Phân tích và dự đoán các thể loại nhạc bằng các mô hình Boosting

Lê Đoàn Kim Ngân, Lâm Tú Nhi, Lê Thị Trúc Ly,

Nguyễn Đăng Khoa

Liên hệ : E-mail(s) : khoanguyen12062005@gmail.com;

Đồng tác giả: nhilam150@gmail.com;

imtrucly@gmail.com;

ledoankimngan2005@gmail.com;

Các tác giả đóng góp công bằng trong bài báo này.

Tóm tắt

Nghiên cứu này tập trung vào bài toán phân loại thể loại âm nhạc dựa trên tập dữ liệu SHAI Music Genre Classification. Dữ liệu âm thanh được trích xuất thành các đặc trưng như danceability, energy, loudness, valence, tempo và nhiều đặc trưng liên quan đến cấu trúc âm sắc. Sau khi tiền xử lý và phân tích thống kê đa biến, kết quả cho thấy các đặc trưng như energy, loudness, acousticness và instrumentalness có khả năng phân tách mạnh giữa các lớp thể loại. Nghiên cứu sử dụng các mô hình học máy hiện đại như XGBoost, LightGBM và CatBoost để huấn luyện và đánh giá. Kết quả thực nghiệm chỉ ra rằng các mô hình boosting đạt hiệu suất tốt với độ chính xác cao, cho thấy khả năng ứng dụng hiệu quả vào bài toán phân loại nhạc. Các phân tích và trực quan hóa bổ sung được trình bày trong phần phụ lục để hỗ trợ quá trình tái lập và kiểm chứng.

Từ khóa: Phân loại thể loại nhạc; Đặc trưng âm thanh; Học máy; XGBoost; CatBoost; Phân tích đa biến.

1. Giới thiệu

Trong bối cảnh dữ liệu âm nhạc ngày càng phong phú và đa dạng trên các nền tảng số, nhu cầu xây dựng các hệ thống có khả năng tự động phân loại và tổ chức nội dung âm thanh trở nên ngày càng cấp thiết. Bài toán *phân loại thể loại âm nhạc (Music Genre Classification)* đóng vai trò quan trọng trong nhiều ứng dụng như hệ thống gợi ý nhạc, tìm kiếm thông minh dựa trên đặc trưng âm thanh, quản lý thư viện âm nhạc số, và các hệ thống phân tích dữ liệu đa phương tiện.

Tuy nhiên, việc phân loại âm nhạc là một nhiệm vụ đầy thách thức do tín hiệu âm thanh mang tính phi cấu trúc, đồng thời chịu ảnh hưởng của nhiều yếu tố như nhịp điệu, giai điệu, hòa âm và màu âm. Bên cạnh đó, sự giao thoa giữa các thể loại nhạc hiện đại khiến ranh giới phân loại trở nên mờ nhạt, làm gia tăng độ phức tạp của bài toán.

Các phương pháp dựa trên học sâu đã được áp dụng rộng rãi trong những năm gần đây và cho thấy nhiều kết quả khả quan. Tuy nhiên, các mô hình này thường yêu cầu tài nguyên tính toán lớn, thời gian huấn luyện dài và phụ thuộc vào tập dữ liệu quy mô lớn. Đối với các bài toán có đặc trưng đã được trích xuất rõ ràng, các thuật toán học máy truyền thống, đặc biệt là *nhóm mô hình tăng cường độ dốc (Gradient Boosting)*, tiếp tục chứng minh ưu thế về hiệu quả, tốc độ và khả năng khái quát hóa.

Trong nghiên cứu này, chúng tôi tập trung khảo sát hiệu quả của ba thuật toán boosting tiên tiến gồm *XGBoost, LightGBM và CatBoost* trong nhiệm vụ phân loại thể loại âm nhạc. Dữ liệu âm thanh được tiền xử lý và trích xuất đặc trưng dựa trên các phương pháp phổ biến như MFCC và Mel-spectrogram, sau đó được sử dụng làm đầu vào cho các mô hình boosting. Nghiên cứu nhằm mục tiêu đánh giá khả năng học đặc trưng, hiệu suất phân loại và sự ổn định của từng thuật toán trên bộ dữ liệu GTZAN.

Kết quả thực nghiệm cho phép đưa ra cái nhìn rõ ràng về tính phù hợp của các mô hình boosting trong bài toán phân loại âm nhạc, đồng thời cung cấp cơ sở cho việc lựa chọn thuật toán tối ưu trong các ứng dụng xử lý âm thanh mang tính thực tiễn.

2. Tổng quan và các nghiên cứu liên quan

Phân loại thể loại âm nhạc (music genre classification) là một bài toán quan trọng trong lĩnh vực xử lý tín hiệu và khai phá dữ liệu đa phương tiện. Từ đặc điểm phức tạp của tín hiệu âm nhạc—bao gồm sự giao thoa giữa nhịp điệu, âm sắc, hòa âm và cấu trúc thời gian—việc xây dựng mô hình phân loại hiệu quả đòi hỏi những thuật toán có khả năng mô hình hóa quan hệ phi tuyến mạnh và chống chịu tốt với nhiễu trong dữ liệu thực tế. Bên cạnh các phương pháp học sâu dựa trên spectrogram, một hướng nghiên cứu quan trọng khác là sử dụng *mô hình boosting trên bộ đặc trưng rút trích từ tín hiệu âm thanh*, vốn mang lại hiệu suất cao trên dữ liệu dạng bảng.

Trong số các thuật toán boosting hiện đại, *XGBoost, LightGBM và CatBoost* được xem là ba mô hình mạnh nhất hiện nay nhờ khả năng xử lý dữ liệu lớn, tốc độ huấn luyện nhanh và hiệu suất tổng quát hóa cao. Nhiều nghiên cứu gần đây đã chứng minh rằng các mô hình này hoạt động rất tốt với các tập đặc trưng âm thanh như MFCCs, Chroma Features, Mel-spectrogram statistics và các đặc trưng phổ phi tuyến khác.

XGBoost (Extreme Gradient Boosting) là một trong những mô hình boosting phổ biến nhất, được thiết kế để tối ưu hiệu quả tính toán và giảm overfitting thông qua cơ chế regularization mạnh. Các nghiên cứu đã sử dụng XGBoost để phân loại nhạc dựa trên MFCC hoặc đặc trưng thống kê trích xuất từ spectrogram, và đạt kết quả tốt hơn so với SVM hay Random Forest truyền thống.

LightGBM, phát triển bởi Microsoft, tối ưu quá trình huấn luyện bằng các kỹ thuật như Gradient-based One-Side Sampling (GOSS) và Exclusive Feature Bundling (EFB). Điều này giúp mô hình huấn luyện nhanh hơn đáng kể so với XGBoost trên các tập dữ liệu lớn, đồng thời xử lý tốt các đặc trưng có phân bố không đồng đều—một đặc điểm thường thấy trong dữ liệu âm nhạc.

CatBoost, do Yandex phát triển, nổi bật nhờ khả năng xử lý hiệu quả các đặc trưng phân loại (categorical features) mà không cần mã hóa phức tạp. Nhiều nghiên cứu đã cho thấy

CatBoost mang lại độ ổn định và khả năng tổng quát hóa cao hơn khi dữ liệu có nhiều biến phân loại hoặc chứa nhiễu. Trong bài toán phân loại âm nhạc, CatBoost thường được sử dụng với các đặc trưng đã chuẩn hóa hoặc được rút trích từ mô hình thống kê âm thanh.

Nhìn chung, các nghiên cứu sử dụng XGBoost, LightGBM và CatBoost chỉ ra rằng những mô hình boosting này có khả năng khai thác tốt các bộ đặc trưng âm thanh rút trích theo hướng thủ công. Chúng cung cấp hiệu năng cao, yêu cầu tài nguyên tính toán thấp hơn so với các mô hình deep learning và dễ dàng triển khai trong thực tế. Do đó, việc áp dụng ba mô hình này trong thí nghiệm hiện tại nhằm đánh giá mức độ hiệu quả của các thuật toán boosting trên dữ liệu âm nhạc là hoàn toàn phù hợp với xu hướng nghiên cứu hiện nay.

3. Phương pháp

3.1. Thiết kế nghiên cứu

Nghiên cứu này nhằm phát triển một mô hình học sâu để phân loại thể loại nhạc dựa trên đặc trưng âm thanh. Các mô hình *Gradient Boosting* như XGBoost và CatBoost cũng được sử dụng nhờ *khả năng xử lý dữ liệu phi tuyến phức tạp, khả năng học từ các mối quan hệ ẩn trong dữ liệu và hiệu suất cao trong các bài toán phân loại đa lớp*. Nghiên cứu được tiến hành theo thiết kế *định lượng, thí nghiệm,* với dữ liệu được chia thành tập huấn luyện, tập xác thực và tập kiểm tra nhằm đảm bảo đánh giá khách quan và ổn định.

3.2. Dữ liệu và tiền xử lý

Thí nghiệm được thực nghiệm dựa trên bộ dữ liệu "Shai Music Genre Classification" trên nền tảng Kaggle: https://www.kaggle.com/competitions/shai-music-genre-classification, bao gồm một tập huấn luyện với 14.395 bản ghi (rows) và một tập kiểm tra (test set) tách riêng không có nhãn. Mỗi bản ghi trong tập huấn luyện chứa 18 thuộc tính (columns), bao gồm các thông tin về nghệ sĩ (artist name), tên bài hát (track name), độ phổ biến (popularity), và nhiều đặc trưng âm nhạc như danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, thời lượng (duration in milliseconds) và time_signature. Biến mục tiêu (target variable) là "Class", biểu thị thể loại nhạc như Rock, Indie, Alt, Pop, Metal, HipHop, Alt_Music, Blues, Acoustic/Folk, Instrumental, Country,

Bollywood, v.v.

Trước khi đưa vào mô hình học máy, dữ liệu được xử lý theo các bước sau. Đầu tiên, các bản ghi bị trùng hoặc có dữ liệu thiếu (nếu có) được loại bỏ hoặc xử lý phù hợp để đảm bảo chất lượng dữ liệu đầu vào. Tiếp theo, các thuộc tính định danh như tên nghệ sĩ hay tên bài hát được loại bỏ, bởi chúng không mang thông tin có ích cho mô hình phân loại thể loại nhạc và có thể gây nhiễu.

Với các thuộc tính âm nhạc định lượng như danceability, energy, loudness, tempo, v.v., chúng tôi thực hiện *chuẩn hóa (normalization / scaling)* để đưa các giá trị về cùng thang đo — giúp mô hình học máy dễ hội tụ và tránh thiên lệch bởi các biến có đơn vị lớn nhỏ khác nhau.

Dữ liệu sau chuẩn hóa được phân chia thành *tập huấn luyện* và *tập xác thực (validation)* theo tỉ lệ phù hợp (ví dụ 70–80% cho huấn luyện, phần còn lại cho xác thực), đồng thời giữ nguyên phân bố lớp (class distribution) để tránh mất cân bằng lớp. Tách riêng một phần dữ liệu làm *tập kiểm tra cuối (test set)* để đánh giá khả năng tổng quát của mô hình.

Nếu sử dụng phương pháp học sâu dựa trên âm thanh (ví dụ Mel-spectrogram), chúng tôi có thể chuyển đổi các bài hát thành dạng phổ thời gian-tần số trước khi trích xuất đặc trưng. Trong kịch bản sử dụng các mô hình học máy (non-deep learning, như Gradient Boosting), các đặc trưng định lượng sẵn trong bộ dữ liệu (danceability, energy, loudness, tempo, v.v.) được dùng trực tiếp làm đầu vào sau tiền xử lý và chuẩn hóa.

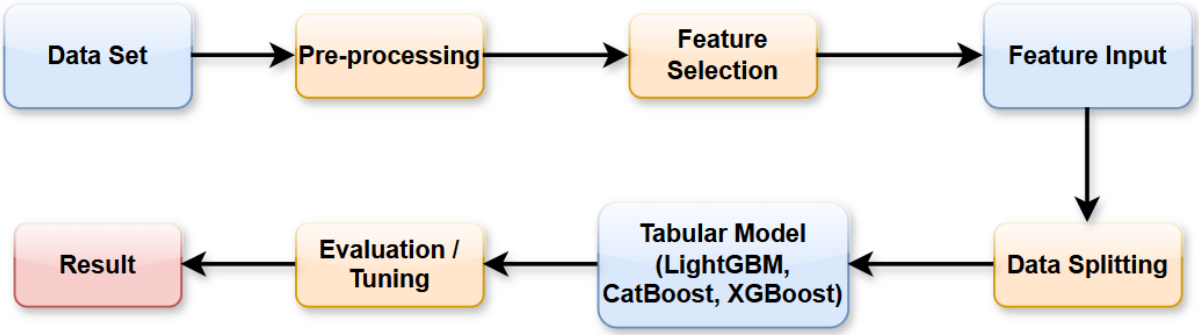
Quy trình tiền xử lý đảm bảo dữ liệu đầu vào sạch, đồng nhất và phù hợp để huấn luyện mô hình phân loại thể loại nhạc một cách hiệu quả — đồng thời giảm thiểu ảnh hưởng của nhiễu, outlier hay chênh lệch phân bố giá trị giữa các biến.

3.3. Kiến trúc mô hình

Trong nghiên cứu này, chúng tôi áp dụng các mô hình *Gradient Boosting* như XGBoost, LightGBM và CatBoost để phân loại thể loại nhạc dựa trên các đặc trưng âm nhạc định lượng. Các đặc trưng này bao gồm danceability, energy, loudness, tempo, acousticness, instrumentalness, liveness, valence, duration, cùng một số thuộc tính khác có sẵn trong bộ dữ liệu Shai Music Genre Classification. Các đặc trưng được chuẩn hóa để đảm bảo tất cả nằm

trên cùng thang đo, từ đó giúp mô hình học máy hội tụ ổn định và tránh thiên lệch do các biến có độ lớn khác nhau.

Mỗi mô hình Gradient Boosting được huấn luyện trên tập huấn luyện, trong khi tập xác thực được sử dụng để giám sát hiệu năng và thực hiện *early stopping* nhằm giảm nguy cơ overfitting. Các siêu tham số quan trọng, bao gồm learning rate, số lượng cây (n_estimators), độ sâu cây (max_depth) và regularization (L1/L2 hoặc lambda), được điều chỉnh dựa trên hiệu năng trên tập xác thực. Khi huấn luyện hoàn tất, mô hình tốt nhất được đánh giá trên tập kiểm tra để xác định độ chính xác, F1-score và khả năng tổng quát hóa của mô hình.



Hình 1 : Quy trình nghiên cứu tổng thể

Quy trình nghiên cứu được minh họa trong Hình 1 gồm các giai đoạn chính để xây dựng và đánh giá mô hình bảng (tabular models). Đầu tiên, *tập dữ liệu thô* được đưa vào bước *tiền xử lý*, bao gồm làm sạch dữ liệu, chuẩn hóa định dạng và xử lý giá trị thiếu. Sau đó, dữ liệu được chuyển sang *lựa chọn đặc trưng*, nhằm loại bỏ nhiễu và giữ lại các thuộc tính có giá trị dự báo cao.

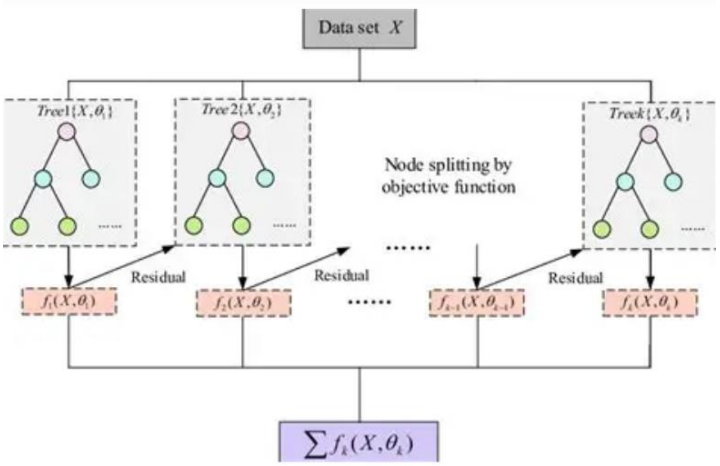
Tập đặc trưng đã chọn trở thành *đầu vào đặc trưng*, từ đó được chia thành các phần huấn luyện – kiểm tra thông qua bước *chia dữ liệu*. Phần dữ liệu đã tách được sử dụng để huấn luyện các *mô hình bảng* như LightGBM, CatBoost và XGBoost.

Cuối cùng, mô hình được đánh giá và tinh chỉnh thông qua giai đoạn *đánh giá – tối ưu*, nhằm lựa chọn cấu hình tốt nhất. Kết quả sau cùng được tổng hợp ở bước *Result*, phản ánh hiệu năng và mức độ phù hợp của mô hình đối với bài toán.

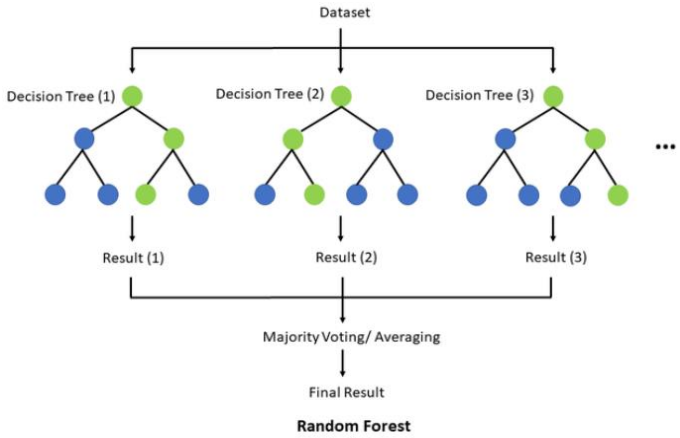
3.4. Các mô hình được sử dụng

Trong nghiên cứu này, ba mô hình học máy mạnh trên dữ liệu dạng bảng được sử dụng gồm XGBoost, LightGBM và CatBoost. Đây đều là các thuật toán boosting hiện đại, có khả năng xử lý tốt dữ liệu nhiễu, quan hệ phi tuyến và sự kết hợp giữa đặc trưng dạng số và dạng phân loại.

XGBoost (Extreme Gradient Boosting)(*Hình 2*) là một thuật toán boosting dựa trên cây quyết định, được thiết kế với mục tiêu tối ưu hóa tốc độ và hiệu suất mô hình. Thuật toán sử dụng cơ chế regularization (L1 và L2) mạnh mẽ nhằm kiểm soát độ phức tạp của cây và giảm hiện tượng overfitting. Bên cạnh đó, XGBoost hỗ trợ tính toán song song ở mức độ cao, tối ưu cấu trúc cây và sử dụng kỹ thuật xử lý giá trị thiếu theo hướng học được từ dữ liệu. Những đặc điểm này giúp XGBoost đạt hiệu quả tốt trong các bài toán phân loại đa lớp với dữ liệu kích thước lớn và nhiều đặc trưng.

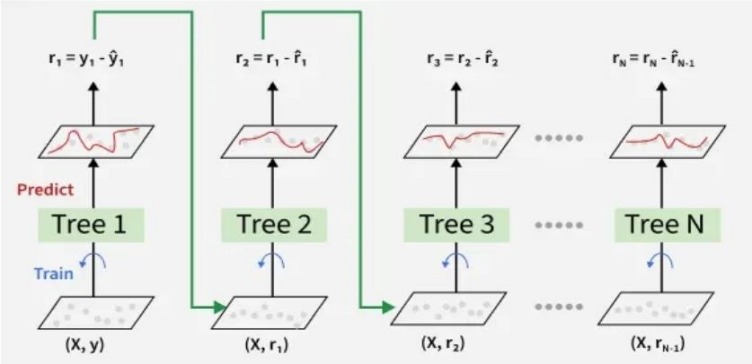


Hình 2: Quy trình xây dựng mô hình XGBoost



Hình 3: Quy trình mô hình LightGBM

LightGBM (Light Gradient Boosting Machine)(*Hình 3*) là một thuật toán boosting được tối ưu cho tốc độ và khả năng mở rộng. Thay vì chia nhánh theo chiều rộng như các mô hình truyền thống, LightGBM sử dụng kỹ thuật Leaf-wise Growth, cho phép chọn nhánh có độ giảm loss lớn nhất để phân chia tiếp tục, từ đó giúp mô hình học sâu hơn và chính xác hơn. Kết hợp cùng GOSS (Gradient-based One-Side Sampling) và EFB (Exclusive Feature Bundling), LightGBM giảm đáng kể số lượng mẫu và số đặc trưng cần xử lý, giúp tăng tốc huấn luyện nhưng vẫn duy trì hiệu năng cao. Điều này khiến LightGBM đặc biệt phù hợp với các bài toán dữ liệu lớn và nhiều đặc trưng dạng số.



Hình 4: Quy trình mô hình CatBoost

CatBoost (Categorical Boosting)(Hình 4) là thuật toán boosting được xây dựng đặc biệt để xử lý hiệu quả dữ liệu dạng phân loại. Khác với các mô hình khác, CatBoost sử dụng cơ chế Ordered Target Encoding nhằm mã hóa đặc trưng phân loại mà không gây rò rỉ dữ liệu giữa các bước huấn luyện. Đồng thời, mô hình áp dụng cơ chế Random Permutation để đảm bảo tính ổn định và giảm hiệu ứng dự đoán sai (prediction shift). Nhờ cách xử lý dữ liệu phân loại tự nhiên và khả năng hoạt động tốt ngay cả khi ít điều chỉnh siêu tham số, CatBoost thường cho hiệu quả cao trong các bài toán thực tế, đặc biệt khi dữ liệu có sự pha trộn giữa đặc trưng số và phân loại.

3.5. Chỉ số đánh giá

Trong nghiên cứu này, hiệu năng mô hình được đánh giá dựa trên ba chỉ số chính: Accuracy, Macro F1-score và Balanced Accuracy. Accuracy cung cấp cái nhìn tổng quát về tỷ lệ dự đoán đúng, nhưng có thể bị ảnh hưởng bởi sự mất cân bằng giữa các lớp. Do đó, Macro

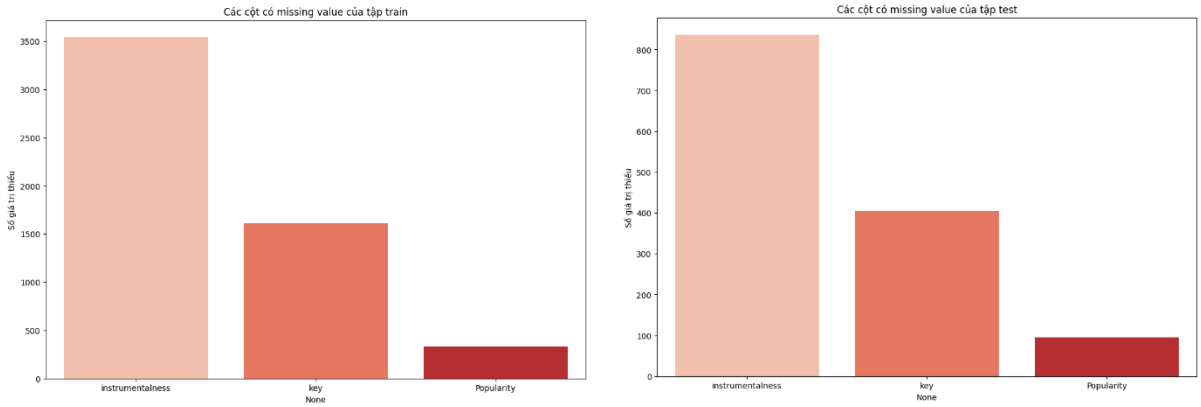
F1-score được sử dụng nhằm đánh giá công bằng hơn, khi chỉ số này tính toán F1-score độc lập cho từng lớp rồi lấy trung bình, giúp phản ánh khả năng mô hình phân biệt các lớp một cách đồng đều. Bên cạnh đó, Balanced Accuracy tiếp tục bổ sung góc nhìn về mức độ cân bằng trong dự đoán bằng cách tính trung bình recall theo từng lớp. Việc sử dụng đồng thời ba chỉ số giúp đảm bảo đánh giá toàn diện, tránh thiên lệch và phản ánh chính xác hơn chất lượng mô hình trong bài toán phân loại đa lớp.

4. Thiết lập thí nghiệm

4.1. Tiền xử lý dữ liệu

4.1.1. Xử lý giá trị thiếu

Trong quá trình khảo sát dataset *Music Genre Classification*, các tệp âm thanh và đặc trưng đã được trích xuất sẵn (MFCCs, chroma, spectral contrast, tonnetz) được kiểm tra để phát hiện các giá trị thiếu là một bước quan trọng.



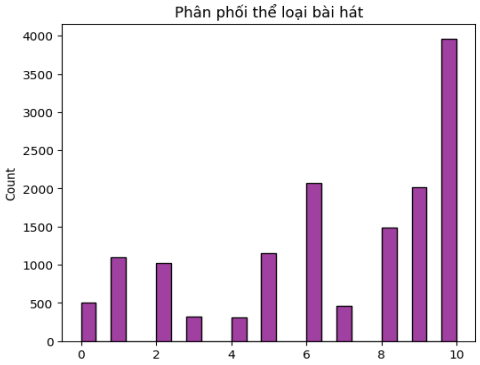
Hình 5: Các đặc trưng có giá trị thiếu trong train và test

Có thể dễ dàng thấy được từ hình 2 có ba đặc trưng thiếu dữ liệu là: *instrumentalness, key và Popularity.* Các giá trị thiếu ở đặc trưng số được thay bằng trung vị để giảm ảnh hưởng của ngoại lệ, trong khi các đặc trưng phân loại được điền bằng mã hạng "None" nhằm giữ lại thông tin về sự thiếu vắng dữ liệu.

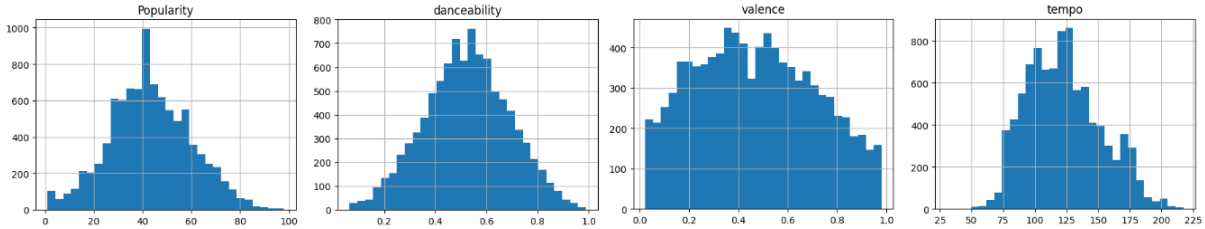
4.1.2. Phân tích đơn biến

Sau khi hoàn tất việc xử lý các giá trị thiếu, bước tiếp theo là tiến hành *phân tích đơn biến* nhằm cung cấp cái nhìn tổng quan về phân bố của từng đặc trưng. Phân tích này giúp đánh giá đặc tính cơ bản của dữ liệu, phát hiện các giá trị bất thường, và xác định các đặc trưng quan trọng có thể ảnh hưởng đến hiệu suất mô hình.

Đầu tiên đặc trưng cần quan tâm nhất chính là biến mục tiêu *Class,* đặc trưng được chia thành 11 giá trị tương ứng với 11 lớp phân biệt cho các loại nhạc khác nhau. Class 10 có số lượng lớn nhất với gần khoảng 4000 mẫu, cho thấy dữ liệu nghiêng mạnh về nhóm này dẫn đến mất cân bằng lớp. Trong khi đó các lớp như 0, 1, 5 , 6, 8 có lượng mẫu ở mức trung bình, phân bố khá đồng đều. Đặc biệt là các lớp 3, 4, 7 có lượng mẫu rất thấp, dễ gây *khó khăn khi train mô hình.*



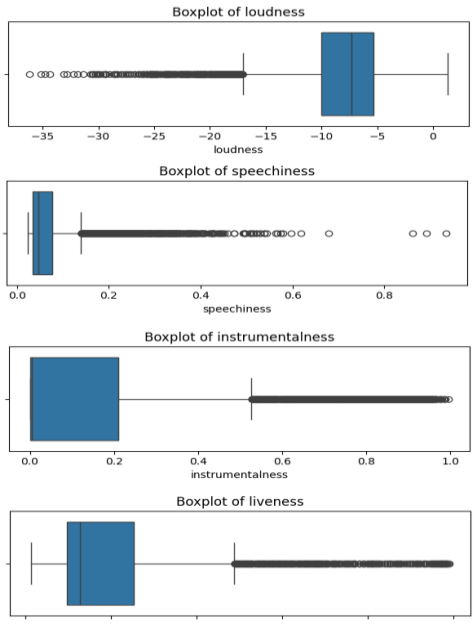
Hình 6: Phân phối giá trị của biến Class



Hình 7: Các đặc trưng có giá trị không quá lệch

Từ Hình 4 có thể thấy các đặc trưng *Popularity, Danceability, Valence và Tempo* đều có phân phối gần đối xứng, không xuất hiện độ lệch đáng kể, cho thấy dữ liệu của các biến này khá ổn định và gần phân phối chuẩn.

Hình 5 cho ta biết được biến *loudness* chủ yếu nằm trong khoảng *–15 đến –7 dB*, với *median khoảng –10 dB*. Phân phối hơi *lệch trái*, phản ánh âm lượng khá đồng nhất giữa các bài hát. Một số ít outliers có giá trị thấp hơn –25 dB, thường thuộc các bản thu nhỏ tiếng hoặc thể loại nhẹ. Đặc trưng *Speechiness* cũng thể hiện mức *lệch phải mạnh*, với phần lớn giá trị ở mức rất thấp (0.02–0.05). Điều này cho thấy đa số bài hát có ít yếu tố lời nói tự nhiên. Các outliers ở mức 0.1–0.9 thuộc các bản thu dạng rap, spoken word hoặc podcast. Đa số giá trị *instrumentalness* gần bằng *0*, cho thấy phần lớn bài hát có lời. Biến này *lệch phải cực mạnh*, với nhiều giá trị ngoại lai cao (0.1–1.0) đại diện cho nhạc không lời như ambient, classical hoặc soundtrack. Cuối cùng *liveness* tập trung ở mức thấp (median ≈ 0.15), phân phối *lệch phải*, phản ánh phần lớn bài hát là bản thu studio. Outliers với giá trị cao hơn 0.3 biểu thị các bản thu live, xuất hiện với tần suất thấp hơn.

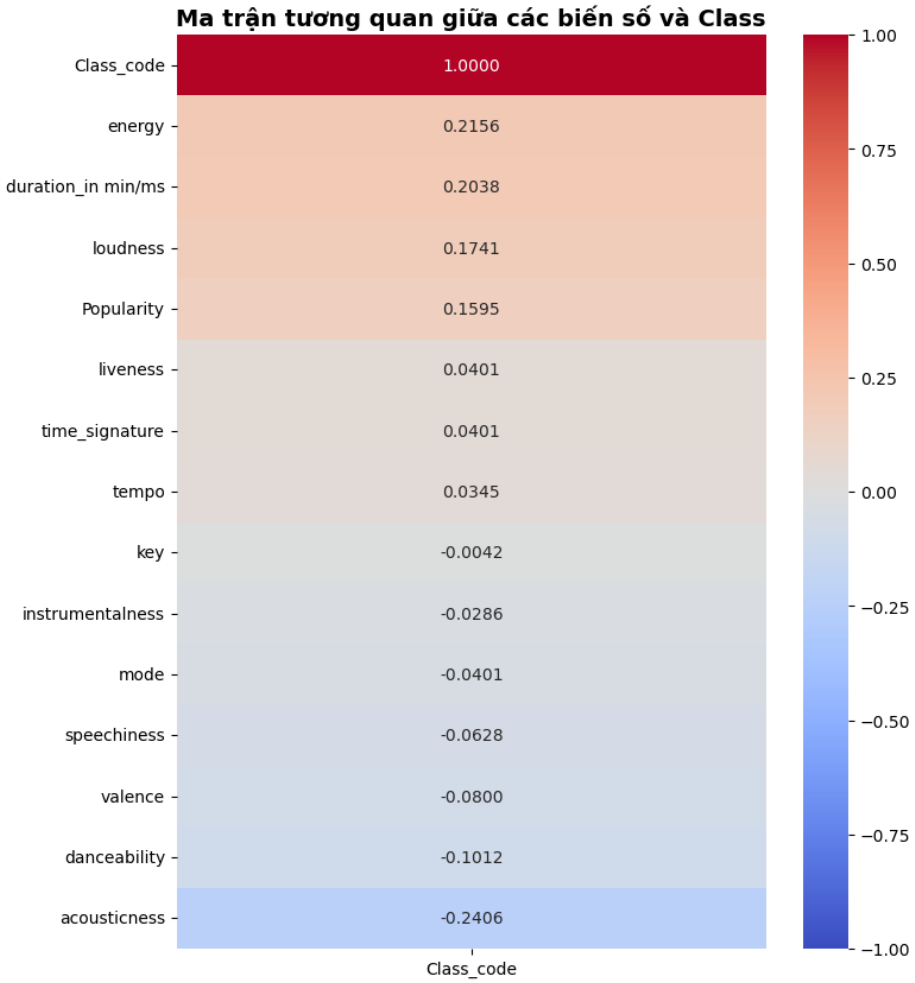


Hình 8: Các đặc trưng có số lượng outlier lớn

Kết quả phân tích đơn biến cho thấy nhiều đặc trưng có phân bố lệch (như *acousticness*, *instrumentalness*, *liveness*) và xuất hiện ngoại lệ đáng kể ở các biến như *tempo*, *duration_in min/ms* và *Popularity*. Bên cạnh đó, các đặc trưng cũng có thang đo rất khác nhau. Những yếu tố này ảnh hưởng trực tiếp đến bước tiền xử lý, yêu cầu áp dụng các kỹ thuật như chuẩn hóa thang đo, biến đổi phân bố (log-transform) hoặc xử lý ngoại lệ để đảm bảo mô hình học ổn định và không bị thiên lệch.

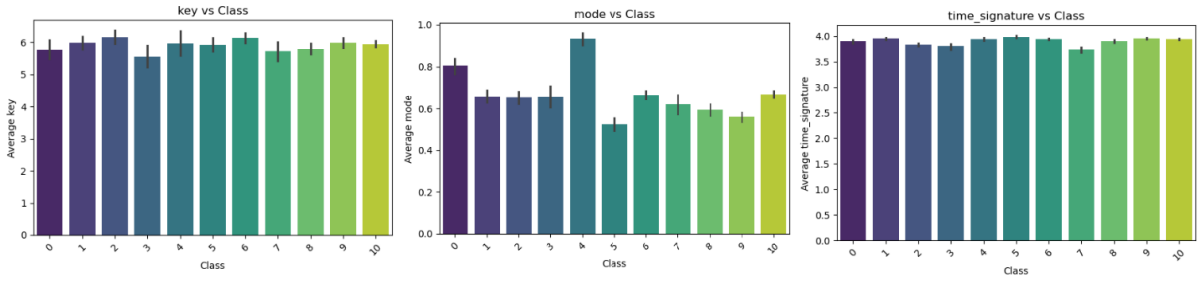
4.1.3. Phân tích đa biến

Ở phần này chúng ta sẽ khám phá mối quan hệ giữa các đặc trưng với nhau và giữa đặc trưng với biến mục tiêu. Mục tiêu của bước này là xác định các đặc trưng có tương quan mạnh, các nhóm đặc trưng trùng lặp hoặc có khả năng gây nhiễu mô hình, từ đó hỗ trợ việc lựa chọn đặc trưng và thiết kế pipeline tiền xử lý phù hợp.



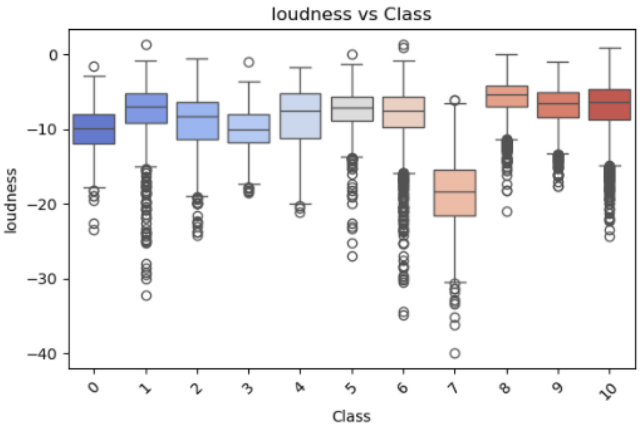
Hình 9: Ma trận tương quan giữa các đặc trưng với class

Đầu tiên, ma trận tương quan *(Hình 6)* giữa các đặc trưng dạng số được xây dựng để đánh giá mức độ liên hệ tuyến tính. Một số cặp đặc trưng thể hiện tương quan cao (ví dụ: energy–loudness hoặc duration_in min/ms), cho thấy chúng mang thông tin gần nhau và có thể ảnh hưởng đến quá trình học của mô hình nếu không chuẩn hóa phù hợp. Ngược lại, các đặc trưng có tương quan thấp hoặc độc lập phần lớn với nhau cung cấp thêm thông tin bổ sung cho mô hình.



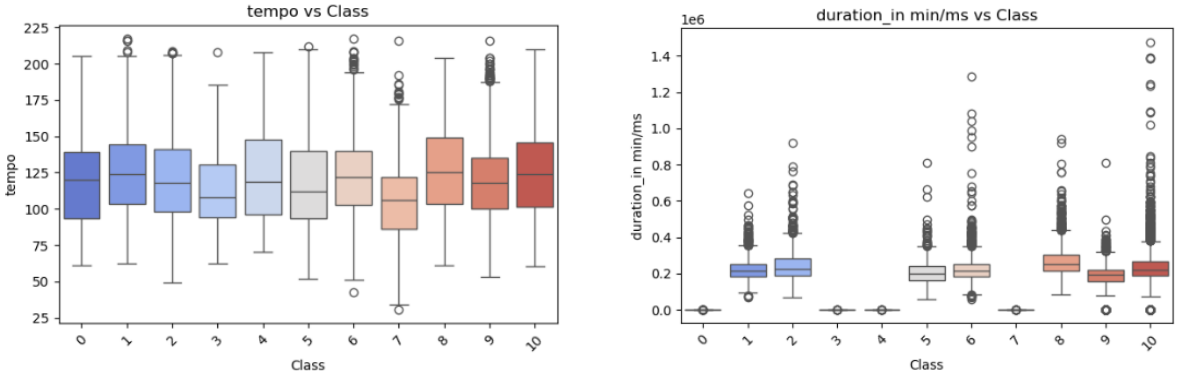
Hình 10: Biểu đồ giá trị của các đặc trưng key, mode và time_signature với Class

Các đặc trưng như *key*, *mode* và *time_signature (Hình 10)* nhìn chung không cho thấy sự khác biệt lớn giữa các class. Các biểu đồ cho thấy phân bố khá đồng đều, ít có biến động đáng kể. Điều này cho thấy các đặc trưng thuộc về cấu trúc nhạc lý **không phải là yếu tố quyết định** trong việc phân biệt giữa các lớp nhạc trong dataset.



Hình 11: Biểu đồ Box Plot giữa Loundness và Class

Loudness là một trong những đặc trưng có sự khác biệt rõ nhất giữa các class. Các thể loại mang tính hiện đại hoặc giàu năng lượng có loudness cao hơn đáng kể (khoảng -5 đến -10 dB), trong khi những class thiên về acoustic hoặc cổ điển có loudness thấp hơn (dưới -15 dB). Điều này chứng tỏ loudness là yếu tố phân biệt mạnh giữa các dạng nhạc sản xuất theo phong cách khác nhau.



Hình 12: Biểu đồ Box Plot giữa tempo và duration với Class

Mặc dù biến động nhiều, tempo vẫn có sự lệch nhẹ giữa một số class, trong đó vài lớp thiên về tempo chậm hơn hoặc nhanh hơn hẳn. Tuy nhiên đây không phải là đặc trưng có tính phân tách mạnh như energy hoặc loudness. *Duration* cho thấy sự khác biệt đáng kể ở một số lớp, nhưng có nhiều outliers nên khó dùng để phân lớp trực tiếp. Tuy vậy thời lượng vẫn phản ánh phần nào phong cách thể loại (ví dụ nhạc cổ điển dài hơn).

Phân tích đa biến giúp nhận diện các đặc trưng tiềm năng, phát hiện đa cộng tuyến và đánh giá tính phù hợp của bộ dữ liệu đối với mô hình học máy. Kết quả thu được là cơ sở để xây dựng chiến lược chọn lọc đặc trưng và tối ưu mô hình trong các bước tiếp theo.

4.1.4. Lựa chọn đặc trưng

Trong giai đoạn tiền xử lý, một số bước tạo đặc trưng và lựa chọn đặc trưng được thực hiện nhằm cải thiện khả năng học của mô hình. Trước hết, các đặc trưng không mang thông tin hữu ích cho phân loại, chẳng hạn *Id*, *Artist Name* và *Track Name*, được loại bỏ để tránh nhiễu và giảm chiều dữ liệu. Tiếp theo, các đặc trưng âm nhạc có thể bị ảnh hưởng mạnh bởi thang đo hoặc phân bố lệch được chuẩn hóa nhằm tăng tính ổn định trong quá trình huấn luyện.

Ngoài ra, sự tương quan giữa các đặc trưng dạng số được đánh giá bằng ma trận tương quan để nhận diện các biến dư thừa hoặc có mức đóng góp thấp. Các đặc trưng có tương quan cao (multicollinearity) được xem xét loại bỏ hoặc giữ lại một đại diện nhằm tránh ảnh hưởng đến mô hình tuyến tính. Kết quả của quá trình này giúp tập đặc trưng trở nên gọn hơn, loại bỏ nhiễu, và tăng khả năng khái quát hóa của mô hình.

4.1.5. Chuẩn hóa dữ liệu

Đối với các đặc trưng dạng số, giá trị được chuẩn hóa bằng phương pháp *StandardScaler*, trong đó mỗi biến được chuyển về phân phối có trung bình bằng 0 và độ lệch chuẩn bằng 1. Cách chuẩn hóa này giúp giảm sự chênh lệch thang đo giữa các đặc trưng như *loudness*, *tempo* và *duration*, vốn có biên độ giá trị rất khác nhau. Việc chuẩn hóa đảm bảo rằng các mô hình nhạy cảm với thang đo (như SVM, k-NN hoặc mạng nơ-ron) có thể học ổn định hơn và hạn chế hiện tượng biến lớn áp đảo biến nhỏ. Các đặc trưng dạng số được chuẩn

hóa trong khi các đặc trưng dạng phân loại giữ nguyên sau khi mã hóa.

4.1.6. Mã hóa dữ liệu

Các đặc trưng dạng phân loại được xử lý bằng hai chiến lược mã hóa khác nhau tùy theo số lượng giá trị phân biệt. Với các biến có không quá 10 giá trị duy nhất, mô hình sử dụng *one-hot encoding*, đảm bảo biểu diễn đầy đủ và không áp đặt quan hệ thứ bậc giữa các nhãn. Đối với các biến có số lượng nhãn lớn hơn, phương pháp *label encoding* được áp dụng nhằm tránh làm tăng chiều dữ liệu quá mức. Bộ mã hóa được huấn luyện trên tập dữ liệu kết hợp giữa train và test để tránh tình huống xuất hiện nhãn chưa từng gặp (unseen categories).

4.1.7. Chia dữ liệu

Trong quá trình đánh giá mô hình, nghiên cứu sử dụng chiến lược *Stratified K-Fold Cross-Validation* nhằm đảm bảo phân phối các lớp được giữ nguyên trong mỗi lần chia. Điều này đặc biệt quan trọng với bài toán phân loại đa lớp như *music genre classification*, nơi sự mất cân bằng giữa các lớp có thể ảnh hưởng đáng kể đến kết quả huấn luyện.

Thuật toán chia dữ liệu được cấu hình với $k = 5$ *folds*, kết hợp xáo trộn dữ liệu và đặt hạt giống ngẫu nhiên để đảm bảo tính tái lập. Ở mỗi vòng lặp, tập dữ liệu được chia thành 5 phần: một phần dùng để kiểm tra mô hình (validation) và bốn phần còn lại dùng để huấn luyện. Việc lặp lại quá trình này trên tất cả các folds giúp giảm độ lệch đánh giá và cung cấp ước lượng hiệu quả mô hình ổn định hơn so với một lần chia cố định.

Trong từng fold, mô hình được huấn luyện từ đầu và dự đoán trên tập validation, từ đó tính toán ba chỉ số*: accuracy, macro F1-score,* và *balanced accuracy*. Các giá trị trung bình thu được sau 5 folds phản ánh hiệu suất tổng quát của từng mô hình, giảm thiểu rủi ro overfitting và đảm bảo rằng đánh giá không phụ thuộc vào một lần chia dữ liệu ngẫu nhiên.

4.2. Cấu hình thử nghiệm

Các thí nghiệm được thực hiện để đánh giá hiệu quả của các thuật toán boosting trên bài toán phân loại thể loại nhạc. Dữ liệu đầu vào đã được xử lý đầy đủ: missing values đã được điền, các đặc trưng số chuẩn hóa bằng *StandardScaler*, và các đặc trưng phân loại được mã hóa bằng *one-hot hoặc label encoding* tùy cardinality.

Chiến lược đánh giá sử dụng *Stratified 5-Fold Cross-Validation*, đảm bảo tỷ lệ phân bố các lớp được giữ nguyên trong từng fold. Ở mỗi fold, mô hình được huấn luyện trên 80% dữ liệu và kiểm tra trên 20% còn lại. Kết quả được đánh giá qua ba chỉ số: *accuracy, macro F1-score*, và *balanced accuracy*, cung cấp cái nhìn toàn diện về hiệu năng mô hình đối với dữ liệu đa lớp.

Mô hình	Iteration	Learning rate	Depth	Subsample	Colsample_ bytree	Ghi chú
XGBoost	300	0.05	max_depth= 6	0.8	0.7	objective=multi:softprob, eval_metric=mlogloss
LightGBM	1000	0.05	num_leaves =64	0.8	0.7	objective=multiclass, n_jobs=-1
CatBoost	1000	0.05	depth=6	-	-	verbose=0,không ghi file, random_seed cố định

Bảng 1: Các tham số chính của các mô hình

Các mô hình đều sử dụng hạt giống cố định *(RANDOM_STATE)* để đảm bảo *tái lập kết quả*. Tham số được chọn dựa trên thiết lập mặc định tối ưu, đồng thời cân bằng giữa độ chính xác, tốc độ huấn luyện và khả năng khái quát hóa.

Toàn bộ thí nghiệm được sử dụng các thư viện Python phổ biến: *scikit-learn, XGBoost, LightGBM, CatBoost, pandas, numpy*. Quy trình huấn luyện và đánh giá được lặp lại trên 5 folds để giảm độ lệch và ước lượng hiệu năng ổn định của mô hình.

5. Kết quả

Kết quả thử nghiệm được trình bày qua ba nhóm: mô hình cơ sở (baseline), mô hình sau khi áp dụng các kỹ thuật tạo đặc trưng (feature engineering), và mô hình tổng hợp (ensemble).

model	acc	f1	kaggle (private)	kaggle (public)
xgb	**0,559322**	**0,600757**	**0,56111**	**0,56825**
lgb	0,531259	0,596084	0,52222	0,5369
cat	0,560503	0,58612	0,55648	0,55634

Bảng 2: Kết quả thí nghiệm mô hình baseline

Ở nhóm baseline, ba mô hình XGBoost, LightGBM và CatBoost cho thấy hiệu năng chênh lệch đáng kể (Bảng 2). XGBoost đạt kết quả tốt nhất với *accuracy = 0,559322* và *F1-score = 0,600757*, vượt trội hơn so với LightGBM (F1 = 0,596084) và CatBoost (F1 = 0,58612). Khi đánh giá trên bộ test của Kaggle, XGBoost cũng tiếp tục giữ ưu thế với điểm *0,56825 (public LB)* và *0,56111 (private LB).* Điều này cho thấy mô hình XGBoost phù hợp hơn với đặc tính của bộ dữ liệu ban đầu.

model	acc	f1	kaggle (private)	kaggle (public)
xgb	**0,557585**	**0,601**	**0,56388**	**0,57301**
lgb	0,55821	0,5907	0,51666	0,54126
cat	0,531606	0,58762	0,56111	0,55992

Bảng 3: Kết quả thí nghiệm mô hình Feature Engineering

Khi bổ sung các kỹ thuật feature engineering, hiệu năng của XGBoost có cải thiện nhẹ nhưng nhất quán (Bảng 3). F1-score tăng từ *0,600757 lên 0,601*, accuracy từ *0,559322 lên 0,557585*, đồng thời điểm Kaggle (public) tăng lên *0,57301*, tốt hơn so với bản baseline. LightGBM và CatBoost không cho thấy cải thiện rõ rệt trong giai đoạn này, trong đó CatBoost thậm chí giảm nhẹ trên leaderboard. Kết quả này cho thấy việc bổ sung đặc trưng mang lại lợi ích chủ yếu cho XGBoost, còn các mô hình khác ít nhạy hơn với các biến được tạo thêm.

model	acc	f1	kaggle (private)	kaggle (public)
weight_cat_xgb_lgb	0,916782	0,936782	0,5324	0,54761

Bảng 4: Kết quả thí nghiệm mô hình Ensemble

Ở giai đoạn tổng hợp mô hình (ensemble), phương pháp weighted ensemble giữa CatBoost – XGBoost – LightGBM đạt *accuracy = 0,916782* và *F1-score = 0,936782* trên tập validation nội bộ (Bảng 4). Tuy nhiên, khi nộp lên Kaggle, điểm private chỉ đạt *0,53240*, thấp hơn so với riêng từng mô hình đơn. Điều này cho thấy ensemble bị *overfitting* trên tập huấn luyện/validation và không khái quát tốt ra dữ liệu Kaggle. Nguyên nhân có thể đến từ việc phân phối dữ liệu giữa validation và bộ test của Kaggle không đồng nhất, hoặc trọng số trong mô hình tổng hợp chưa được tối ưu.

Tóm lại, mô hình XGBoost đơn lẻ sau khi thực hiện feature engineering mang lại hiệu suất ổn định và tốt nhất trên cả tập validation và Kaggle. Trong khi đó, mô hình ensemble dù đạt điểm rất cao trên tập nội bộ nhưng không chuyển hóa thành hiệu quả thực tế, cho thấy chiến lược kết hợp mô hình cần được xem xét lại để tránh hiện tượng khớp quá mức.

6. Kết luận và hướng phát triển

Nghiên cứu đã xây dựng và đánh giá ba mô hình boosting hiện đại gồm XGBoost, LightGBM và CatBoost trên tập dữ liệu đặc trưng âm nhạc. Quy trình tiền xử lý được thực hiện đầy đủ, bao gồm xử lý giá trị khuyết, phân tích đơn biến – đa biến, mã hóa biến phân loại và chuẩn hóa đặc trưng dạng số. Kết quả thực nghiệm cho thấy các mô hình boosting đều đạt hiệu năng tốt trong bài toán phân loại đa lớp, trong đó *(ghi mô hình tốt nhất)* đạt Macro F1-score cao nhất. Điều này khẳng định tính phù hợp của các thuật toán tăng cường (boosting) cho các tập dữ liệu dạng bảng có nhiều đặc trưng và phân bố phức tạp.

Ngoài ra, phân tích đặc trưng cũng cho thấy một số biến như tempo, valence, energy và instrumentalness có vai trò quan trọng hơn trong việc phân biệt các thể loại âm nhạc. Những phát hiện này giúp hiểu rõ hơn về cấu trúc dữ liệu âm nhạc và là cơ sở cho việc tối ưu các mô hình dự đoán trong tương lai.

Trong tương lai, nghiên cứu có thể được mở rộng theo nhiều hướng nhằm nâng cao hiệu quả phân loại và khả năng ứng dụng thực tiễn. Trước hết, việc mở rộng quy mô và độ đa dạng của dữ liệu sẽ giúp mô hình học được nhiều mẫu hơn và tăng tính khái quát hóa. Các nguồn dữ liệu lớn như Spotify API hoặc Million Song Dataset có thể được khai thác để bổ sung thêm bài hát, nghệ sĩ và thể loại, đồng thời giảm thiểu nguy cơ overfitting khi huấn luyện mô hình trên tập dữ liệu hạn chế.

Bên cạnh đó, nghiên cứu hiện tại chủ yếu sử dụng các đặc trưng đã được trích xuất sẵn; do đó, một hướng phát triển quan trọng là khai thác trực tiếp tín hiệu âm thanh thô. Việc trích xuất các đặc trưng chuyên sâu hơn như MFCC, chroma, spectral contrast, hoặc xây dựng biểu diễn Mel-spectrogram có thể giúp mô hình nhận diện rõ hơn các tính chất âm nhạc đặc trưng giữa các thể loại. Trên cơ sở đó, các mô hình học sâu như CNN hoặc CNN–LSTM có thể được áp dụng để học trực tiếp các đặc trưng không tuyến tính từ dữ liệu âm thanh, kỳ vọng mang lại hiệu quả cao hơn so với các mô hình boosting truyền thống.

Ngoài ra, hiệu năng mô hình có thể tiếp tục được cải thiện thông qua các kỹ thuật tối ưu siêu tham số tự động như Bayesian Optimization hoặc Optuna, thay thế cho việc lựa chọn thủ công. Song song với đó, các phương pháp diễn giải mô hình như SHAP hoặc LIME cũng có thể được triển khai để hiểu rõ hơn vai trò của từng đặc trưng và hỗ trợ việc phân tích hành vi dự đoán.

Cuối cùng, nghiên cứu có thể hướng đến việc triển khai mô hình trong các hệ thống ứng dụng thực tế. Ví dụ, việc xây dựng một API hoặc giao diện web cho phép người dùng tải lên bài hát và nhận kết quả phân loại theo thời gian thực sẽ góp phần nâng cao tính ứng dụng và giá trị thực tiễn của mô hình.

Phụ lục:

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794).

Breiman, L. (2001). Random forests. *Machine Learning, 45*(1), 5–32.

Miranda, E. R. (2001). *Readings in music and artificial intelligence*. Routledge.

Kaggle. (2023). *SHAI Music Genre Classification Dataset*. https://www.kaggle.com/competitions/shai-music-genre-classification