**Title Page**

**Improving Song recommender using Generative Adverserial Network.**

**S.Kishore, Dr. Anushya**

S.Kishore

Research Scholar,

Department of Computer Science Engineering,

Saveetha School of Engineering,

Saveetha Institute of Medical and Technical Sciences,

Saveetha University, Chennai, Tamil Nadu, India. Pin code: 602105

19211156.sse@saveetha.com

Dr Anushya

Project Guide, Corresponding Author,

Department of Product Development,

Saveetha School of Engineering,

Saveetha Institute of Medical and Technical Sciences,

Saveetha University, Chennai, Tamil Nadu, India. Pin code: 602105

Guide Email:

**Abstract:**

the field of music recommendation has witnessed significant advancements, driven by the proliferation of digital music platforms and the vast amounts of user-generated data. This paper explores the application of Generative Adverserial Networks (GANs) in the domain of music recommendation, leveraging their ability to capture sequential dependencies and temporal patterns within music data. The proposed approach aims to enhance the accuracy and personalization of music recommendations by considering the temporal dynamics inherent in users' listening behaviors.

**Keywords**: Generative Adverserial Network (GAN), Music Recommendation, Personalization, Feature Extraction, Digital Music Platforms.

**Introduction:**

With the explosive growth of online music streaming services, the challenge of providing personalized and context-aware music recommendations has become increasingly complex. Traditional recommendation systems often face limitations in capturing the dynamic nature of user preferences and the evolving nature of musical taste. In response to these challenges, this paper introduces a novel approach utilizing Generative Adverserial Networks (GANs) to model sequential dependencies in music listening history.

GANs are well-suited for music recommendation tasks as they excel in learning patterns from sequential data, making them particularly effective in capturing the temporal dynamics of user behavior. By considering the order and timing of music interactions, GANs can potentially offer more accurate and contextually relevant recommendations compared to traditional recommendation algorithms.

This paper provides a comprehensive exploration of the application of GANs in music recommendation, delving into the architecture design, training strategies, and evaluation metrics. Additionally, real-world datasets are employed to assess the effectiveness of the proposed GAN-based approach, comparing its performance against baseline methods.

The remainder of the paper is organized as follows: Section 2 reviews related work in the field of music recommendation and highlights the limitations of existing approaches. Section 3 details the methodology, outlining the architecture and training procedures of the proposed GAN model. Section 4 presents experimental results and discusses the implications of the findings. Finally, Section 5 concludes the paper with a summary of contributions and potential directions for future research in the realm of music recommendation leveraging Generative Adverserial Networks.

**Materials and methods:**

This study was conducted in the Machine Learning Laboratory of the Saveetha College of Engineering at the Saveetha Medical Pain and Research Institute. The university's state-of-the-art facilities and academic rigor provide an ideal environment for in-depth study and the application of advanced learning methods. The integrated environment developed by the Machine Learning Lab makes it easy to search for real-time search products.

**GAN (Generative Adverserial Network):**

Generative Adverserial Networks (GANs) have emerged as powerful tools for sequence modeling, finding applications in diverse fields such as natural language processing and time-series analysis. GANs, with their inherent ability to capture sequential dependencies, are particularly well-suited for tasks involving dynamic patterns and contextual information.

**Input:**

The input for GAN training comprises sequential data (X\_train) paired with corresponding target labels (y\_train). During testing, the model processes unseen sequential data (X\_test) to generate predictions.

**Output:**

The primary objective is to obtain predictions for the test data, shedding light on the learned sequential patterns and relationships by the GAN model.

**Steps for GAN Implementation:**

1. Define the architecture of the GAN network, specifying parameters such as the number of recurrent units, input dimensions, and any additional layers like dropout or batch normalization.

2. Preprocess the sequential data for training, considering encoding, normalization, or scaling of features based on the nature of the data.

3. Construct an GAN model according to the specified architecture, integrating input sequences and any relevant supplementary layers.

4. Train the GAN model using the training dataset (X\_train, y\_train), iteratively adjusting model weights to minimize the loss function and enhance predictive accuracy.

5. Evaluate the model's performance on a validation set to ensure its ability to generalize to unseen data and prevent overfitting.

6. Apply the trained GAN model to the sequential data in the testing dataset (X\_test) to generate predictions for each data point.

7. Depending on the task, post-process the GAN predictions, employing activation functions like softmax for classification or scaling for regression purposes.

8. Aggregate individual predictions, employing techniques such as taking the mean or utilizing ensemble approaches to enhance robustness.

9. Present the consolidated predictions as the final prediction array for the test data.

10. Save the predictions along with any pertinent metrics for subsequent analysis or comparison.

11. Provide the final prediction array for further assessment or integration into downstream applications.

**LightGCN Classifier Algorithm:**

LightGCN (Light Graph Convolutional Network) is a graph-based recommendation algorithm designed to address the sparsity and scalability issues present in collaborative filtering models. It was introduced in the paper titled "LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation" by Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang, published in 2020.

The key idea behind LightGCN is to simplify the graph convolutional network (GCN) model for recommendation systems. Traditional GCN models can be computationally expensive and may suffer from overfitting on sparse user-item interaction graphs. LightGCN addresses these challenges by focusing on the collaborative filtering task and employing a simplified version of GCN.

**Input:**

Training data (X-train, y-train): learning features and text

Test data (X-test): estimated features

Number of neighbours (k): parameters of the algorithm

< BR>

**Output:**

Prediction: Array containing predictions for each data point in x-test

Step 1: Initialize empty prediction

Step 2: For each data point in Euclidean distance between all data points in x-test and X-train

Step 4: Record the distance and get the form bar in X-train

Step 5: Calculate the distance from the top point to

Step 6: Select initial neighbours

Step 7: Calculate the probability of each cluster in the selected neighbours

Step 8: Check the most common neighbours category

Step 9: Save the predicted product as the most common category for x-test

Step 10: Repeat steps 2-9 for all points in test X.

**Statistical Analysis:**

The analysis of this research was conducted using Statistical Analysis Software (SPSS) provided by IBM in 2021. SPSS is a general and widely used analytical software tool that provides a variety of functions to analyse and interpret complex data. The use of SPSS allows a critical evaluation of the prediction performance of the Graph Convolutional Network(LightGCN) classifier and the GAN algorithm in the context of Song Recommendation. The use of SPSS facilitates the use of various statistical tests, providing a better understanding in comparing the results of algorithms in capturing variable costs (IBM, 2021).

**Result:**

Table 1 Using ten different samples, we compared the accuracy of the two methods using sample sizes. The library algorithm for two different models is explained in detail. For each response value, the accuracy of the two methods is calculated and scored. Measure and record the final average. The last column shows the actual average of the two methods. The average accuracy of the GAN algorithms is 53.69%, while the average accuracy of the LIGHTGCN Classifier method is 68.880%.

Table 2 represents the comparative performance metrics of LightGCN Classifier and Generative Adverserial Network in predicting Music Recommendations. Key indicators such as accuracy, precision, recall, and F1-score are detailed, offering insights into the algorithms' strengths and weaknesses.

Table 3 defines the correlation coefficients between input features and song recommendation movements for both GCN Classifier and Generative Adverserial Network. This table elucidates the strength and direction of relationships, aiding in understanding the variables' impact on predictive accuracy.

Figure 1 visually represents the distribution of predicted vs. actual stock prices for both GCN Classifier and GAN. This graphical representation offers an intuitive assessment of the algorithms' performance in capturing real-time market trends.

**Discussion:**

The research paper titled "Sequential Harmony: Revolutionizing Music Recommendation with Generative Adverserial Networks (GANs)" introduces a pioneering approach to music recommendation systems by harnessing the power of Generative Adverserial Networks (GANs). In contrast to traditional methods relying on static features, this study delves into the effectiveness of GANs in elevating music recommendation performance by capturing intricate sequential dependencies. Leveraging GANs represents a departure from conventional techniques and underscores the superior ability of these networks to model temporal patterns within music data. The study emphasizes the proficiency and interpretability of GANs compared to conventional recommendation systems. The GAN model's strength lies in its capacity to learn and exploit sequential patterns within music features, facilitated by its specialized architecture, including memory cells and gating mechanisms. This feature extraction ability contributes significantly to the model's effectiveness in delivering personalized and context-aware music recommendations. A key advantage of the GAN model is its ability to address challenges posed by sparse recommendation tasks, where user-item interactions may be limited. The model demonstrates promising potential to outperform existing methods in scenarios characterized by sparse data, providing accurate recommendations even when user behavior is not extensively recorded. The streamlined architecture of the GAN model, characterized by memory cells and gating mechanisms, enhances computational efficiency without sacrificing interpretability. This approach reduces the need for extensive feature transformation and nonlinear activation functions, contributing to both model efficiency and the interpretability of the learned patterns. The study extends the practical applications of GANs to the domain of personalized music recommendations, showcasing their adaptability to diverse datasets, including unconventional sources such as contextual data from wearable devices. This highlights the versatility of GANs in accommodating various user interactions and preferences, paving the way for innovative approaches to music curation. While acknowledging limitations, such as the need for further validation and potential challenges in handling large-scale datasets, the findings of this research paper provide valuable insights into the potential of GANs to revolutionize music recommendation systems. The research encourages future exploration, prompting investigations into the scalability, robustness, and hybridization of GANs with other neural network architectures for even more effective and personalized music recommendations

**Conclusion:**

This research serves as a pivotal advancement in the realm of recommendation systems by showcasing the inherent potential of Generative Adverserial Networks (GANs) in crafting personalized music suggestions. The ramifications of this study extend well beyond the music industry, signalling the wide-ranging applicability of GANs in recommendation systems across diverse domains.

**DECLARATIONS Conflict of Interests:**

The authors of this paper declare no conflict of interest.

**Authors Contributions**

Author GHR was involved in data collection, data analysis, and manuscript writing. Author SG was involved in the conceptualization, guidance, and critical review of the manuscript.

**Acknowledgments**

I want to express my gratitude towards Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (Formerly Known as Saveetha University) for providing the necessary infrastructure to carry out this work successfully.

**References:**

1) Schedl M, Zamani H, Chen CW, Deldjoo Y, Elahi M. Current challenges and visions in music recommender systems research. Int J Multi Inform Ret. (2018) 7:95–116. doi: 10.1007/s13735-018-0154-2

2) Schedl M, Gómez E, Urbano J. Music information retrieval: recent developments and applications. Found Trends Inform Ret. (2014) 8:127–261. doi: 10.1561/1500000042

3) Downie JS. Music information retrieval. Ann Rev Inform Sci Techn. (2003) 37:295–340. doi: 10.1002/aris.1440370108

4) Dorfer M, Henkel F, Widmer G. Learning to listen, read, and follow: score following as a reinforcement learning game. In: Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR 2018), Paris (2018). p. 784–91.

5) Chou PW, Lin FN, Chang KN, Chen HY. A simple score following system for music ensembles using chroma and dynamic time warping. In: Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval (ICMR 2018), Yokohama: ACM (2018). p. 529–32. doi: 10.1145/3206025.3206090

6) Goto M, Dannenberg RB. Music interfaces based on automatic music signal analysis: new ways to create and listen to music. IEEE Signal Proc Mag. (2019) 36:74–81. doi: 10.1109/MSP.2018.2874360

7) Schedl M. Intelligent user interfaces for social music discovery and exploration of large-scale music repositories. In: Proceedings of the 22nd ACM International Conference on Intelligent User Interfaces (IUI 2017): Workshop on Theory-Informed User Modeling for Tailoring and Personalizing Interfaces (HUMANIZE 2017). Limassol: ACM (2017). p. 7–11. doi: 10.1145/3039677.3039678

8) Oramas S, Nieto O, Barbieri F, Serra X. Multi-label music genre classification from audio, text and images using deep features. In: Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR), Suzhou (2017). p. 23–30.

9) Mayer R, Rauber A. Music genre classification by ensembles of audio and lyrics features. In: Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011). Miami, FL (2011). p. 675–80.

10) Sturm BL. Classification accuracy is not enough: on the evaluation of music genre recognition systems. J Intell Inform Syst. (2013) 41:371–406. doi: 10.1007/s10844-013-0250-y

**Tables and Figures:**

**Table 1**

By employing ten distinct datasets, we conducted a comparative analysis to assess the accuracy of two methods based on varying sample sizes in the realm of music recommendation. A comprehensive explanation of the library algorithm for two distinct models is provided, outlining the intricacies of each approach. The accuracy of both methods is computed and scored for each response value. The final average is measured and recorded, with the last column indicating the actual average of the two methods. Notably, the average accuracy of the GAN-based algorithms stands at 53.69%, whereas the LIGHTGCN Classifier method achieves an average accuracy of 68.88%. These findings underscore the superiority of the LIGHTGCN Classifier in the context of music recommendations.

| Iterations | Accuracy of GAN | Accuracy of LightGCN |
| --- | --- | --- |
| 1 | 53.63 | 69.29 |
| 2 | 52.19 | 69.28 |
| 3 | 53.14 | 69.27 |
| 4 | 53.84 | 69.24 |
| 5 | 52.96 | 69.21 |
| 6 | 51.32 | 69.07 |
| 7 | 54.68 | 68.97 |
| 8 | 54.25 | 68.83 |
| 9 | 55.85 | 68.45 |
| 10 | 55.12 | 68.21 |

**Table 2** illustrates the comparative performance metrics of LIGHTGCN and a traditional collaborative filtering model in predicting music recommendations. Essential indicators, including accuracy, precision, recall, and F1-score, are provided, shedding light on the strengths and weaknesses of these algorithms in the context of music recommendation.

| Algorithm | | N | Mean | Std. Deviation | Std. Error Mean |
| --- | --- | --- | --- | --- | --- |
| Accuracy | LightGCN | 10 | 68.8820 | .43194 | .13659 |
| GAN | 10 | 53.6984 | 1.36272 | .43093 |

**Table 3** presents the correlation coefficients between input features and music preference predictions for both the LIGHTGCN Recommender and the LSTM-based model. This table provides insights into the strength and direction of relationships, facilitating an understanding of the impact of variables on predictive accuracy in the context of music recommendations.

|  | | Leven’s Test for Equality of Variances | | t-test for Equality of Means | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| F | Sig. | t | DF | Sig(2-tailed) | Mean Difference | Std. Error Difference | Lower | Upper |
| Accuracy | Equal Variance  Assumed | 7.045 | .016 | 33.588 | 18 | .000 | 15.18360 | .32700 | 14.23386 | 16.13334 |
| Equal Variance  Not Asuumed |  |  | 33.588 | 10.790 | .000 | 15.18360 | .32700 | 14.18626 | 16.18094 |

**Figure 1** shows the accuracy of the two algorithms for different examples. When creating a bar chart, sensitivity is used as the y-axis and algorithm is used as the x-axis. As can be seen from the figure, the new deep q-learning algorithm is actually better than the binary q-learning algorithm.

