

음악 장르 분류를 위한 데이터 생성 및 머신러닝 적용 방안[☆]

Generating Data and Applying Machine Learning Methods for Music Genre Classification

엄 빛 찬¹ 조 동 휘¹ 남 춘 성^{1*}
Bit-Chan Eom Dong-Hwi Cho Choon-Sung Nam

요 약

본 논문은 머신러닝을 활용하여 많은 양의 음악 데이터를 분류하여 장르 정보가 입력되어 있지 않은 음악 장르 분류 정확도 향상을 목표로 한다. 음악의 장르를 구분하기 위해 기존 연구에서 많이 사용되던 GTZAN 데이터 세트 대신 직접 데이터를 수집하고 전처리하는 방안을 제시한다. 이를 위해 GTZAN 데이터 세트보다 분류 성능이 뛰어난 데이터 세트를 생성하기 위해 Onset의 에너지 레벨이 가장 높은 일정 구간을 추출한다. 학습에 사용하는 음악 데이터의 주요 특성으로는 Mel Frequency Cepstral Coefficient(MFCC)를 포함한 57개의 특성을 이용한다. 전처리된 데이터를 통해 Support Vector Machine(SVM) 모델을 이용하여 Blues, Classical, Jazz, Country, Disco, Pop, Rock, Metal, Hiphop으로 분류한 학습 정확도가 85%를 기록하였고, 테스트 정확도가 71%를 보여주었다.

☞ 주제어 : 음악 장르 분류, 머신러닝, 음악 특징 추출, GTZAN, 서포트 벡터 머신

ABSTRACT

This paper aims to enhance the accuracy of music genre classification for music tracks where genre information is not provided, by utilizing machine learning to classify a large amount of music data. The paper proposes collecting and preprocessing data instead of using the commonly employed GTZAN dataset in previous research for genre classification in music. To create a dataset with superior classification performance compared to the GTZAN dataset, we extract specific segments with the highest energy level of the onset. We utilize 57 features as the main characteristics of the music data used for training, including Mel Frequency Cepstral Coefficients (MFCC). We achieved a training accuracy of 85% and a testing accuracy of 71% using the Support Vector Machine (SVM) model to classify into Classical, Jazz, Country, Disco, Soul, Rock, Metal, and Hiphop genres based on preprocessed data.

☞ keyword : Music genre classification, Machine Learning, Music feature extraction, GTZAN, Support vector machine

1. 서 론

최근 1인 미디어의 부상으로 개인이 만든 미디어 콘텐츠들이 표 1과 같이 여러 플랫폼을 통해 제공되고 있다. 미디어를 제작할 때 배경음악(BGM, BackGround Music)은 콘텐츠 제작에 필수적인 요소로 배경음악은 단순한 음향효과에 그치지 않고 콘텐츠를 접하는 사람의 집중력과 인지력에 영향을 미친다[1]. 또한, 배경음악은 장소에

따라 연주되는 음악 장르와 종류에 따라 고객들의 소비 심리가 조절되기도 한다[2]. 배경음악을 결정하는 것은 콘텐츠 시청자와의 상호작용을 높여 콘텐츠 만족도와 새로운 소비 욕구를 일으킬 수 있다. 따라서 콘텐츠를 제작할 때 적절한 장르의 음악을 선정하는 것은 굉장히 중요하다. 하지만 음악은 음악의 흐름에 있어 변화가 다양하고 하나의 음악 내의 다양한 장르의 특색이 포함되어 있어 음악을 듣는 것만으로 청취자가 어떤 장르의 음악인지 판단하는 것은 쉬운 일이 아니다[3]. 전문적인 지식을 가진 사람이라도 다양한 장르적 특성을 포함하는 노래를 하나의 장르로 완벽하게 구분하는 것은 어렵다.

모델 학습에 있어서 현재까지도 G. Tzanetakis 및 P. Cook에 의해 2002년 발표된 GTZAN 데이터 세트가 가장 많이 활용되고 있다[6][7]. GTZAN 데이터 세트는 개인 CD, 라디오, 녹음 파일들을 다양한 소스에서 2000~2001년 수집된 자료들로 10개의 장르(Blues, Classical,

¹ Department of Software Convergence Engineering, Inha University, Incheon, 15798, Korea

* Corresponding author (namgun99@inha.ac.kr)

[Received 21 February 2024, Reviewed 26 February 2024(R2 09 May 2024, R3 01 July 2024), Accepted 24 July 2024]

[☆] 이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 지역지능화혁신인재양성사업(IITP-2024-RS-2023-00259678, 50%)과 정부(교육부와 한국연구재단)의 재원으로 2019년 대한민국 교육부와 한국연구재단의 지원을 받아 수행된 연구임(NRF-2019S1A5A2A03040702, 50%)

Country, Disco, Hiphop, Jazz, Metal, Pop, Reggae, Rock)로 이루어지며, 각 장르별 100개의 곡, 3초 또는 30초 길이의 22,050Hz 오디오 파일로 이루어져 총 1,000개의 음악 파일로 구성된다. GTZAN을 이용한 음악 장르 분류를 하는 방법으로 머신러닝 방법과 딥러닝 방법이 있다. GTZAN 데이터는 학습을 진행할 때 원본 30초를 그대로 학습에 이용하여 1,000개의 데이터로 진행하거나, 원본 30초에서 3초 단위로 데이터를 분할하여 10개의 데이터를 만들어 총 10,000개의 데이터로 학습을 진행한다. 먼저 3초로 나눈 데이터로 학습을 진행했을 때 머신러닝 방법에서는 Logistic Regression(LR) : 73.7%, Random Forests(RF) : 89.5%, Support Vector Machines(SVM) : 88.0%, k-Nearest Neighbors(KNN) : 90.3%, Decision Trees(DT) : 66.1%, Extra Tree(ET) 92.3%, Linear Discriminant Analysis(LDA) 70.3%, XGBoost(XGB) : 89.2%의 정확도를, 딥러닝에서는 Convolutional Neural Networks(CNN) 91.0%로 최대 92.3%의 정확도를 보였다 [6,7]. 위 결과를 보았을 때 머신러닝 방법과 딥러닝 방식을 통한 정확도는 비슷한 결과를 나타내고 있다. 하지만 ‘Shazam’ 음악 플랫폼을 활용한 튀르키예에서 가장 많이 검색한 노래 10개를 선정해 추가 데이터로 테스트를 진행할 결과 71.6%의 정확도가 나타났다[8]. 이는 GTZAN을 학습한 모델이 한 음원에 있는 3초 단위의 데이터 10개 중에서 일부는 학습 데이터로 사용되고, 일부는 테스트 데이터로 사용되므로 테스트에 영향을 주어 분류 정확도가 92.3%로 높게 나오는 것이다. 하지만 위 연구[8]와 같이 음원이 학습되지 않은 음악을 분류할 경우 정확도가 떨어짐을 알 수 있다. 따라서 음원을 학습시 음원별로 나누어 모델을 학습하여야 한다. KNN방식을 통해 각 3초 단위로 분리된 GTZAN 데이터를 음원 단위로 분리하여 실험한 결과 37%의 정확도를 보였다. 이러한 이유로 3초 단위의 데이터 세트를 사용하는 것은 잘못된 방법이며, 음원의 특징을 반영하기 위해서 GTZAN에서 제공하는 30초 단위의 데이터 세트와의 비교가 필요하다. GTZAN 30초 데이터 세트를 적용하여 머신러닝 방법을 이용한 경우 LR : 81.0%, RF : 75.7%, SVM : 75.4%, KNN : 72.8%의 정확도를 나타내었다. 그리고 딥러닝 방법에서 CNN 53.5%, Multilayer Perceptron(MLP) : 75.2%, Long Short-Term Memory(LSTM) 69.0%, SVM과 LSTM의 앙상블 조합에서는 89.0%로 최대 89.0%의 정확도를 보였다[9, 10]. 하지만, GTZAN 데이터 세트에는 동일한 노래 혹은 동일한 아티스트 음원의 반복, 잘못된 라벨링, 왜곡과 같

은 자체적 결함이 존재하기 때문에 정확도는 더 높을 수 밖에 없다.[11,12]

따라서 본 연구에서는 결함이 있는 GTZAN 데이터 세트가 아닌 새로운 데이터 세트를 구성하고 머신러닝을 통해 결과를 도출한다. 기존의 GTZAN과 음원의 앞부분 30초 동안 추출한 데이터 세트 Front Energy Data Set(FEDS)와 각 음원의 30초동안 에너지 파워가 가장 높은 지점을 추출한 데이터 세트 Segment Energy Date Set(SEDS)를 통해 머신러닝 모델별 분류 정확도를 비교 평가한다.

(표 1) 음원제공 플랫폼별 제공 장르
(Table 1) Genres provided by music service platform

플랫폼(장르 수)	제공 장르
Viodio (11)	Cinematic, NewAge, Children, EDM, Jazz, Rock, Hiphop, Embient, World, Punk, Acoustic
Epidemic Sound (12)	Eletronic, Beats, Hiphop, Small Emotions, Pop, Acoustic, Rock, Funk, Classical, RnR, Jazz, Country
Artist (27)	Acoustic, Ambient, Blues, Children, Cinematic, Classical, Corporate, Country, Electronic, Pantasy, Folk, Funk, Hiphop, Holiday, Indie, Jazz, Latin, Lofi, Lounge, Pop, Reggae, Retro, Rock, Singer-Songwriter, Soul, World, Worship
MusicBed (20)	Alternative, Ambient, Blues, Cinematic, Classical, Country, Electronic, Folk, Hiphop, Indie, Jazz, Pop, Post Rock, RnB, Rock, Singer-Songwriter, Soul, Spoken Word, Vintage, World

2. 데이터 세트 구성 방법

자체적으로 생성한 데이터 세트는 구분을 위해 데이터 생성 방식에 따라 FEDS(Front Energy Data Set)와 SEDS(Segment Energy Data Set)로 명명하였다. FEDS는 자체적으로 수집한 데이터 세트로 10개의 장르(Classical, County, Disco, Hiphop, Jazz, Metal, Reggae, Rock, Pop, Blues)를 장르별로 100곡씩 총 1,000곡의 첫 번째 Onset의 시작점부터 30초간 추출한 데이터 세트를 의미한다. SEDS는 자체적으로 수집한 데이터 세트로 10개의 장르(Classical, County, Disco, Hiphop, Jazz, Metal, Reggae, Rock, Pop, Blues)를 장르별로 100곡씩 총 1000곡의 가장 높은 에너지를 가진 30초의 구간을 탐색하여 추출한 데이터 세트를 의미한다. 이는 표 2와 같이 데이터셋을 구성한다.

(표 2) FEDS와 SEDS 의 장르별 데이터 개수
(Table 2) The number of data by genre in FEDS and SEDS

Genre	FEDS	SEDS
Classical	100	100
Country	100	100
Disco	100	100
Hiphop	100	100
Jazz	100	100
Metal	100	100
Reggae	100	100
Rock	100	100
Blues	100	100
Pop	100	100
Total	1,000	1,000

위와 같이 음악장르를 구분하기 위한 데이터 세트 구축 방안으로 Python의 Librosa 라이브러리를 이용한다. 음원의 파형, 주파수, 비트, 템포와 같이 각각의 음원이 가지는 음악 데이터 신호들을 추출하고, 음악 특성상 곡 전체를 모두 처리하여 특징을 추출하기에는 저장소의 용량, 방대한 연산량에 따른 시간 소모, 음악 내부적인 변화성 등의 문제가 있다. 따라서 수작업을 통하여 곡 일부분을 추출하여 사용하거나, 곡의 고정된 일정 부분을 사용하여야 한다[3]. 이에 본 연구는 FEDS와 SEDS 데이터 세트를 구축하기 위해 표 2의 장르들에 대하여 30초 이상 분량의 음원을 .wav파일 형식으로 수집하였다. 수집한 데이터 세트에서 장르 구분을 위한 30초 분량의 음악을 추출하기 위하여 박자의 첫 번째 비트 혹은 강조되는 리듬 변화가 발생하는 지점인 Onset을 기준으로 음악의 주파수를 분리·탐색하였다. Onset 지점을 검출하는 것이 곧 음악의 리듬을 분석하는 것이며 이를 통해 음악의 장르적 특징을 찾을 수 있다. 이 Onset을 기준으로 하여 첫 번째 Onset부터 30초를 추출하는 방식(FEDS)과 에너지레벨이 가장 높은 30초 구간을 탐색하여 추출하는 방식(SEDS)을 사용하여 데이터 세트를 구성하였다. 30초 추출한 음원 데이터는 22,050Hz의 샘플레이트로 데이터 세트를 구축하였다. 샘플레이트가 높을수록 음질은 향상 되지만 데이터의 크기가 커지면서 데이터 처리가 느려지는 현상이 발생하기 때문에 본 실험에서는 무거운 데이터 처리를 필요로 하는 44,100Hz 보다 22,050Hz 샘플레이트의 데이터 세트를 구축하였다.

3. 음악 데이터 특성 추출 방법 및 학습 모델

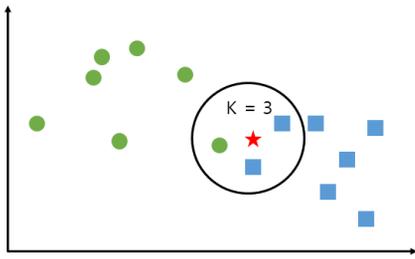
3.1 음악 데이터 특성 추출 방법

GTZAN에서 데이터 특성 추출 방법[5]과 같이 본 연구에서 생성한 2가지 데이터 세트(FEDS, SEDS)의 특성을 추출하였다. 이러한 특성은 표 3와 같다.

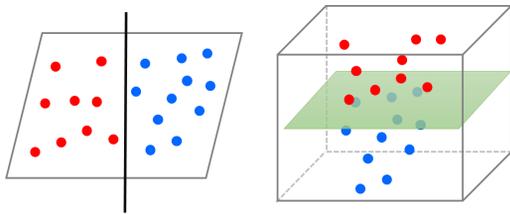
각각의 데이터 세트와 기존의 GTZAN 데이터 세트는 각 데이터 세트에 해당하는 범위에 Onset으로 부터 Chroma_STFT, RMS, Spectral_centroid, Spectral_bandwidth, Roll off, ZCR, Hamonym, Perceptr, Tempo, MFCC와 같은 주파수 표본 데이터를 추출한다. MFCC의 경우 Mel-Fiter를 통해 20개의 주파수 영역을 분할하였다. 추출한 데이터는 단일 값으로 표현되는 Tempo를 제외하면 전부 배열 형태의 데이터로, 데이터 간의 특성 개수가 달라 질 수 있으므로, 배열 형태의 결과를 갖는 Chroma_STFT, RMS, Spectral_centroid, Spectral_bandwidth, Rolloff, ZCR, Hamonym, Perceptr, MFCC 데이터는 모두 표본의 평균과 분산값을 사용한다.

(표 3) GTZAN 특성 목록
(Table 3) GTZAN Features List

Features	Description
Chroma_STFT	시간-주파수 특성을 표현하는 STFT (Short-Time-Fourier Transform)와 달리 음악적 화음 구조를 표현하는 특성
RMS(Root Mean Square)	RMS는 데이터의 변동성, 진폭을 통해 평균 에너지를 나타내며, 음악의 전체적인 에너지 크기를 표현하는 특성
Spectral_centroid	주파수의 진폭을 기중평균으로 계산, 밝고 높은 음색일수록 중심 주파수가 높아지는 특성
Spectral_bandwidth	주파수 스펙트럼의 분포의 넓이를 표현하는 특성
Rolloff	고주파수에서 0으로 감소하는 주파수를 표현하는 특성
ZCR (Zero Crossing Rate)	음성이나 음악신호의 파형의 변화를 추적하는 특성
Harmony	화음의 변화를 나타내는 특성
Perceptr	지각적 특성(perceptual features)의 변동률을 나타내는 특성
Tempo	주파수의 빠르기를 나타내는 특성
MFCC	주파수 스펙트럼을 20가지 영역으로 세분화하여 저주파~고주파 영역의 DCT (Discrete Cosin Transform) 변환을 적용한 특성



(그림 1) KNN 이웃 탐색 방법
(Figure 1) KNN Neighbor Search



(그림 2) SVM 차원 확장
(Figure 2) SVM Dimension Expansion

3.2 음악 장르 학습 방안

음악 장르 별 주파수 스펙트럼의 유사도를 통해 해당 음악이 어떤 장르에 가까운지를 판단하기 위하여 벡터 간의 거리 측정 방식을 사용한다. 이를 위해 KNN과 SVM을 학습에 사용한다. 그림 1과 같이 대표적인 거리 기반의 분류 모델인 KNN은 각 데이터 간의 2차원 상의 평면에서 새로운 데이터가 입력되면 해당 데이터의 위치를 기준으로 가장 가까운 K개의 데이터의 집합을 통해 데이터의 클래스를 판단한다. 그림 2와 같이 SVM은 KNN과 마찬가지로 거리 기반의 분류 모델이지만 KNN과 다르게 클래스 간의 균집을 기준으로 새로운 데이터의 클래스를 결정하며, 저차원을 고차원으로 투영시키는 방식으로 인해 비선형적 데이터 학습에 사용될 수 있다.

4. 데이터 전처리

거리 기반의 학습 모델인 KNN과 SVM은 데이터의 크기와 표준편차에 의해서 결과값이 영향을 크게 받는다는 특징이 있기 때문에 전처리 과정에서 모델에 적용할 해당 데이터 세트들은 정규화가 필요하다. 정규화 방식은 표 4에서와 같이 sklearn에서 제공하는 정규화 방법들을

비교하여 가장 원본 데이터의 특성을 유지 할 수 있는 방법을 선정한다.

(표 4) 스케일러
(Table 4) Scaler

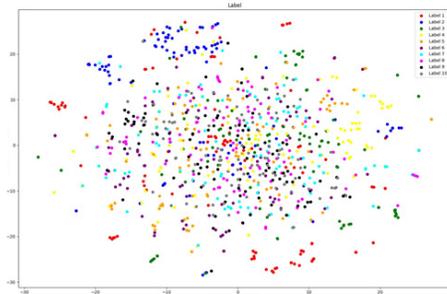
Scaler	Description
StandardScaler	평균 0, 분산 1인 정규분포를 갖도록 표준화, 이상치에 민감
MinMaxScaler	모든 특성이 0과 1사이의 값을 갖도록 표준화, 이상치에 민감
RobustScaler	이상치의 영향을 최소화, 중앙값과 IRQ(Interquartile Range)를 통해 표준화
Normalizer	데이터 벡터의 유클리디안 길이가 1이 되도록 위치를 조정, 방향이나 각도가 중요시 되는 데이터에서 사용

GTZAN, SEDS, FEDS 데이터 세트에 각각 4가지 스케일링 방식(StandardScaler, MinMaxScaler, RobustScaler, Normalizer)을 적용하여 KNN을 통해 정확도를 측정한 결과, 표 5과 같이 평균적으로 StandardScaler를 적용하였을 경우 가장 높은 정확도를 나타내었다.

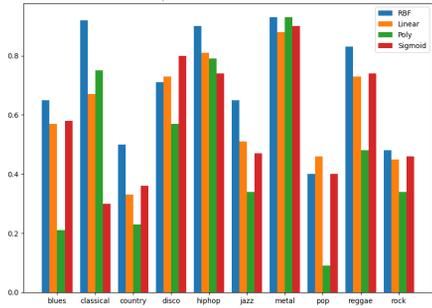
(표 5) 스케일러에 따른 평균 정확도 비교
(Table 5) Comparison of average accuracy according to scaler

Scaler	Result
StandardScaler	0.65
MinMaxScaler	0.61
RobustScaler	0.43
Normalizer	0.21

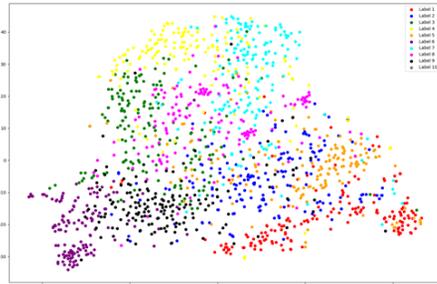
이러한 결과는 시각화를 통해 더 잘 나타낼 수 있는데 그림 3과 4와 같이 scaler를 적용하기 전과 후의 음악 장르별 데이터 분포를 보면 알 수 있다. StandardScaler를 적용하기 전의 음악 장르 데이터 분포는 그림 3과 같이 음악 장르가 뒤섞여 있는 것을 볼 수 있다. 반면에 그림 4와 같이 StandardScaler를 적용한 후의 음악 장르 데이터 분포는 일정한 영역별로 각 장르가 분포되어 있는 것을 볼 수 있다. 즉, StandardScaler 적용 전에 비해 각 음악 장르별로 음악 데이터가 밀집되어 있는 형태를 확인할 수 있다.



(그림 3) 정규화 전
(Figure 3) Before Normalization



(그림 5) SVM 커널 테스트 결과
(Figure 5) SVM Kernel Test Results



(그림 4) 정규화 후
(Figure 4) After Normalization

(표 6) SVM 커널 정확도
(Table 6) SVM Kernel Accuracy

SVM Kernel	Accuracy
RBF	71%
Linear	66%
Poly	64%
Sigmoid	54%

5. 연구 결과

5.1 모델 학습 결과

KNN에서 거리측정을 위한 3가지 방식인 Cosin, Manhattan, Minkowski를 이용하여 정확도를 측정 한 결과 Cosin은 56%, Mahanttan은 64%, Minkowski는 65%로 Minkowski가 가장 높은 정확도를 보여주었다. SVM에서 모든 파라미터에 대하여 디폴트(default) 파라미터를 사용하여 학습한 결과 정확도가 71%로 KNN의 65%에 비해 정확도가 6% 높았다. 따라서 SVM 방식이 음악 장르 분류에 좀 더 나은 것으로 판단하여 정확도 향상을 위한 추가 학습을 진행한다.

SVM에서 커널 조절을 통해 Linear, Poly, Sigmoid, RBF를 각각 비교하였을 경우, 그림 5와 같이 RBF가 pop과 disco 장르를 제외하고 가장 높은 정확도를 가진다. 또한, SVM 커널 변화에 따른 정확도는 표 6과 같다.

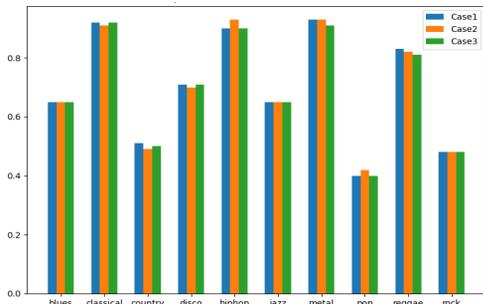
RBF 커널은 기본적으로 비선형 데이터들을 클래스별로 군집화하고 해당 군집들의 경계를 나누어 테스트 데이터가 속할 군집을 결정하는 알고리즘이기 때문에 각 군집의 결정경계인 서포트 벡터를 결정하는 기준이 중요하다. RBF의 결정경계를 결정하는 파라미터는 C와 Gamma로 C가 증가할수록 결정경계를 생성할 때의 제약 조건이 약해지고 Gamma가 증가할수록 군집의 결정경계가 영향력을 미치는 거리가 좁아진다. 이를 통해 적절한 C와 Gamma를 찾기 위하여 그리드서치를 통해 C의 비용을 각 0.001, 0.01, 0.1, 1, 10, 100으로 설정한 결과 1의 결과가 가장 높게 나왔다. 또한, Gamma값은 1/FeatureCount로 계산된 기본 값에 10% 간격으로 경사 하강법을 적용한 결과, 1% 이상의 정확도가 증가하는 Case가 없었기 때문에 1/FeatureCount의 수식 값을 그대로 사용하였다.

SVM은 일반적으로 고차원 데이터를 처리하기 적합한 모델이지만 데이터간의 거리로 인한 희소성 문제, 계산 비용 증가, 과적합 위험이 발생할 가능성이 높아짐으로 비선형 차원 축소 기법인 KernelPCA를 사용하여 정확도를 측정하였다. 그리드 서치(Grid search)를 통해 최적 차원 축소 후 정확도를 실험한 결과 가장 높은 정확도를 기록한 2가지 Case(Case 2, Case 3)와 PCA를 적용하지 않은 Case(Case 1)와 비교 결과는 표 7과 같다. 표에서 D는

Dimension을 의미한다. 차원 축소(PCA)를 진행하지 않고 RBF Kernel를 사용한 Case 1의 학습정확도는 84%이고 테스트 정확도는 71%이다. 차원 축소를 사용한 Case 2와 Case 3는 학습정확도가 89%, 84%로 5%높거나 같은 학습 정확도를 가지지만 여전히 테스트 결과는 71%, 70%로 별 다른 차이를 보이지 않는다. 따라서 해당 음악 데이터의 특성은 저차원 데이터로 축소하지 않아도 모델 정확도에 영향이 없음을 알 수 있다. 이에 대한 SEDS데이터 세트에 대한 각 클래스별 정확도를 비교해 보면 그림 6과 같다. 그림 6에서 Case별 클래스 간의 정확도 차이가 1~3%로 차원 축소를 하지 않아도 결과에 영향이 없음을 알 수 있다.

(표 7) SVM모델에서 PCA 커널에 따른 정확도 비교
(Table 7) Comparing accuracy in SVM models based on PCA kernels

Case	Kernel	PCA	D	Training Accuracy	Test Accuracy
Case1	RBF	X	57	84%	71%
Case2	RBF	Cosin	51	89%	71%
Case3	RBF	Linear	55	84%	70%



(그림 6) Case별 정확도 비교
(Figure 6) Coparing accuracy by case

5.2 데이터 세트별 테스트 결과

SVM모델에서 가장 성능이 좋은 모델의 Optimal Parameter는 다음 표 8과 같다. SVM의 경우 Linear, Poly, Sigmoid, RBF 커널중 RBF 커널의 정확도가 가장 높았기에 RBF를 사용하여 C 파라미터의 값은 0.001, 0.01, 0.1, 1, 10, 100 범위에서 그리드 서치를 통해 1을 사용하였고, G 파라미터는 기본 Gamma 계산 수식에서 사용되는 $1/FeatureCount$ 에서 음수와 양수 방향으로 각각 10회씩 $1/FeatureCount * (1/100)$ 만큼 조정하며 실험하였다. 또한 차원의 수가

(표 8) 최적 파라미터
(Table 8) Optimal Parameter

Parameter	Value
Model	SVM
Kernel	RBF
Gamma	Auto
C(Cost)	1
Dimension	57
BinaryClassifier	OvR or OvO
Train_Data_Count	800
Test_Data_Count	200
CV_Training_Accuracy	0.84
CV_Test_Accuracy	0.71

10개 이상으로 데이터 개수에 비해 많은 양의 차원으로 인해 차원 축소가 성능에 영향을 미치는지 실험하기 위하여 PCA 차원 축소 기법을 적용하였고, 차원 축소 기법의 경우 KPCA에서 사용할 수 있는 4가지 커널에 대하여 차원의 수 D 파라미터는 $\min(1) \sim \max(FeatureCount)$ 범위에서 그리드 서치를 통해 탐색하였다. 추가로 PCA 기법을 사용하지 않고 전통적인 상관관계 분석을 통해 rms_var, chroma_stft_var, perceptpr_var, harmony_var 피처를 제거하여 실험한 결과 전체 차원을 사용한 모델이 가장 높은 정확도를 보여주었다.

같은 데이터 세트와 모델을 사용하더라도 표본 추출 방식이나 데이터의 학습 비율에 따라 정확도의 차이가 발생 할 수 있다. 본 연구에서는 GTZAN 데이터 세트와 FEDS, SEDS 데이터 세트의 보다 신뢰성 높은 비교를 위해 각 데이터 세트마다 10-Fold Cross-validation을 적용하였다. 교차검증 결과는 표 9과 같이 GTZAN 데이터 세트는 평균 68%의 정확도를 보였으며, FEDS와 SEDS는 각각 평균 64%, 70%으로 정확도를 보여주었다. 교차검증 결과 동일한 음원에서 추출한 FEDS와 SEDS를 비교 했을 때 SEDS가 FEDS에 비해 정확도가 6% 높았으며, 이를 통해 음악 데이터 세트 생성시 에너지 레벨이 높은 구간을 추출하는 방안이 음악 장르별 특징을 분석함에 있어 더욱 신뢰성이 높다고 볼 수 있다.

GTZAN과 SEDS의 정확도는 2%의 차이를 보이지만 10개의 표본에 대한 표준편차가 GTZAN은 10%, SEDS는 3%로 큰 차이를 보이고 있다. 이는 앞선 설명한 GTZAN 데이터 세트의 결함으로 인한 결과로 판단된다. 따라서 음악 장르별 특징을 분석하기 위하여 SEDS 데이터 세트의 데이터 생성 방식을 사용하는 것이 효과적이다.

(표 9) GTZAN, FEDS, SEDS 비교 결과
(Table 9) GTZAN And FEDS And SEDS Comparison Results

	GTZAN	FEDS	SEDS
Fold-1	0.48	0.65	0.67
Fold-2	0.63	0.68	0.72
Fold-3	0.79	0.66	0.66
Fold-4	0.76	0.64	0.73
Fold-5	0.61	0.64	0.70
Fold-6	0.59	0.70	0.75
Fold-7	0.76	0.61	0.7
Fold-8	0.78	0.56	0.68
Fold-9	0.56	0.59	0.65
Fold-10	0.75	0.66	0.69
평균	0.68	0.64	0.7
표준편차	0.1	0.04	0.03

6. 결 론

본 논문에서는 GTZAN이 가지는 데이터 생성 문제점과 음원 학습시 음원별 데이터 학습을 위해 30초 단위로 새로운 FEDS와 SEDS 데이터 생성을 통해 기존 GTZAN 데이터와 분류 정확도를 비교 평가하였다. 57개의 음악 특성을 활용하여 머신러닝 방법인 KNN과 SVM 모델의 음악 분류 성능을 비교한 결과 SVM의 분류 정확도가 더 높았으며, SVM 모델의 학습 결과, 57개의 특성을 모두 사용하였을 경우 가장 좋은 성능을 보였다. GTZAN 데이터 세트는 학습 데이터와 테스트 데이터의 표본에 따라 정확도의 편차가 굉장히 높은 모습을 보인다. 특정 Fold에서 높은 정확도를 보여주기도 하지만 지나치게 낮은 정확도의 Fold 또한 존재하므로 학습 결과의 신뢰성을 보장할 수 없다. 이러한 GTZAN 데이터 세트의 신뢰성 및 중복, 라벨링 오류 문제를 해결하기 위해 음악 데이터로부터 Onset을 추출하여 평균 에너지 레벨이 가장 높은 구간을 추출하는 SEDS가 기존의 GTZAN 및 임의의 구간을 추출하는 방안에 비해 더 높은 성능과 신뢰성을 가짐을 검증하였다. 다만 분류 정확도가 71%로 다소 낮기 때문에 이를 위한 새로운 머신러닝 및 딥러닝 모델 학습 방법 및 음악 분류를 위한 음악의 분류 요소를 찾기 위한 방법을 적용하여 분류 정확도 높일 예정이다.

음악의 장르는 시간이 지날수록 더욱 다양해지고, 새로운 음악은 대부분의 사람이 쉽게 접할 수 있는 Web, Mobile 플랫폼을 통해 수많은 사람들에게 제공되고 있다. 그러나 대형 플랫폼과 달리 개인이 창작하거나 리믹스한

음악의 경우 분류나 구분이 명확하지 않은 경우가 대부분이라 음악을 이용하는 이용자들에게 쉽게 노출되지 않는 경우가 대부분이다. 머신러닝을 통한 음악 분류를 통해 개인이 업로드 하는 음악에 대한 접근성을 향상 시킬 수 있다면 소규모 플랫폼 및 음악 시장을 더 활성화 시키고, 이용자들에게 새로운 음악에 대한 접근성을 더욱 높일 수 있을 것이다.

참고문헌(Reference)

- [1] Jose M. Ausín, Enrique Bigne, Javier Marín, Jaime Guixeres and Mariano Alcañiz, "The background music-content congruence of TV advertisements: A neurophysiological study," *European Research on Management and Business Economics*, Vol.27, No.2, 2021. <https://doi.org/10.1016/j.iedeen.2021.100154>
- [2] I.D Choi, "The Effect of Background Music in TV-Commercials on Brand Image: Focused on Genre of Music," *The Korean Journal of Advertising and Public Relations*, Vol.7, No.2, pp.35-72. 2005. <https://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE02499488>
- [3] J.I Lee and B.M Kim, "Music Genre Classification based on Musical Features of Representative Segments," *Journal of KIISE:Software and Applications*, Vol.35, No.11, pp.692-700, 2008. <https://koreascience.kr/article/JAKO200800557077308.page>
- [4] Zhouyu Fu, Guojun Lu, Kai Ming Ting, and Dengsheng Zhang, "A survey of audio-based music classification and annotation," *IEEE Transactions on Multimedia*, Vol.13, No.2, pp.303-319, 2011. <https://doi.org/10.1109/TMM.2010.2098858>
- [5] W.J Jang, H.W Yun, S.H Shin, H.J Cho, W. Jang, and H.C Park, "Music Genre Classification using Spikegram and Deep Neural Network," *Journal of Broadcast Engineering*, Vol.22, No.6, pp. 693-701, 2017. <https://doi.org/10.5909/JBE.2017.22.6.693>
- [6] Hizlisoy, Serhat, Recep Sinan Arslan, and Emel Çolakoğlu, "Music Genre Recognition Based on Hybrid Feature Vector with Machine Learning Methods," *Çukurova Üniversitesi Mühendislik Fakültesi Dergisi*, Vol.38, No.3, pp.739-750, 2023. <https://doi.org/10.21605/cukurovaumfd.1377737>

- [7] Ghildiyal, Anirudh, Komal Singh, and Sachin Sharma, "Music genre classification using machine learning," in Proc. of 2020 4th international conference on electronics, communication and aerospace technology (ICECA), IEEE, 2020.
<https://doi.org/10.1109/ICECA49313.2020.9297444>
- [8] Tarimer, İlhan, and Buse Cennet Karadağ, "Genres Classification of Popular Songs Listening by Using Keras," Gazi University Journal of Science Part A: Engineering and Innovation, Vol.11, No.1, pp.123-136, 2024. <https://doi.org/10.54287/gujsa.1374878>
- [9] Ndou, Ndiatenda, Ritesh Ajoodha, and Ashwini Jadhav, "Music genre classification: A review of deep-learning and traditional machine-learning approaches," in Proc. of 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), IEEE, 2021.
<https://doi.org/10.1109/IEMTRONICSS2119.2021.9422487>
- [10] Fulzele, Prasenjeet et al., "A hybrid model for music genre classification using LSTM and SVM," in Proc. of 2018 eleventh international conference on contemporary computing (IC3), IEEE, 2018.
<https://doi.org/10.1109/IC3.2018.8530557>
- [11] S.H Kim, D.S Kim, and B.W Suh, "Music Genre Classification using Multimodal Deep Learning," PROCEEDINGS OF HCI KOREA 2016, pp.389-395, 2016. <https://doi.org/10.17210/hcik.2016.01.389>
- [12] Bob L. Sturm, "The State of the Art Ten Years After a State of the Art: Future Research in Music Information Retrieval," Journal of New Music Research, Vol.43, No.2, pp.147-172 2014.
<http://dx.doi.org/10.1080/09298215.2014.894533>

● 저 자 소 개 ●



엄 빛 찬(Bit-chan Eom)

2022년~현재 인하대학교 소프트웨어융합공학과(학사과정)
 관심분야 : 정보통신, AI
 E-mail : ubc445@naver.com



조 동 휘(Dong-Hwi Cho)

2021년~현재 인하대학교 소프트웨어융합공학과(학사과정)
 관심분야 : 머신러닝, 딥러닝, 영상처리
 E-mail : junejoe97@inha.edu



남 춘 성(Choon-Sung Nam)

2005년 상명대학교 소프트웨어학과(이학사)
 2007년 숭실대학교 대학원 컴퓨터학과(공학석사)
 2011년 성균관대학교 대학원 전자전기컴퓨터학과(공학박사)
 2020년~현재 인하대학교 소프트웨어융합공학과 조교수
 관심분야 : 무선 센서네트워크, 인터페이스 개발, 머신러닝, 딥러닝, 반도체 공정 etc.
 E-mail : namgun99@inha.ac.kr