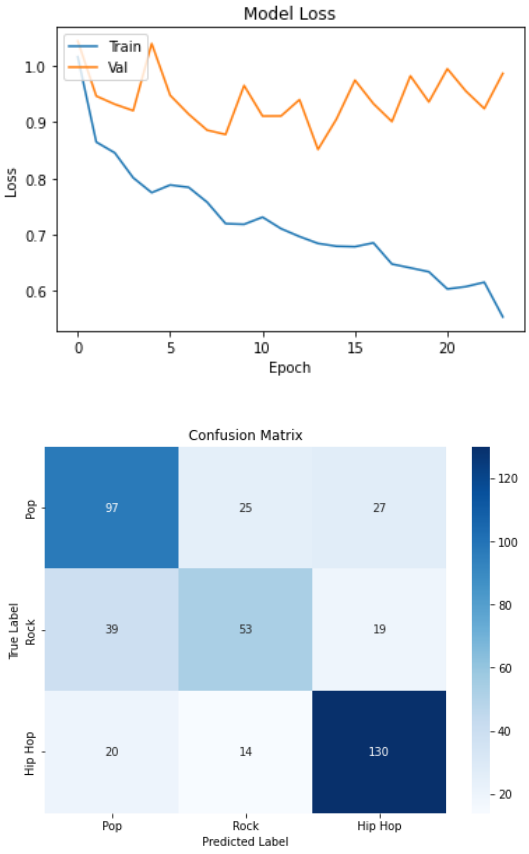


Fig. 13. LSTM results with 3 different genres and stress levels and attention

Here as we can see, the accuracy of my model increases to 66% which is also acceptable for 3 class classifications. The results for GRU are a little lower but close to LSTM.

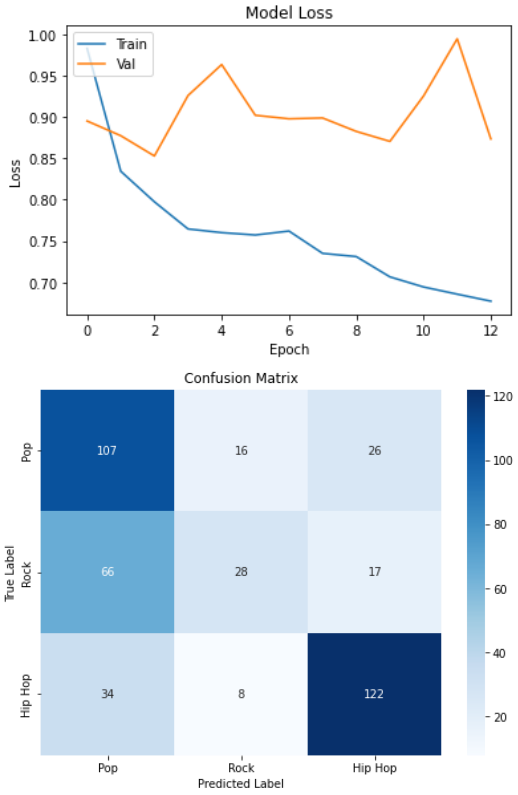


Fig. 14. GRU results with 3 different genres and stress levels and attention

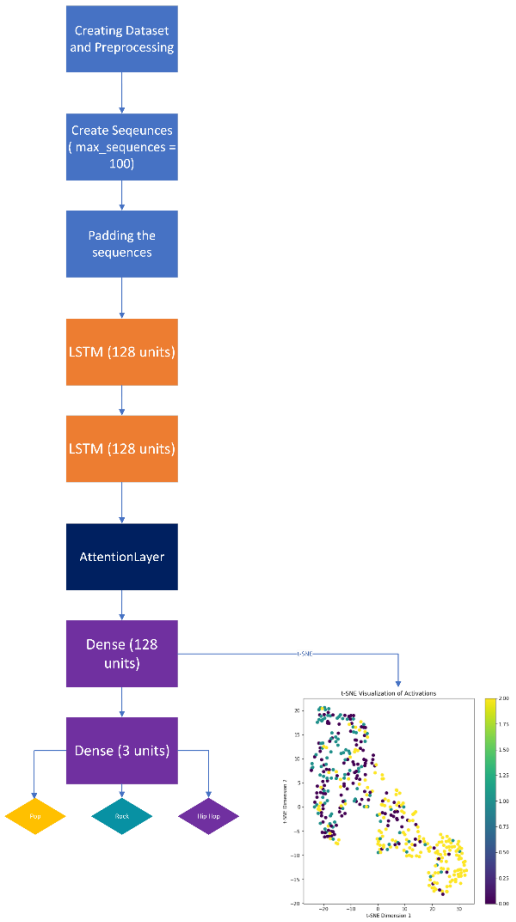


Fig. 15. Overall Architecture with T-SNE visualization of activations

Figure 15 presents an overview of the project workflow and the final architecture that achieved the best results for 3-class classification (pop, rock, hiphop). It also includes a t-SNE visualization of the data before the final classification layer. The t-SNE visualization reveals a key challenge: pop and rock genres appear clustered closely together. This suggests difficulty in distinguishing between these genres, which is further confirmed by the confusion matrix.

D. Compartition Between LSTM And GRU Models

Number of genres	Attention layer	Stress scores	LSTM	GRU
6	x	x	0.53	0.52
4	x	x	0.57	0.56
3	✓	✓	0.66	0.6

As we can see, the results for LSTM are better than GRU since the architecture is almost more complicated than GRU and I know reducing the number of classes to reach better accuracy is not necessarily improvement, but **I just want to try and satisfy my curiosity to know whether better results can be achieved by removing some sparse genres or not.**

V. LIMITATION AND FUTURE WORKS

This study investigated the efficacy of LSTM/GRU models utilizing embedding vectors to predict music genres based on user listening sequences. While the approach displayed promise, several limitations warrant consideration for future research endeavors.

One notable challenge lies in the inherent data imbalance stemming from the multitude of genres categorized as "Other." To mitigate this, prospective investigations could explore techniques for managing imbalanced datasets. Apart from conventional methods like oversampling or undersampling minority classes, which I've already employed but yielded limited improvements, novel approaches such as hierarchical classification strategies hold potential. Furthermore, augmenting the model with richer user context data beyond mere listening history could enhance accuracy. Integrating demographic information, user-stated musical preferences, or mood indicators derived from music features could offer valuable insights into user behavior.

Exploration of alternative or more intricate model architectures, such as convolutional neural networks (CNNs) for music feature extraction or recurrent neural networks (RNNs) with varied gate structures, may unlock further enhancements in genre prediction performance. Additionally, developing techniques to elucidate the rationale behind the model's predictions would be beneficial. Methods like Layer-wise Relevance Propagation (LRP) could shed light on which segments of the user sequence data contribute most significantly to genre prediction.

Another limitation pertains to the dataset's reliance solely on users' self-reported tweets to label them as part of the mental disorder group, lacking clinical assessment. Despite efforts to

verify users' account metadata from Twitter, further investigation is warranted to confirm their mental health status, considering the potential for misrepresentation or inaccuracy.

By addressing these limitations and pursuing these promising avenues for future work, we can refine the music genre prediction model, facilitating the development of a more robust and user-centric music recommendation system.

While this study focused on model selection and optimization, future research should delve deeper into understanding user preferences. Social media platforms offer a rich tapestry of user activity data beyond listening history. By incorporating user posts, shares, and interactions related to music on these platforms, music recommendation systems can gain a more comprehensive understanding of individual tastes. Investigating techniques for effectively integrating social media data with traditional listening history data holds immense potential for the future of music recommendation. By leveraging the wealth of information available on social media, music recommendation systems can evolve to provide users with even more personalized and tailored listening experiences.

VI. CONCLUSION

This study compared the effectiveness of LSTM and GRU models for music genre prediction using user listening sequences and embedding vectors. We explored the impact of the number of genres (6, 4, and 3) on model performance, along with the inclusion of an attention layer and stress scores.

The results revealed a trend where LSTMs achieved slightly higher accuracy compared to GRUs across all genre configurations. For instance, with 6 genres, the LSTM model reached an accuracy of 0.53, while the GRU model achieved 0.52. This pattern continued with 4 and 3 genres, suggesting a potential benefit of the LSTM architecture's increased complexity for this specific task. However, it's important to acknowledge the trade-off between accuracy and computational efficiency. LSTMs generally require more computational resources compared to GRUs. The choice between these models might depend on the specific application and available resources.

In conclusion, this study suggests that LSTMs might hold a slight edge over GRUs for music genre prediction in this context. However, further investigation into factors like computational efficiency and hyperparameter optimization is recommended for a more comprehensive understanding.

VII. REFFERENCES

[1] Song, Y., Dixon, S., & Pearce, M. (2012, June). A survey of music recommendation systems and future perspectives. In *9th international symposium on computer music modeling and retrieval* (Vol. 4, pp. 395-410).

[2] Barbedo, J. G. S., & Lopes, A. (2006). Automatic genre classification of musical signals. *EURASIP Journal on Advances in Signal Processing*, *2007*, 1-12.

[3] Thoma, M. V., La Marca, R., Brönnimann, R., Finkel, L., Ehlert, U., & Nater, U. M. (2013). The effect of music on the human stress response. *PloS one*, *8*(8), e70156.

[4] M. Thelwall, TensiStrength: Stress and relaxation magnitude detection for social media texts, Inf. Process. Manage. 53 (1) (2017) 106–121

[5] Gulmatico, J. S., Susa, J. A. B., Malbog, M. A. F., Acoba, A., Nipas, M. D., & Mindoro, J. N. (2022). SpotiPred: A machine learning approach prediction of Spotify music popularity by audio features. *2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T)*, 1–5.

[6] Kaminski, G., Dridi, S., Graff, C., & Gentaz, E. (2009). Human ability to detect kinship in strangers' faces: effects of the degree of relatedness. *Proceedings of the Royal Society B: Biological Sciences*, *276*(1670), 3193–3200.

[7] Allawadi, K., & Vij, C. (2023). A Smart Spotify Assistance and Recommendation System. *2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT)*, 286–291.

[8] Elbir, A., & Aydin, N. (2020). Music genre classification and music recommendation by using deep learning.

[9] Shin, S.-H., Yun, H.-W., Jang, W.-J., et al.: 'Extraction of acoustic features based on auditory spike code and its application to music genre classification', IET Signal Process., 2019, 13, (2), pp. 230–234

[10] Bahuleyan, H. (2018). Music genre classification using machine learning techniques. *arXiv preprint arXiv:1804.01149*.

[11] S. Joshi, T. Jain and N. Nair, "Emotion Based Music Recommendation System Using LSTM - CNN Architecture," *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, 2021, pp. 01-06, doi: 10.1109/ICCCNT51525.2021.9579813.

[12] M. Slaney. Web-scale multimedia analysis: Does content matter? MultiMedia, IEEE, 18(2):12–15, 2011.

[13] Honglak Lee, Peter Pham, Yan Largman, and Andrew Ng. Unsupervised feature learning for audio

[14] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

[15] Wu, C., Wu, F., Qi, T., Huang, Y., & Xie, X. (2021). Fastformer: Additive attention can be all you need. *arXiv preprint arXiv:2108.09084*.