TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

ĐỒ ÁN

**TỐT NGHIỆP ĐẠI HỌC**

NGÀNH CÔNG NGHỆ THÔNG TIN

**PHÂN LỚP MỘT SỐ LÀN ĐIỆU CHÈO VÀ DÂN CA QUAN HỌ**

Sinh viên thực hiện: **Lại Văn Hải**

Lớp: IS2 – K58

Giáo viên hướng dẫn: PGS.TS **Trịnh Văn Loan**

HÀ NỘI 05-2018

PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

1. Thông tin về sinh viên

Họ và tên sinh viên: Lại Văn Hải

Điện thoại liên lạc: 0166 455 2105 Email: hailv.hedspi@gmail.com

Lớp: IS2 Hệ đào tạo: Đại học chính quy

Đồ án tốt nghiệp được thực hiện tại: trường đại học Bách Khoa Hà Nội

Thời gian làm ĐATN: Từ ngày 01/01/2018 đến 28/05/2018

2. Mục đích nội dung của ĐATN

Xây dựng hệ thống phân lớp một số làn điệu chèo và dân ca quan họ sử dụng mô hình GMM thông qua qua hai bộ công cụ ALIZE và SPro.

3. Các nhiệm vụ cụ thể của ĐATN

- Tìm hiểu tổng quan về phân lớp.

- Tìm hiểu một số mô hình phân lớp thường dùng.

- Tìm hiểu ứng dụng của mô hình GMM trong phân lớp làn điệu chèo và dân ca quan họ.

- Tìm hiểu về bộ công cụ ALIZE, SPro.

- Sử dụng bộ công cụ ALIZE và SPro trong phân lớp làn điệu chèo và dân ca quan họ.

4. Lời cam đoan của sinh viên:

Tôi Lại Văn Hải cam kết ĐATN là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của PGS.TS.Trịnh Văn Loan

Các kết quả nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 28 tháng 05 năm 2018*  Tác giả ĐATN  Lại Văn Hải |

5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của ĐATN và cho phép bảo vệ:

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 28 tháng 05 năm 2018*  Giáo viên hướng dẫn  PGS.TS. Trịnh Văn Loan |

LỜI CẢM ƠN

Để có được đồ án tốt nghiệp như ngày hôm nay nếu chỉ nhờ sự cố gắng của riêng bản thân em thì không thể nào có thể hoàn thành được. Chính vì vậy, lời đầu tiên em xin bày tỏ lòng biết ơn sâu sắc tới PGS.TS. Trịnh Văn Loan – Bộ môn Kỹ thuật Máy tính – Viện Công nghệ Thông tin và Truyền thông – Đại học Bách khoa Hà Nội, thầy đã tận tình dìu dắt em qua ba kỳ đồ án, xuất phát điểm chỉ với nền tảng duy nhất là sự hứng thú với đề tài nghiên cứu, do đó để bổ sung những mảng kiến thức còn thiếu này thầy đã tạo điều kiện cho em và nhóm đồ án được tham gia môn học mà trong chương trình đào tạo của chúng em không có. Qua mỗi buổi học không chỉ có kiến thức chuyên ngành mà bên cạnh đó còn có những câu chuyện của thầy giúp em được định hướng rất nhiều điều cả trong học tập cũng như cuộc sống.

Em xin chân thành cảm ơn thầy Chu Bá Thành đã nhiệt tình chỉ dạy những phần kiến thức vô cùng quan trọng liên quan đến đề tài của em, nhờ có sự hướng dẫn tận tình của thầy mà rất nhiều vấn đề khúc mắc của em đều đã được giải đáp.

Cuối cùng em xin gừi lời cảm ơn đến người thân và những bạn bè xung quanh em đã luôn ở bên ủng hộ và khích lệ tinh thần cho em vào những lúc khó khăn nhất.

Hà Nội, ngày 28 tháng 05 năm 2018

Tác giả ĐATN

Lại Văn Hải

TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP

Hiện nay sự bùng nổ về công nghệ thông tin đã giúp cho các dịch vụ trực tuyến phát triển mạnh mẽ, một trong số đó có thể kể tới dịch vụ nghe nhạc trực tuyến. Tuy nhiên, cùng với sự phát triển này là sự bùng nổ về dữ liệu dẫn tới việc quản lý, phân loại và truy xuất dữ liệu đòi hỏi phải có các công cụ hữu ích hỗ trợ cho công việc này chứ không chỉ đơn thuần là các thao tác thủ công. Xuất phát từ nhu cầu thực tiễn đó, việc tạo ra các công cụ hỗ trợ cho việc phân lớp là vô cùng cần thiết.

Trên thế giới đã có nhiều công trình nghiên cứu về phân loại các thể loại âm nhạc hiện đại như Pop, Rock, Ballad,…Tuy nhiên, bên cạnh đó mỗi quốc gia hay dân tộc lại có cho riêng mình một kho tàng đồ sộ những thể loại âm nhạc khác, ví dụ tại Việt Nam còn có các thể loại như dân ca quan họ, chèo, cải lương… Các làn điệu trong dân ca quan họ và chèo rất phong phú nhưng trong khuôn khổ của đồ án tốt nghiệp này em chỉ xin trình bày về “**Phân lớp một số làn điệu chèo và dân ca quan họ**”.

Mục đích của đồ án tốt nghiệp là xây dựng hệ thống phân lớp một số làn điều chèo và dân ca quan họ dựa trên mô hình GMM, để triển khai được hệ thống này thì trong đồ án em xin trình bày những nội dung như sau:

* Tìm hiểu những khái niệm cơ sở liên quan đến phân lớp.
* Nghiên cứu cách thức hoạt động của một số mô hình phân lớp thường dùng như mô hình K-láng giềng gần nhất, mô hình cây quyết định, mô hình Markov ẩn và mô hình hỗn hợp Gauss.
* Hệ thống thử nghiệm được xây dựng thông qua hai bộ công cụ ALIZE và SPro, do đó trong đồ án sẽ giới thiệu về hai bộ công cụ này cũng như cách sử dụng chúng trong việc xây dựng hệ thống phân lớp một số làn điệu chèo và dân ca quan họ.

Bằng việc triển khai hệ thử nghiệm với bộ dữ liệu 1000 file âm thành bao gồm 25 làn điệu chèo và 25 làn điệu quan họ, sử dụng cơ cở lý thuyết được đề cập ở phần trên kết hợp với ngôn ngữ lập trình Python chạy trên môi trường Linux, đồ án đã thu được một số kết quả thực nghiệm như sau:

* Độ chính xác của hệ thống phân lớp sơ bộ (xác định một làn điệu là làn điệu chèo hay quan họ) đạt được kết quả tốt nhất là với số thành phần Gauss là 4096.
* Độ chính xác của hệ thống phân lớp cụ thể (xác định cụ thể tên của làn điệu), đối với làn điệu chèo kết quả cao nhất thu được là với số thành phần Gauss là 512. Đối với làn điệu quan họ kết quả cao nhất thu được là với số thành phần Gauss là 1024.

MỤC LỤC

[LỜI CẢM ƠN ii](#_Toc515211611)

[TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP iii](#_Toc515211612)

[MỤC LỤC iv](#_Toc515211613)

[DANH MỤC KÝ HIỆU, CHỮ VIẾT TẮT vii](#_Toc515211614)

[DANH MỤC HÌNH viii](#_Toc515211615)

[DANH MỤC BẢNG ix](#_Toc515211616)

[MỞ ĐẦU 1](#_Toc515211617)

[Lý do chọn đề tài 1](#_Toc515211618)

[Mục đích, phạm vi nghiên cứu 1](#_Toc515211619)

[Đối tượng nghiên cứu 1](#_Toc515211620)

[Phương pháp nghiên cứu 2](#_Toc515211621)

[Nhiệm vụ nghiên cứu 2](#_Toc515211622)

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT PHÂN LỚP 3](#_Toc515211623)

[1.1. Tổng quan về phân lớp 3](#_Toc515211624)

[1.1.1. Khái niệm 3](#_Toc515211625)

[2.1.1. Các bước để xây dựng một hệ thống phân lớp 3](#_Toc515211626)

[1.2. Một số mô hình phân lớp thường dùng 4](#_Toc515211627)

[1.2.1. Mô hình K-láng giềng gần nhất (K-NN) 4](#_Toc515211628)

[1.2.2. Mô hình cây quyết định (Decision Tree) 5](#_Toc515211629)

[a. Khái niệm 5](#_Toc515211630)

[b. Xây dựng cây quyết định 6](#_Toc515211631)

[1.2.3. Mô hình Markov ẩn (HMM) 7](#_Toc515211632)

[a. Mô hình Markov 7](#_Toc515211633)

[b. Mô hình Markov ẩn 8](#_Toc515211634)

[c. Ứng dụng của HMM 10](#_Toc515211635)

[1.2.4. Mô hình hỗn hợp Gauss 10](#_Toc515211636)

[a. Đặc tả mô hình 10](#_Toc515211637)

[b. Bài toán ước lượng mật độ 12](#_Toc515211638)

[1.3. Phân lớp làn điệu chèo và dân ca quan họ 14](#_Toc515211639)

[1.3.1. Đặc trưng của làn điệu chèo và dân ca quan họ 14](#_Toc515211640)

[a. Đặc trưng của làn điệu chèo 14](#_Toc515211641)

[b. Đặc trưng của làn điệu dân ca quan họ 14](#_Toc515211642)

[1.3.2. Phân lớp làn điệu chèo và dân ca quan họ là gì? 15](#_Toc515211643)

[1.3.3. Các ứng dụng của phân lớp làn điệu 15](#_Toc515211644)

[1.3.4. Phương pháp phân lớp làn điệu chèo và dân ca quan họ 15](#_Toc515211645)

[1.4. Tìm hiểu một số công trình phân lớp âm nhạc đã có tại Việt Nam 17](#_Toc515211646)

[1.4.1. Phân loại nhạc theo thể loại dùng phép biến đổi Wavelet rời rạc, tác giả: Phan Anh Cang, Phan Thượng Cang.[4] 17](#_Toc515211648)

[a. Cơ sở dữ liệu 17](#_Toc515211649)

[b. Phương pháp nhận dạng 17](#_Toc515211650)

[c. Kết quả nhận dạng 18](#_Toc515211651)

[CHƯƠNG 2. CÁC CÔNG CỤ SỬ DỤNG TRONG ĐỂ TÀI 19](#_Toc515211652)

[2.1. Bộ công cụ SPro 19](#_Toc515211654)

[2.2. Bộ công cụ ALIZE 19](#_Toc515211655)

[2.2.1. Nguồn gốc 19](#_Toc515211656)

[2.1.2. Giới thiệu về ALIZE 19](#_Toc515211657)

[2.3. Sử dụng SPro và ALIZE trong phân lớp làn điệu 20](#_Toc515211658)

[2.3.1. Trích chọn đặc trưng 21](#_Toc515211659)

[2.3.2. Loại bỏ khoảng lặng 22](#_Toc515211660)

[2.3.3. Chuẩn hóa đặc trưng 24](#_Toc515211661)

[2.3.4. Huấn luyện mô hình nền 24](#_Toc515211662)

[2.3.5. Huấn luyện mô hình đích 25](#_Toc515211663)

[2.3.6. Thử nghiệm 26](#_Toc515211664)

[CHƯƠNG 3. TRIỂN KHAI HỆ THỐNG THỬ NGHIỆM PHÂN LỚP LÀN ĐIỆU CHÈO VÀ DÂN CA QUAN HỌ 28](#_Toc515211665)

[3.1. Sơ đồ tổng quan quá trình xây dựng hệ thống phân lớp làn điệu chèo và dân ca quan họ 28](#_Toc515211667)

[3.1.1. Tiến hành thử nghiệm với dữ liệu không nằm trong tập dữ liệu huấn luyện 28](#_Toc515211668)

[3.1.2. Tiến hành thử nghiệm với dữ liệu lấy từ tập huấn luyện 28](#_Toc515211670)

[3.2. Chuẩn bị dữ liệu 29](#_Toc515211671)

[3.3. Phân lớp làn điệu sử dụng ALIZE 32](#_Toc515211672)

[3.3.1. Bước 1: Tạo thư mục làm việc 32](#_Toc515211673)

[3.3.2. Bước 2: Tính tham số MFCC 33](#_Toc515211674)

[3.3.3. Bước 3: Dò tìm năng lượng 33](#_Toc515211675)

[3.3.4. Bước 4: Phát hiện âm thanh trong tín hiệu 33](#_Toc515211676)

[3.3.5. Bước 5: Chuẩn hóa các tham số của tín hiệu 34](#_Toc515211677)

[3.3.6. Bước 6: Tạo sanh sách file wav 34](#_Toc515211678)

[3.3.7. Bước 7: Phân chia dữ liệu thành hai tập huấn luyện và thử nghiệm 35](#_Toc515211679)

[3.3.8. Bước 8: Tạo file World.lst và World.lenght 35](#_Toc515211680)

[3.3.9. Bước 9: Chuẩn hóa TrainWorldInit 35](#_Toc515211681)

[3.3.10. Bước 10: Chuẩn hóa TrainWorldFinal 36](#_Toc515211682)

[3.3.11. Bước 11: Tạo file chứa danh sách các file được huấn luyện 36](#_Toc515211683)

[3.3.12. Bước 12: Huấn luyện GMM 36](#_Toc515211684)

[3.3.13. Bước 13: Tạo file thử nghiệm 37](#_Toc515211685)

[a. Phân lớp sơ bộ 37](#_Toc515211686)

[b. Phân lớp cụ thể 37](#_Toc515211687)

[3.3.14. Bước 14: Tiến hành thử nghiệm 37](#_Toc515211688)

[CHƯƠNG 4. KẾT QUẢ THỬ NGHIỆM 39](#_Toc515211689)

[4.1. Kết quả thử nghiệm với dữ liệu không nằm trong tập huấn luyện 39](#_Toc515211690)

[4.1.1. Kết quả thử nghiệm phân lớp sơ bộ 39](#_Toc515211691)

[4.1.2. Kết quả thử nghiệm phân lớp cụ thể 41](#_Toc515211692)

[a. Kết quả phân lớp làn điệu chèo 41](#_Toc515211693)

[b. Kết quả phân lớp làn điệu dân ca quan họ 43](#_Toc515211694)

[4.2. Kết quả thử nghiệm với dữ liệu lấy từ tập huấn luyện 46](#_Toc515211695)

[4.2.1. Kết quả thử nghiệm phân lớp sơ bộ 46](#_Toc515211696)

[4.2.2. Kết quả thử nghiệm phân lớp cụ thể 48](#_Toc515211697)

[a. Kết quả phân lớp làn điệu chèo 48](#_Toc515211698)

[b. Kết quả phân lớp làn điệu dân ca quan họ 50](#_Toc515211699)

[CHƯƠNG 5. KẾT LUẬN 53](#_Toc515211700)

[5.1. Các kết quả đạt được 53](#_Toc515211702)

[5.2. Các vấn đề chưa đạt được 53](#_Toc515211703)

[5.3. Hướng phát triển của đề tài 53](#_Toc515211704)

[TÀI LIỆU THAM KHẢO 54](#_Toc515211705)

[PHỤ LỤC 55](#_Toc515211706)

[A. Kết quả thử nghiệm với dữ liệu nằm không nằm trong tập huấn luyện. 55](#_Toc515211707)

[A.1. Kết quả thử nghiệm phân lớp sơ bộ 55](#_Toc515211708)

[A.2. Kết quả thử nghiệm phân lớp cụ thể làn điệu chèo 56](#_Toc515211709)

[A.3. Kết quả thử nghiệm phân lớp cụ thể làn điệu dân ca quan họ 60](#_Toc515211710)

[B. Kết quả thử nghiệm với dữ liệu lấy từ tập huấn luyện 64](#_Toc515211711)

[B.1. Kết quả thử nghiệm phân lớp sơ bộ 64](#_Toc515211712)

[B.2. Kết quả thử nghiệm phân lớp cụ thể làn điệu chèo 66](#_Toc515211713)

[B.3. Kết quả thử nghiệm phân lớp cụ thể làn điệu dân ca quan họ 70](#_Toc515211714)

DANH MỤC KÝ HIỆU, CHỮ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| ***Chữ viết tắt*** | ***Chữ viết đầy đủ*** | ***Ý nghĩa*** |
| **EM** | Expectance Maximization | Cực đại hóa kỳ vọng |
| **FFT** | Fast Fourier Transform | Biến đổi Fourier nhanh |
| **GMM** | Gaussion Mixture Model | Mô hình hỗn hợp Gauss |
| **HMM** | Hidden Markov Model | Mô hình Markov ẩn |
| **K-NN** | K-Nearest Neighbors | K-láng giềng gần nhất |
| **LPC** | Linear Predictive Coding | Mã hóa dự đoán tuyến tính |
| **LPGL** | Lesser General Public License | Giấy phép phần mềm tự do |
| **MAP** | Maximum A Posteriori | Cực đại xác suất hậu nghiệm |
| **MFCC** | Mel-Frequency Cepstral Coefficients | Các hệ số Cepstral theo thang đo tần số Mel |
| **ML** | Machine Learning | Học máy |
| **UBM** | Universal Background Model | Huấn luyện mô hình nền |
| **UML** | Unified Modeling Language | Ngôn ngữ mô hình hóa thống nhất |

DANH MỤC HÌNH

[Hình 1.1. Mô tả việc triển khai thuật toán K-NN 4](#_Toc515211715)

[Hình 1.2. Mô tả bài toán phân lớp sử dụng cây quyết định 6](#_Toc515211716)

[Hình 1.3. Mô hình Markov bậc 1 8](#_Toc515211717)

[Hình 1.4. Mô hình Markov bậc 2 8](#_Toc515211718)

[Hình 1.5. Mô hình Markov trong bài toán dự báo thời tiết 9](#_Toc515211719)

[Hình 1.6. Mô hình Markov ẩn 3 trạng thái 9](#_Toc515211720)

[Hình 1.7. Hàm mật độ Gauss 11](#_Toc515211721)

[Hình 1.8. Mô hình GMM 11](#_Toc515211722)

[Hình 1.9. Hàm mật độ của GMM có 3 phân phối Gauss 12](#_Toc515211723)

[Hình 1.10. Mô hình phân lớp sơ bộ làn điệu chèo và dân ca quan họ 16](#_Toc515211724)

[Hình 1.11. Mô hình phân lớp cụ thể làn điệu chèo và dân ca quan họ 17](#_Toc515211725)

[Hình 1.12. Mô hình tổng quát hệ thống phân loại nhạc theo thể loại 18](#_Toc515211726)

[Hình 2.1. Thành phần của gói công cụ ALIZE 20](#_Toc515211727)

[Hình 2.2. Sơ đồ sử dụng các công cụ của SPro và ALIZE trong phân lớp làn điệu[6] 21](#_Toc515211728)

[Hình 3.1. Hệ thống thử nghiệm với dữ liệu không nằm trong tập huấn luyện 28](#_Toc515211729)

[Hình 3.2. Thử nghiệm hệ thống phân lớp với các dữ liệu dùng để thử nghiệm lấy từ tập huấn luyện 29](#_Toc515211730)

[Hình 3.3. Các bước xây dựng hệ phân lớp sử dụng SPro & ALIZE 32](#_Toc515211731)

[Hình 4.1. Kết quả phân lớp sơ bộ khi thay đổi hệ số M với dữ liệu thử nghiệm không nằm trong tập huấn luyện 40](#_Toc515211732)

[Hình 4.2. Kết quả nhận dạng cụ thể làn điệu chèo khi thay đổi hệ số M với dữ liệu thử nghiệm không nằm trong tập huấn luyện 43](#_Toc515211733)

[Hình 4.3. Kết quả nhận dạng cụ thể làn điệu dân ca quan họ khi thay đổi hệ số M với dữ liệu thử nghiệm không nằm trong tập huấn luyện 46](#_Toc515211734)

[Hình 4.4. Biểu đồ kết quả của hệ thống phân lớp sơ bộ có độ dài của file thử nghiệm và số thành phần Gauss thay đổi 48](#_Toc515211735)

[Hình 4.5. Biểu đồ kết quả phân lớp cụ thể làn điệu chèo có độ dài file thử nghiệm và số thành phần Gauss thay đổi 50](#_Toc515211736)

[Hình 4.6. Biểu đồ kết quả phân lớp cụ thể làn điệu dân ca quan họ có độ dài file thử nghiệm và số thành phần Gauss thay đổi 52](#_Toc515211737)

DANH MỤC BẢNG

[Bảng 2.1: Mô tả tùy chọn sử dụng chuẩn hóa năng lượng 23](#_Toc515211738)

[Bảng 2.2: Mô tả tùy chọn sử dụng phát hiện tín hiệu tiếng nói 23](#_Toc515211739)

[Bảng 2.3: Mô tả tùy chọn chuẩn hóa đặc trưng 24](#_Toc515211740)

[Bảng 2.4: Mô tả tùy chọn sử dụng trong huấn luyện mô hình nền 25](#_Toc515211741)

[Bảng 2.5: Mô tả tùy chọn trong huấn luyện mô hình đích 25](#_Toc515211742)

[Bảng 2.6: Mô tả tùy chọn tính điểm số của mô hình GMM 26](#_Toc515211743)

[Bảng 3.1: Thông tin về 25 làn điệu chèo 30](#_Toc515211744)

[Bảng 3.2: Thông tin về 25 làn điệu quan họ 31](#_Toc515211745)

[Bảng 4.1. Ma trận nhầm lẫn phân lớp sơ bộ với dữ liệu thử nghiệm không nằm trong tập huấn luyện (M = 512) 40](#_Toc515211746)

[Bảng 4.2. Ma trận nhầm lẫn nhận nhận dạng cụ thể làn điệu chèo với dữ liệu thử nghiệm không nằm trong tập huấn luyện (với M = 512) 41](#_Toc515211747)

[Bảng 4.3. Ma trận nhầm lẫn nhận nhận dạng cụ thể làn điệu dân ca quan họ với dữ liệu thử nghiệm không nằm trong tập huấn luyện (với M = 512) 44](#_Toc515211748)

[Bảng 4.4. Ma trận nhầm lẫn phân lớp sơ bộ với dữ liệu thử nghiệm lấy từ tập huấn luyện (M = 512) 46](#_Toc515211749)

[Bảng 4.5. Độ chính xác trung bình của hệ thống phân lớp sơ bộ có độ dài của file thử nghiệm và số thành phần Gauss thay đổi 47](#_Toc515211750)

[Bảng 4.6. Độ chính xác trung bình của hệ thống phân lớp cụ thể làn điệu chèo tùy theo độ dài của file thử nghiệm với M = 512 49](#_Toc515211751)

[Bảng 4.7. Độ chính xác trung bình của hệ thống phân lớp cụ thể làn điệu chèo có độ dài của file thử nghiệm và số thành phần Gauss thay đổi 49](#_Toc515211752)

[Bảng 4.8. Độ chính xác trung bình của hệ thống phân lớp cụ thể làn điệu dân ca quan họ tùy theo độ dài của file thử nghiệm với M = 512 51](#_Toc515211753)

[Bảng 4.9. Độ chính xác trung bình của hệ thống phân lớp cụ thể làn điệu dân ca quan họ có độ dài của file thử nghiệm và số thành phần Gauss thay đổi 51](#_Toc515211754)

[Bảng A.1. Kết quả phân lớp sơ bộ đối của từng làn điệu với các hệ số M khác nhau 55](#_Toc515211755)

[Bảng A.2. Ma trận nhầm lẫn phân lớp sơ bộ với các hệ số M khác nhau 56](#_Toc515211756)

[Bảng A.3. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 16 56](#_Toc515211757)

[Bảng A.4. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 32 57](#_Toc515211758)

[Bảng A.5. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 64 57](#_Toc515211759)

[Bảng A.6. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 128 58](#_Toc515211760)

[Bảng A.7. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 256 58](#_Toc515211761)

[Bảng A.8. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 1024 59](#_Toc515211762)

[Bảng A.9. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 2048 59](#_Toc515211763)

[Bảng A.10. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 4096 60](#_Toc515211764)

[Bảng A.11. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 16 60](#_Toc515211765)

[Bảng A.12. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 32 61](#_Toc515211766)

[Bảng A.13. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 64 61](#_Toc515211767)

[Bảng A.14. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 128 62](#_Toc515211768)

[Bảng A.15. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 256 62](#_Toc515211769)

[Bảng A.16. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 1024 63](#_Toc515211770)

[Bảng A.17. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 2048 63](#_Toc515211771)

[Bảng A.18. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 4096 64](#_Toc515211772)

[Bảng B.1. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 16 64](#_Toc515211773)

[Bảng B.2. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 32 64](#_Toc515211774)

[Bảng B.3. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 64 65](#_Toc515211775)

[Bảng B.4. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 128 65](#_Toc515211776)

[Bảng B.5. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 256 65](#_Toc515211777)

[Bảng B.6. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 1024 65](#_Toc515211778)

[Bảng B.7. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 2048 66](#_Toc515211779)

[Bảng B.8. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 4096 66](#_Toc515211780)

[Bảng B.9. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 4s 66](#_Toc515211781)

[Bảng B.10. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 6s 67](#_Toc515211782)

[Bảng B.11. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 8s 67](#_Toc515211783)

[Bảng B.12. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 10s 68](#_Toc515211784)

[Bảng B.13. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 12s 68](#_Toc515211785)

[Bảng B.14. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 14s 69](#_Toc515211786)

[Bảng B.15. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 16s 69](#_Toc515211787)

[Bảng B.16. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng độ dài của file huấn luyện 70](#_Toc515211788)

[Bảng B.17. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 4s 70](#_Toc515211789)

[Bảng B.18. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 6s 71](#_Toc515211790)

[Bảng B.19. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 8s 71](#_Toc515211791)

[Bảng B.20. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 10s 72](#_Toc515211792)

[Bảng B.21. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 12s 72](#_Toc515211793)

[Bảng B.22. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 14s 73](#_Toc515211794)

[Bảng B.23. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 16s 73](#_Toc515211795)

[Bảng B.24. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng độ dài của file huấn luyện 74](#_Toc515211796)

MỞ ĐẦU

Lý do chọn đề tài

Hiện nay đang trong giai đoạn của cuộc Cách mạng công nghệ 4.0, sự bùng nổ về thông tin đang diễn ra trên toàn cầu, chưa khi nào mà việc con người đặt câu hỏi và tìm kiếm câu trả lời lại dễ dàng như hiện nay. Nhưng đối với bất kỳ câu chuyện nào cũng tồn tại hai mặt của một vấn đề, sự bùng nổ về mặt dữ liệu dẫn tới nhu cầu về mặt quản lý cũng khắt khe hơn, một trong những nguồn dữ liệu mà em muốn nhắc tới đó chính là âm nhạc.

Âm nhạc là một trong những thông tin phổ biến nhất và trên các trang web về nghe nhạc trực tuyến có đến hàng triệu bản nhạc, điều này đặt ra một thách thức lớn trong việc tổ chức phân loại, truy xuất và tìm kiếm. Một trang web nghe nhạc mà chỉ liệt kê một cách đơn thuần danh sách các bài hát có trong hệ thống thì không thể coi đây là một trang web tốt được, chính vì vậy nó đòi hỏi phải được phân loại, sắp xếp theo tên ca sĩ, tên album, thể loại, năm phát hành,… Từ nhu cầu này đã nảy sinh ra yêu cầu về việc phân loại âm nhạc, nhưng điều gì sẽ xảy ra nếu chúng ra thực hiện điều này thủ công, có thể với số lượng vài trăm file âm thanh thì chưa có vấn đề gì xảy ra nhưng câu chuyện sẽ trở nên hoàn toàn khác khi con số này lên tới hàng triệu. Do đó phân loại nhạc tự động thành các thể loại khác nhau là một nhiệm vụ vô cùng quan trọng để làm nền tảng cho việc truy xuất và tổ chức các thư viện nhạc.

Là một người có tình yêu với văn hóa truyền thống nên em rất hứng thú với những giá trị văn hóa cổ truyền của dân tộc và nhạc cổ truyền cũng không phải là ngoại lệ. Tuy nhiên, các thể loại nhạc cổ tryền của Việt Nam rất là phong phú và dâ dạng chính vì vậy trong khuôn khổ của một đồ án tốt nghiệp em đã quyết định lựa chọn đề tài “**Phân lớp một số làn điệu chèo và dân ca quan họ**” làm để tài nghiên cứu của mình. Với mong muốn thông qua việc này có thể góp một nhỏ vào việc trùng hưng lại những làn điệu chèo và dân ca quan họ cổ, để không chỉ có thế hệ chúng em mà cả những thế hệ sau này vẫn sẽ tiếp tục duy trì ngọn lửa văn hóa truyền thống này.

Mục đích, phạm vi nghiên cứu

* Tìm hiểu tổng quan về phân lớp.
* Tìm hiểu một số mô hình phân lớp thường dùng.
* Tìm hiểu gói thư viện LIA-RAL của bộ công cụ ALIZE dùng để xây dựng mô hình GMM.
* Tìm hiểu công cụ SPro dùng để trích chọn đặc trưng.
* Thực hiện phân lớp một số làn điệu chèo và dân ca quan họ sử dụng mô hình GMM.

Đối tượng nghiên cứu

* Nghiên cứu tổng quan về phân lớp.
* Nghiên cứu một số mô hình phân lớp thường dùng.
* Nghiên cứu sử dụng mô hình GMM trong phân lớp một số làn điệu chèo và dân ca quan họ.
* Nghiên cứu bộ công cụ ALIZE và SPro.

Phương pháp nghiên cứu

* Nghiên cứu lý thuyết về nhận dạng thể loại âm nhạc.
* Nghiên cứu sử dụng bộ công cụ ALIZE trên môi trường Linux trong phân lớp làn điệu âm nhạc.
* Nghiên cứu sử dụng công cụ SPro trên môi trường Linux trong trích chọn đặc trưng.
* Thu thập các file âm thanh của chèo và dân ca quan họ, tiến hành thử nghiệm nhận dạng cho từng làn điệu một.

Nhiệm vụ nghiên cứu

* Tìm hiểu khái quát về phân lớp.
* Tìm hiểu mô hình GMM và một số mô hình phân lớp khác.
* Tìm hiểu về bộ công cụ ALIZE, SPro.
* Thực hiện phân lớp một số làn điệu chèo và dân ca quan họ sử dụng bộ công cụ ALIZE và SPro.

Nội dung đồ án được trình bày trong 53 trang và được chia thành 5 chương:

* Chương 1: Trình bày tổng quan về phân lớp nói chung và phân lớp làn điệu chèo và dân ca quan họ nói riêng. Chương này cũng trình bày một số kết quả nghiên cứu liên quan đến phân loại thể loại âm nhạc tại Việt Nam.
* Chương 2: Trình bày về bộ công cụ ALIZE và SPro cùng với cách sử dụng những bộ công cụ này việc thử nghiệm thực tế.
* Chương 3: Chương này sẽ trình bày về cơ sở dữ liệu được sử dụng trong đề tài, cách triển khai hệ thống phân lớp với bộ công cụ ALIZE và SPro.
* Chương 4: Trình bày kết quả thu được của đề tài.
* Chương 5: Nêu lên những vấn đề đã giải quyết trong đề tài và định hướng phát triển.

CHƯƠNG 1. CƠ SỞ LÝ THUYẾT PHÂN LỚP

* 1. Tổng quan về phân lớp

1. Khái niệm

Ngày nay phân lớp dữ liệu là một trong những hướng nghiên cứu chính của khai phá dữ liệu, nhất là trong thời đại bùng nổ của big data đặt ra nhu cầu từ một cơ sở dữ liệu với nhiều thông tin ẩn con người có thể trích rút ra những thông tin có lợi để phục vụ cho các nghiệp vụ thông minh.

Phân lớp được hiểu là quá trình nhóm các đối tượng có những điểm tương đồng lại với nhau thành hữu hạn các nhóm. Có hai dạng bài toán phân lớp thường được đề cập đến trong học máy (ML) đó là:

* **Phân lớp nhị phân (Binary Classification)**: được hiểu là việc phân các phần tử của một tập hợp nhất định thành hai nhóm dựa trên cơ sở của các quy tắc phân lớp. Một trong số các bài toán tiêu biểu của phân lớp nhị phân là xác định một bức thư điện tử có phải là thư rác hay không.
* **Phân lớp đa lớp (Multiclass Classification)**: tương tự với logic của phân lớp nhị phân, tuy nhiên sự khác biệt lớn nhất của phân lớp đa lớp và phân lớp nhị phân đó chính là độ chính xác của bài toán bởi vì việc phân thành hai lớp sẽ dễ hơn rất nhiều so với việc phân thành nhiều lớp. Một số bài toán thực tế của phân lớp đa lớp là: nhận dạng khuôn mặt, nhận dạng chữ viết tay, nhận dạng ngôn ngữ,…

Việc giải quyết bài toán phân lớp đa lớp có thể được đơn giản hóa bằng việc chuyển thành nhiều bài toán phân lớp nhị phân. Ví dụ: bài toán nhận diện khuôn mặt thì chúng ta có thể chia thành hai bài toán: bài toán một là nhận diện khuôn mặt là nam hay nữ, sau đó mới đi đến giải quyết bài toán thứ hai là nhận diện cụ thể khuôn mặt là của người nào. Thời gian thực hiện của việc chia nhỏ công việc này sẽ lâu hơn nhưng đem lại kết quả chính xác sẽ cao hơn do việc thu nhỏ phạm vi của bài toán.

1. Các bước để xây dựng một hệ thống phân lớp

Để xây dựng một hệ thống phân lớp cần thực hiện qua hai bước sau đây:

* Bước 1: Xây dựng mô hình từ tập dữ liệu huấn luyện.
* Bước 2: Đánh giá mô hình, mục đích của việc này để kiểm tra tính đúng đắn của mô hình để từ đó có thể dùng nó để phân lớp dữ liệu mới.

Ở bước 1, đối với tập dữ liệu huấn luyện trước tiên sẽ được chuẩn hóa và lựa chọn những thông tin cần thiết liên quan đến bài toán phân lớp cần xử lý sau đó mỗi bộ dữ liệu sẽ được phân vào một lớp xác định trước (xác định bởi thuộc tính dán nhãn lớp), từ tập dữ liệu này sẽ xây dựng được mô hình dựa trên các luật phân lớp, các cây quyết định hoặc các công thức toán học.

Bước 2 sẽ tiến hành phân lớp cho những đối tượng mới hoặc chưa được phân lớp từ đó chúng ta có thể đánh giá được độ chính xác của mô hình dựa trên việc so sánh lớp biết trước của bộ dữ liệu dùng để kiểm tra với kết quả thu được từ mô hình.

* 1. Một số mô hình phân lớp thường dùng
     1. Mô hình K-láng giềng gần nhất (K-NN)

K-láng giềng gần nhất là một trong những thuật toán học có giám sát (Supervised learning) - là thuật toán dự đoán đầu ra (outcome) của một dữ liệu mới (new input) dựa trên các cặp (input, outcome) đã biết từ trước. Khi tiến hành huấn luyện thì thuật toán này không học một điều gì từ dữ liệu huấn luyện, mọi tính toán sẽ được thực hiện khi nó cần dự đoán kết quả của dữ liệu mới.

Trước khi triển khai thuật toán cần phải chuẩn bị một tập dữ liệu huấn luyện – các dữ liệu này đều đã được dán nhãn phân lớp. Khi cần xác định một đối tượng chưa biết thuộc vào lớp nào, thuật toán sẽ được tiến hành thông qua các bước sau đây:

1. Xác định giá trị tham số K (số điểm láng giềng gần nhất).
2. Tính khoảng cách giữa đối tượng cần phân lớp với các điểm trong tập dữ liệu dùng để huấn luyện.
3. Lấy tất cả các lớp của K điểm gần nhất.
4. Trong các điểm láng giềng này lớp nào có các điểm chiếm phần lớn thì đối tượng cần phân lớp sẽ thuộc về lớp đó.[8]

Cụ thể, kết quả phân lớp trả về chỉ dựa trên thông tin của K điểm dữ liệu trong tập huấn luyện gần nó nhất, không quan tâm đến việc có một vài điểm dữ liệu trong những điểm này là nhiễu (điều này dễ xảy ra với giá trị K nhỏ)

Ví dụ về việc triển khai thuật toán K-NN

Hình 1.1. Mô tả việc triển khai thuật toán K-NN

Tập dữ liệu huấn luyện của chúng ta có là hai lớp được ký hiệu là hình vuông và hình tam giác, đối tượng cần phân lớp là hình tròn. Bây giờ chúng ta sẽ đi tiến hành dự đoán lớp của đối tượng này dựa vào việc lựa chọn số láng giềng gần nhất với nó. Nhìn vào hình 1.1, ta xét với với một số giá trị K như sau:

* – so sánh với 1 điểm láng giềng gần nhất thì đối tượng này sẽ thuộc lớp hình vuông do khoảng cách tới 1 điểm gần nhất là hình vuông.
* – so sánh với 2 điểm láng giềng gần nhất thì sẽ không xác định được lớp mà đối tượng này thuộc về bởi vì 2 điểm gần nhất thì 1 điểm thuộc về lớp hình vuông và một điểm thuộc về lớp hình tam giác.
* – so sánh với 5 điểm láng giềng gần nhất thì đối tượng này sẽ thuộc vào lớp tam giác, bởi trong 5 điểm gần nhất thì có 3 điểm thuộc lớp tam giác và 2 điểm thuộc lớp hình vuông nên chúng ta sẽ chọn lớp mà có số điểm chiếm ưu thế hơn.

Trong thực tế, khi sử dụng thuật toán K-NN để đạt được hiệu quả tốt nhất người ta thường thử nghiệm với các giá trị K khác nhau, sau đó tổng hợp kết quả và lựa chọn ra giá trị K đem lại kết quả thử nghiệm chính xác cao nhất. Bên cạnh việc thay đổi giá trị K thì cũng cần luân phiên thay kiểm thử với bộ dữ liệu huấn luyện và thử nghiệm khác nhau sau đó lấy kết quả trung bình, vì rất có thể dữ liệu phân chia trong một trường hợp cụ thể là rất tốt hoặc rất xấu.

Ưu điểm và nhược điểm của thuật toán K-NN

* Ưu điểm:
  + Độ phức tạp tính toán của quá trình huấn luyện bằng 0 (do không có quá trình huấn luyện).
  + Việc dự đoán kết quả của dữ liệu mới đơn giản.
* Nhược điểm:
  + Kết quả trả về dễ bị ảnh hưởng bởi nhiễu khi giá trị K nhỏ.
  + Vì mọi thao tác tính toán đều nằm ở giai đoạn thử nghiệm nên nếu với bộ dữ liệu huấn luyện lớn thì việc tính khoảng cách đến từng điểm dữ liệu sẽ gây ảnh hưởng đến thời gian chạy của chương trình.
    1. Mô hình cây quyết định (Decision Tree)

1. Khái niệm

Cây quyết định là một kiểu mô hình dự báo, đây là một dạng mô hình phân cấp có cấu trúc được dùng để phân lớp các đối tượng dựa vào tập hợp các luật.

Một cây quyết định bao gồm các thành phần sau:

* Các nút trong (internal node) hay nút không phải nút lá (non-leaf node) được dùng để biểu diễn một thuộc tính hay một câu hỏi. Các nút này thường có hai hay nhiều nút con, các nút con này có thể là một nút lá hoặc là một nút trong khác. Nếu tất cả các nút không phải nút lá đều có hai nút con thì ta nói rằng đó là một cây quyết định nhị phân (Binary Decision Tree).
* Nút lá (leaf node) biểu diễn các lớp quyết định.
* Nút không phải là nút lá mà ở trên cùng thì đó gọi là nút gốc (root node).
* Các nhánh (branch) biểu diễn các giá trị có thể có của thuộc tính hoặc câu trả lời cho câu hỏi được nêu ra ở nút cha.

Cây quyết định có thể được dùng để phân lớp bằng cách xuất phát từ gốc của cây và di chuyển theo các nhánh cho đến khi gặp nút lá. Trên cơ sở đó chúng ta có thể xây dựng các luật quyết định.

Cây quyết định là một mô hình học có giám sát, có thể được áp dụng vào cả hai bài toán phân lớp (classification) và hồi quy (regression – ước lượng kết quả trả về là số thực, ví dụ như ước tính giá của một ngôi nhà hay thời gian sửa chữa của một chiếc ô tô).

Ví dụ về bài toán phân lớp:

Đ

S

S

Đ

S

Đ

Hình 1.2. Mô tả bài toán phân lớp sử dụng cây quyết định

Nhìn vào hình 1.2, ta có hai lớp hình vuông và hình tam giác trên mặt phẳng không gian hai chiều. Bài toán đặt ra ở đây là tìm ranh giới phân chia hai lớp này và xác định một điểm dữ liệu mới thuộc vào lớp nào, nhìn vào hình ta thấy ranh giới của hai lớp này là giao nhau của 2 đường thằng , và Từ ranh giới này ta xây dựng nên các luật để xác định một điểm dữ liệu mới được đưa vào sẽ thuộc lớp nào. Ví dụ điểm mới này có tọa độ , đối chiếu với sơ đồ cây quyết định bên phải ta có, nếu thì ta xác định điểm này thuộc lớp hình tam giác, còn nếu không ta xét tiếp đến câu hỏi thứ 2, nếu thì điểm này thuộc lớp hình tam giác, ngược lại thì ta lại xét đến câu hỏi thứ 3… Quá trình này được tiến hành liên tục cho đến khi xác định được lớp của đối tương thì dừng lại.

1. Xây dựng cây quyết định

Đứng trước một bài toán để tạo nên một cây quyết định chúng ta có hai giải pháp như sau:

* **Giải pháp 1:** Tham khảo ý kiến của các chuyên gia liên quan đến vấn đề cần xử lý. Đây là một giải pháp được đánh giá là hữu hiệu bởi độ chính xác cao của nó. Tuy nhiên, không phải lúc nào cũng có thể tìm được chuyên gia để giải quyết vấn đề mà chúng ta gặp phải.
* **Giải pháp 2:** Tạo ra các thuật toán để tự xây dựng cây quyết định, điều này được thực hiện dựa trên những bộ dữ liệu dùng để huấn luyện.[7]

Việc xây dựng một cây quyết định bắt đầu từ gốc, tất cả các tập dữ liệu huấn luyện đều ở gốc, sau đó thuật toán sẽ tìm ra câu hỏi đầu tiên tốt nhất để bắt đầu sau đó phân chia các mẫu dựa trên các câu hỏi được lựa chọn. Sau mỗi câu hỏi, dữ liệu được phân chia vào từng nút con tương ứng với các câu trả lời cho câu hỏi đó. Câu hỏi ở đây chính là một thuộc tính, câu trả lời chính là giá trị của thuộc tính đó. Một phép phân chia tốt nhất là khi dữ liệu trong nút con hoàn toàn thuộc vào một lớp nào đó – lúc này thì nút con được coi như là một nút lá. Khi không còn đặt được thêm câu hỏi nào có giá trị nữa thì thuật toán sẽ dừng lại và kết thúc quá trình xây dựng cây quyết định. Hiệu quả phân lớp của cây quyết định phụ thuộc rất nhiều vào tập dữ liệu huấn luyện, vì vậy cần một tập dữ liệu đủ lớn và chính xác.

Sau khi xây dựng được cây quyết định thì trước khi đưa vào thử nghiệm cần chuẩn hóa cây bằng việc xác định và loại bỏ những nhánh nhiễu (những câu hỏi trả về kết quả không có nhiều giá trị).

Ưu điểm của cây quyết định:[13]

* Đây là một mô hình dễ hình dung, gần với thực tế.
* Do tất cả các thông tin của dữ liệu đều có thể sử dụng để xây dựng nên các luật của cây quyết định nên việc chuẩn hóa dữ liệu là không cần thiết, điều này giúp giảm chi phí thiết kế hệ thống.
* Có thể đưa ra câu trả lời minh bạch về kết quả trả về của cây quyết định dựa vào tập hợp các luật.
* Cây quyết định có thể xử lý tốt một lượng dữ liệu lớn trong khoảng thời gian ngắn.
  + 1. Mô hình Markov ẩn (HMM)

Học thuyết về chuỗi Markov được phát triển vào những năm 1990, được đặt theo tên của nhà toán học người Nga Andrei Andreyevich Markov. Mô hình Markov ẩn phát triển vào cuối những năm 60 và được sử dụng rộng rãi trong lĩnh vực nhận dạng tiếng nói vào những năm 1960 – 1970 và được đưa vào khoa học máy tính năm 1989.

1. Mô hình Markov

Xét một hệ thống bao gồm trạng thái phân biệt, được đánh số thứ tự . Tại thời điểm t bất kỳ, hệ thống có thể chuyển từ trạng thái sang một trong trạng thái còn lại hoặc chuyển về chính trạng thái . Như vậy, ở tại thời điểm , từ trạng thái có nhánh thao tác chuyển trạng thái, mỗi một nhánh này có một xác suất xảy ra khác nhau và được gọi là **xác suất chuyển trạng thái.**

* Gọi là tập hợp tất cả các trạng thái:

(1.1)

* Gọi là trạng thái đạt được ở thời điểm , là ma trận chuyển đổi chứa các giá trị là xác suất chuyển trạng thái từ sang . Xác suất chuyển trạng thái không phụ thuộc vào thời gian và độc lập với các trạng thái chuyển trước đó. Chỉ phụ thuộc duy nhất vào trạng thái hiện tại. Quá trình mang tính ngẫu nhiên này, được coi là “có thuộc tính Markov”

, = P( = | = ) (1.2)

Giữa mỗi bước thời gian, trạng thái tiếp theo được chọn một cách ngẫu nhiên. Trạng thái hiện tại sẽ quyết định xác suất phân bố của trạng thái tiếp theo. Nếu trạng thái tiếp theo chỉ phụ thuộc vào trạng thái hiện tại và không phụ thuộc trạng thái nào trong quá khứ thì đây được gọi là Mô hình Markov bậc 1.

Hình 1.3. Mô hình Markov bậc 1

(1.3)

Mô hình Markov bậc 2 là trạng thái tiếp theo phụ thuộc vào trạng thái hiện tại và trạng thái liền kề trước đó.

Hình 1.4. Mô hình Markov bậc 2

(1.4)

1. Mô hình Markov ẩn

Mô hình Markov ẩn (HMM) là dạng mở rộng của mô hình Markov. Trong mô hình Markov, các sự kiện quan sát được nằm trong mỗi trạng thái và phụ thuộc vào hàm mật độ xác suất trong các trạng thái đó. Để dễ hình dung, chúng ta có thể xem hình 1.5

0.5

0.3

0.5

0.3

0.2

0.05

0.2

0.2

0.75

Hình 1.5. Mô hình Markov trong bài toán dự báo thời tiết

Mô hình Markov thực hiện tính toán trực tiếp trên các trạng thái còn mô hình Markov ẩn không tính toán kết quả trực tiếp trên các trạng thái đó mà phải thông qua một sự kiện trạng thái khác gọi là các sự kiện trạng thái quan sát.

Hình 1.6. Mô hình Markov ẩn 3 trạng thái

Hình 1.6 mô tả mô hình Markov ẩn 3 trạng thái với các sự kiện có thể quan sát được trong mỗi trạng thái là . Xác suất quan sát được sự kiện trong trạng thái phụ thuộc vào hàm xác suất . Hàm được gọi là hàm mật độ xác suất của các sự kiện được quan sát.

Vẫn với bài toán dự báo thời tiết được đề cập ở hình 1.5, nếu xét trong mô hình Markov ẩn thì các trạng thái có thể quan sát được là: lầy lội, ẩm ướt, khô.

HMM được xác định bởi 5 thành phần:

* Tập hợp các trạng thái ẩn: – số trạng thái

(1.5)

* Tập hợp các quan sát: – số các quan sát

(1.6)

* là chuỗi các trạng thái có thể xảy ra, có chiều dài

(1.7)

* Tương ứng với chuỗi các trạng thái có chuỗi các quan sát có thể quan sát được:

(1.8)

* là ma trận xác suất khởi tạo:

, (1.9)

* là ma trận chứa những giá trị xác suất chuyển đổi từ trạng thái sang trạng thái .

, (1.10)

* là ma trận xác suất quan sát, chứa những giá trị xác suất của quan sát từ trạng thái , độc lập với thời gian

,

(1.11)

Tóm lại, các thành phần của HMM bao gồm:

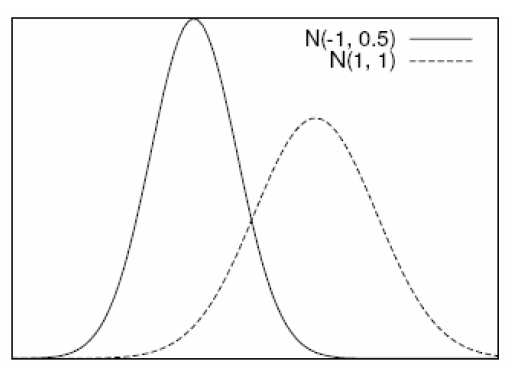
* 2 tham số không đổi về kích cỡ: và (tổng số các trạng thái và các quan sát).
* 3 tập hợp phân bố xác suất: . Một mô hình HMM được ký hiệu

1. Ứng dụng của HMM

* HMM được sử dụng nhiều trong phân tích ngôn ngữ: Nhận dạng tiếng nói (đối tượng quan sát được là âm thanh, đối tượng ẩn là từ ngữ).
* Nhận dạng chữ viết tay (đối tượng quan sát được là ký hiểu, đối tượng ẩn là từ ngữ).
* Phân loại từ ngữ (đối tượng quan sát được là từ ngữ, đối tượng ẩn là các loại từ: danh từ, động từ, tính từ).
* Hệ thống phiên dịch ngôn ngữ (đối tượng quan sát được là từ nước ngoài, đối tượng ẩn là từ ngữ ứng với ngôn ngữ cần dịch).
  + 1. Mô hình hỗn hợp Gauss

1. Đặc tả mô hình

Mô hình hợp Gauss (GMM) là một dạng mô hình thống kê được xây dựng từ việc huấn luyện các tham số thông qua dữ liệu học. Mô hình GMM còn có một số tên gọi khác như Weighted Normal Distribution Sums hay Radial Basis Function Approximations…[14]



Hình 1.7. Hàm mật độ Gauss

Về cơ bản, mô hình GMM xấp xỉ một hàm mật độ xác suất bằng hợp các hàm mật độ Gauss. Hình 1.7 minh họa hai hàm mật độ Gauss với các tham số khác nhau.

Một cách hình thức, hàm mật độ xác suất của phân phối Gauss được cho bởi công thức:

 ( 1.11)

trong đó, μ là giá trị trung bình, σ là độ lệch chuẩn. Trong trường hợp là vector

gồm thành phần, hàm mật độ xác suất của phân phối Gauss được cho

bởi công thức:

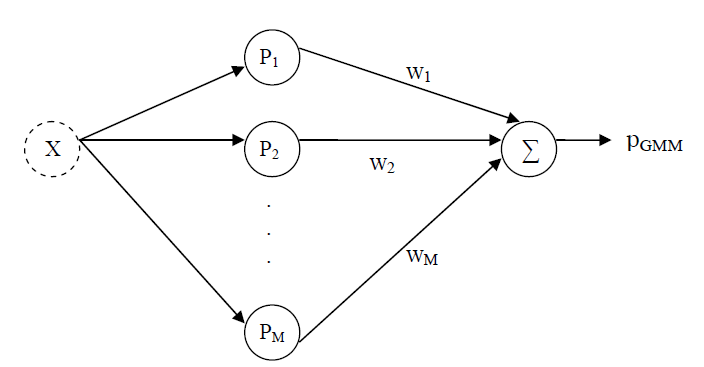


khi đó, là vector trung bình, là ma trận hiệp phương sai. Nếu chọn , công thức (1.11) sẽ trở thành hàm mật độ chuẩn Gauss:



Từ “Gauss” được đặt theo tên của nhà toán học người Đức Carl Friedrich Gauss.

Ông đã định nghĩa hàm mật độ Gauss và áp dụng trong phân tích dữ liệu thiên văn.



Hình 1.8. Mô hình GMM

Cho trước phân phối Gauss , hàm mật độ xác suất của mô hình

GMM được minh họa trong hình 1.8 chính là tổng trọng của phân phối Gauss

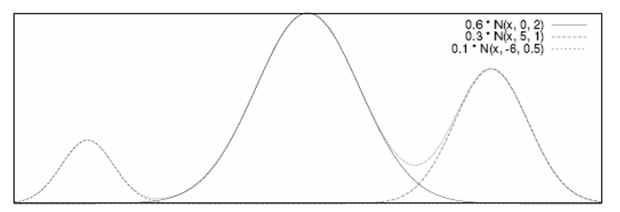
theo công thức:



trong đó, là trọng số của phân phối Gauss thứ, thỏa ràng buộc và

 Các trọng số này thể hiện mức độ ảnh hưởng của mỗi phân phối Gaussđối với mô hình GMM. Như vậy, phân phối Gauss có phương sai và trọng số lớn bao nhiêu thì có mức độ ảnh hưởng lớn bấy nhiêu đối với kết xuất của mô hình.

Hình 1.9 cho thấy mức độ ảnh hưởng của từng phân phối Gauss lên GMM.



Hình 1.9. Hàm mật độ của GMM có 3 phân phối Gauss

Như vậy, một mô hình GMM có M phân phối Gauss sẽ được đại diện bởi bộ tham số . Trong hướng tiếp cận GMM giải quyết bài toàn phân lớp làn điệu, mỗi làn điệu sẽ được mô hình hóa bằng một mô hình GMM mà bộ tham số của nó sẽ được xác định thông qua việc huấn luyện trên tập mẫu học của làn điệu tương ứng.

Tùy thuộc vào cách tổ chức của ma trận hiệp phương sai (Covariance Matrix), GMM có thể có một số biến thể khác nhau:

* Nodal covariance matrices GMM: mỗi phân phối Gauss trong GMM có một ma trận hiệp phương sai riêng.
* Grand Covariance Matrix GMM: mọi phân phối Gauss trong một GMM dùng chung một ma trận hiệp phương sai.
* Global Covariance Matrix GMM: mọi phân phối Gauss trong tất cả các GMM dùng chung một ma trận hiệp phương sai.

Ngoài ra, xét về dạng thức, ma trận hiệp phương sai gồm hai loại: Full (dạng đầy đủ) và Diagonal (dạng ma trận đường chéo). Thông thường, dạng Nodal-Diagonal Covariance Matrices GMM được sử dụng phổ biến nhất.

1. Bài toán ước lượng mật độ

Trong bộ phân loại dựa trên mô hình thống kê, việc ước lượng các tham số của mô hình được thực hiện thông qua huấn luyện trên một số lượng lớn các dữ liệu học. Mục tiêu của bước huấn luyện là nhằm tổng quát hóa, mô hình hóa những đặc điểm chung nhất của tập dữ liệu học. Đối với mô hình GMM, một trong những kỹ thuật xác định bộ tham số của nó được áp dụng khá phổ biến là thuật toán Cực đại hóa kỳ vọng (EM). Bản thân EM là một thuật toán tổng quát, đem lại các kết quả khác nhau đối với các mô hình khác nhau. Ngoài ra, có hai tiêu chí ước lượng khác nhau trong EM:

- Cực đại khả hiện (ML): ước lượng tham số theo hướng cực đại hóa độ tương tự .

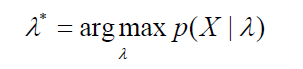
- Cực đại xác suất hậu nghiệm (MAP): ước lượng tham số theo hướng cực đại hóa xác suất quyết định .

Cho trước vectơ đặc trưng trích được từ dữ liệu âm thanh, ta có thể dễ dàng tính được độ tương tự . Tuy nhiên, trong định danh người nói, vai trò quyết định lại nằm ở xác suất . Sử dụng công thức Bayes, ta có tương quan giữa và :



trong đó, là xác suất xuất hiện của vectơ đặc trưng , là tần suất xuất

hiện của dữ liệu cần phân lớp được mô hình hóa bởi GMM tương ứng. Trong luật quyết định Bayes, độc lập; như vậy nếu giả định là đồng nhất cho mọi lớp, ta có thể quy vai trò quyết định từ về và áp dụng EM ước lượng theo hướng cực đại khả hiện:



Như vậy, với tập dữ liệu huấn luyện gồm mẫu  các trọng số, trung bình và phương sai của GMM ở mỗi bước lặp sẽ là:

Trọng số: 

Trung bình: 

Phương sai: 

trong đó,  là các thành phần tương ứng trong các vectơ  . Xác suất cho bởi công thức:



Trong quá trình xây dựng GMM, có hai vấn đề phát sinh là: số phân phối Gauss

của mô hình, và bộ tham số khởi đầu trước khi tiến hành thuật toán EM. Hiện tại,

vẫn chưa có giải pháp tối ưu trên lý thuyết cho việc chọn và . Thông thường, M sẽ được chọn qua thực nghiệm, còn λ0 sẽ được khởi tạo bằng thuật toán K-means

nhằm đem lại khả năng cao hơn cho việc đạt tối ưu toàn cục, đồng thời đẩy nhanh

tốc độ hội tụ trong huấn luyện.

* 1. Phân lớp làn điệu chèo và dân ca quan họ
     1. Đặc trưng của làn điệu chèo và dân ca quan họ

1. Đặc trưng của làn điệu chèo

Hát chèo được hình thành và bắt nguồn từ các làn điệu dân ca, lời hát Chèo lấy trong các sáng tác văn học dân gian ở vùng đồng bằng Bắc Bộ là chủ yếu. Ngoài ra các làn điệu Chèo còn chịu những ảnh hưởng từ hát Văn, hát Xẩm, hát Ca trù, hát Xoan, hát Quan họ...

Hiện nay có khoảng 250 làn điệu chèo[1], nhạc cụ tối thiểu dùng cho hát chèo là ba loại nhạc cụ dây: đàn nguyệt, đàn nhị và đàn bầu đồng thời thêm cả sáo nữa. Ngoài ra, các nhạc công còn sử dụng thêm trống và chũm chọe. Bộ gõ nếu đầy đủ thì có trống cái, trống con, trống cơm, thanh la, mõ. Trống con dùng để giữ nhịp cho hát, cho múa và đệm cho câu hát. Có câu nói "phi trống bất thành chèo" chỉ vị trí quan trọng của chiếc trống trong đêm diễn chèo. Trong chèo hiện đại có sử dụng thêm các nhạc cụ khác để làm phong phú thêm phần đệm như đàn thập lục, đàn tam thập lục, tiêu v.v...[11]

1. Đặc trưng của làn điệu dân ca quan họ

Hát quan họ là hình thức hát đồng giọng: người hát dẫn, người hát luồn, hát đối đáp dẫn giọng, luồn giọng một cách điêu luyện. Giọng của hai người hát cặp với nhau phải tương đồng đến mức hai giọng trở thành một để tạo thành một âm thanh thống nhất.

Dân ca quan họ có 213 giọng khác nhau, với hơn 400 bài hát. Lời một bài hát có hai phần: lời chính và lời phụ. Lời chính là phần cốt lõi, phản ánh nội dung của bài hát, lời phụ gồm tất cả những tiếng nằm ngoài lời ca chính, là tiếng đệm, tiếng đưa hơi như i hi, ư hư, a ha, v.v…

Dân ca quan họ chủ yếu là nghệ thuật phổ lời ca dao và thơ. Nghệ thuật này đỏi hỏi phải sử dụng những tiếng phụ, lời phụ bên cạnh những tiếng chính, lời chính nhằm làm cho tiếng hát trôi chảy, bổ sung ý nghĩa cho lời hát chính, làm cho lời ca them phong phú, linh hoạt, tăng cường tính nhạc của bài ca, phát triển giai điệu, làm cho âm nhạc của bài ca trở nên sinh động, bố cục trở nên hợp lý. Không dùng tiếng phụ, lời phụ thì lời ca dễ đơn điệu, mất cân đối.[12]

* + 1. Phân lớp làn điệu chèo và dân ca quan họ là gì?

Phân lớp làn điệu là một hệ thống sinh trắc học, thực hiện việc tính toán xác định làn điệu dựa trên cơ sở sử dụng các đặc điểm đặc trưng của mỗi làn điệu. Đặc trưng của mỗi làn điệu chứa các thông tin sau:

* Thông tin mức thấp: độ cao, cường độ, tần số, quang phổ,… những đặc điểm này thường được nhận dạng bởi hệ thống phân lớp qua quá trình huấn luyện để đưa ra quyết định.
* Thông tin mức cao: bao gồm lời bài hát, nhịp điệu, tiết tấu... những đặc điểm này thường liên quan đến việc áp dụng kinh nghiệm của con người để xác định từng làn điệu cụ thể.
  + 1. Các ứng dụng của phân lớp làn điệu

Phân lớp thể loại nói chung và phân lớp theo làn điệu nói riêng có những ứng dụng tích cực trong những lĩnh vực sau:

* Tổng hợp có hệ thống các file dữ liệu âm nhạc.
* Hỗ trợ việc tổ chức phân loại, tìm kiếm tên bài hát, làn điệu.
* Xây dựng các ứng dụng nhận dạng tên bài hát, làn điệu.
  + 1. Phương pháp phân lớp làn điệu chèo và dân ca quan họ

Vì số lượng làn điệu chèo và dân ca quan họ rất nhiều nên để tăng độ chính xác đầu tiên chúng ta thực hiện chia bài toán phân lớp theo từng làn điệu ban đầu thành 2 bài toán như sau:

* Bài toán 1: Phân lớp sơ bộ: xác đinh một làn điệu thuộc vào làn điệu chèo hay quan họ.
* Bài toán 2: Phân lớp cụ thể: từ kết quả của *Bài toán 1* nếu kết quả trả về chiếm phần trăm cao về phía làn điệu chèo thì dữ liệu sẽ được đưa vào nhận dạng theo làn điệu cụ thể trong lớp chèo, tương tự đối với làn điệu quan họ.

Hình 1.10 là mô hình phân lớp theo sơ bộ của **Bài toán 1**. Đầu tiên dữ liệu âm thanh sẽ được trích chọn đặc trưng, loại bỏ những khoảng lặng, nhiễu, sau đó dữ liệu này sẽ được chia thành 2 tập dữ liệu dùng để huấn luyện và thử nghiệm. Sau khi huấn luyện xong ta sẽ có một cơ sở dữ liệu liên quan đến hai lớp chèo và quan họ, công việc tiếp theo sẽ là tiến hành thử nghiệm với tập dữ liệu thử nghiệm bên trên với cơ sở dữ liệu này. Cuối cùng là việc tổng hợp kết quả của việc phân lớp sơ bộ để phục vụ cho việc phân lớp theo làn điệu cụ thể của **Bài toán 2**.

***Thử nghiệm***

Tổng hợp kết quả phân lớp

Cơ sở

dữ liệu

phân lớp sơ bộ

Dữ liệu huấn luyện

Dữ liệu thử nghiệm

***Huấn luyện***

Trích chọn đặc trưng

Phân chia dữ liệu huấn luyện và thử nghiệm

Tính độ tương đồng với mô hình phân lớp



Hình 1.10. Mô hình phân lớp sơ bộ làn điệu chèo và dân ca quan họ

Dữ liệu huấn luyện

Dữ liệu thử nghiệm

***Thử nghiệm***

Đối chiếu với kết quả phân lớp theo thể loại

Tập dữ liệu chèo

Tập dữ liệu quan họ

***Thử nghiệm***

Tổng hợp kết quả phân lớp

***Huấn luyện***

Cơ sở dữ liệu

phân lớp chèo

Dữ liệu chèo

Dữ liệu quan họ

Cơ sở dữ liệu

phân lớp quan họ

***Huấn luyện***

Tính độ tương đồng với mô hình phân lớp

Hình 1.11. Mô hình phân lớp cụ thể làn điệu chèo và dân ca quan họ

Tiếp theo, hình 1.11 là mô hình phân lớp cụ thể theo từng làn điệu chèo và dân ca quan họ cụ thể, sử dụng lại những kết quả có được thông qua bài toán 1, ta có được dữ liệu huấn luyện và thử nghiệm đã được trích chọn đặc trưng. Với dữ liệu huấn luyện ta đưa vào huấn luyện với mục đích tạo nên Cơ sở dữ liệu phân lớp theo từng làn điệu cụ thể, còn dữ liệu thử nghiệm thì ta so sánh với kết quả thu được ở giai đoạn một để phia dữ liệu thành hai phần, một phần để nhận diện về làn điệu chèo còn một phần để nhận diện về làn điệu quan họ. Cuối cùng tiến hành thử nghiệm và tổng hợp thành kết quả.

Việc thực hiện hai bài toán này sẽ được diễn ra liên tục và luân phiên nhau bằng việc thay đổi các phần dữ liệu huấn luyện và thử nghiệm cho nhau, thực hiện cho đến khi nào tất cả dữ liệu đều đã từng được sử dụng trpng huấn luyện và thử nghiệm.

1.4. Tìm hiểu một số công trình phân lớp âm nhạc đã có tại Việt Nam

1. * 1. Phân loại nhạc theo thể loại dùng phép biến đổi Wavelet rời rạc, tác giả: Phan Anh Cang, Phan Thượng Cang.[4]
2. Cơ sở dữ liệu

Nghiên cứu sử dụng nguồn dữ liệu cho huấn luyện và kiểm tra là bộ sưu tập nhạc GTZAN, gồm 10 thể loại nhạc (Blues, Classical, Country, Disco, Hiphop, Jazz, Metal, Pop, Reggae, Rock). Dựa trên cấu trúc phân loại âm thanh của bộ sưu tập nhạc GTZAN, bốn thể loại nhạc Classical, Rock, Jazz, Pop được chọn ngẫu nhiên để minh họa cho hệ thống phân loại nhạc theo thể loại.

Dữ liệu được chia thành 2 tập: huấn luyện và kiểm tra. Mỗi file có độ dài 30s với tần số 22050Hz Mono 16-bit ở định dạng wav.

Tập file audio huấn luyện được sử dụng để huấn luyện cho bộ phân loại KNN để đưa ra các quyết định cho hệ thống phân loại nhạc trong khi tập file audio kiểm tra sẽ được sử dụng để đánh giá hiệu quả của phương pháp đề xuất.

1. Phương pháp nhận dạng

Hệ thống phân loại nhạc theo thể loại gồm 2 pha: rút trích đặc trưng và huấn luyện hoặc phân loại.

Hệ thống sử dụng phương pháp biến đổi Wavelet rời rạc (DWT) để rút trích đặc trưng về nhịp điệu. Phương pháp phân loại KNN được sử dụng để nhận dạng các thể loại nhạc. Quá trình huấn luyện bao gồm việc sử dụng các vectơ đặc trưng đã được gán nhãn thể loại để huấn luyện cho mô hình phân loại KNN. Từ đó, bộ phân loại sẽ gán nhãn thể loại cho các vectơ đặc trưng mới một cách tự động.

Việc phân loại nhạc được thực hiện chủ yếu dựa vào 3 tập đặc trưng được rút trích từ tín hiệu audio như sau:

* Tập đặc trưng 1 (ĐT1): 19 đặc trưng về âm sắc.
* Tập đặc trưng 2 (ĐT2): 6 đặc trưng về nhịp điệu.
* Tập đặc trưng 3 (ĐT3): 5 đặc trưng về cao độ

Cơ sở dữ liệu nhạc

Cơ sở dữ liệu vectơ đặc trưng

Vectơ đặc trưng:

* Âm sắc
* Nhịp điệu/tiết tấu
* Cao độ

Trích rút đặc trưng

Huấn luyện nhận dạng(Bộ phân loại KNN)

Vectơ đặc trưng:

* Âm sắc
* Nhịp điệu/tiết tấu
* Cao độ

Trích rút đặc trưng

Tín hiệu audio

Thể loại nhạc:

* Classical
* Rock
* Jazz
* Pop

Hình 1.12. Mô hình tổng quát hệ thống phân loại nhạc theo thể loại

1. Kết quả nhận dạng

Công trình tiến hành thử nghiệm trên một số giá trị tham số (số láng giềng gần nhất). Trong đó, với giá trị , hệ thống cho kết quả phân loại tốt nhất. Việc phân loại dựa trên cả 3 tập đặc trưng thì thu được kết quả như bên dưới đây:

* Với , độ chính xác là
* Với , độ chính xác là
* Với, độ chính xác là
* Với , độ chính xác
* Với , độ chính xác

CHƯƠNG 2. CÁC CÔNG CỤ SỬ DỤNG TRONG ĐỂ TÀI

1. 1. Bộ công cụ SPro

SPro là một bộ công cụ cung cấp các lệnh thực thi các thuật toán trích chọn đặc trưng tiếng nói và trong các ứng dụng nhận dạng người nói.

SPro ban đầu được thiết kế để hiển thị giải phân tích quang phổ nhưng cũng cung cấp các kỹ thuật trích chọn đặc trưng cổ điển sử dụng trong ứng dụng tiếng nói. Có những lệnh chính sau đây:

* Năng lượng filter – bank
* Hệ số cepstral (filter-bank và dự đoán tuyến tính)
* Dự đoán tuyến tính dẫn xuất tiêu biểu (dự báo và phản ánh hệ số, tỷ lệ phạm vi và những dòng cặp phổ).

Các thư viện được viết trong ANSI C, cung cấp các chức năng sau:

* Tín hiệu đầu vào dạng sóng
* Xử lý tín hiệu ở mức thấp (FFT, phân tích LPC,..)
* Xử lý đặc trưng mức thấp (CMS, chuẩn hóa sự khác biệt, …)

Thư viện không cung cấp các hàm cho trích chọn đặc trưng ở mức cao mà trực tiếp chuyển đổi một dạng sóng vào các đặc trưng, chủ yếu là do hàm như vậy sẽ đòi hỏi một số lượng lớn các đối số để được linh hoạt.

SPro được sử dụng trên nền Linux, SPARC/SunOS và HP-UX. Có thể tải xuống SPro tại địa chỉ: [**http://www.irisa.fr/metiss/guig/spro/download.html**](http://www.irisa.fr/metiss/guig/spro/download.html)

* 1. Bộ công cụ ALIZE
     1. Nguồn gốc

ALIZE (ALIZÉ) là một thư viện mở hỗ trợ cho việc nhận dạng giọng nói. Mục đích của dự án này là cung cấp một bộ thư viện nền tảng cho phép bất kỳ ai cũng có thể phát triển các ứng dụng khác nhau trong lĩnh vực nhận dạng giọng nói như việc xác minh/xác định, phân lớp, v.v.

Dự án ALIZE được bắt đầu từ năm 2004, bởi đại học Avignon LIA thuộc tập đoàn ELISA với mục đích tạo ra một thư viện C++ mã nguồn mở để nhận dạng giọng nói. Tất cả mã nguồn của bộ công cụ này được phân phối thông qua giấy phép phần mềm nguồn mở (LPGL) và đã được thử nghiệm trên các nền tảng khác nhau bao gồm Window, Linux và Mac-OS.

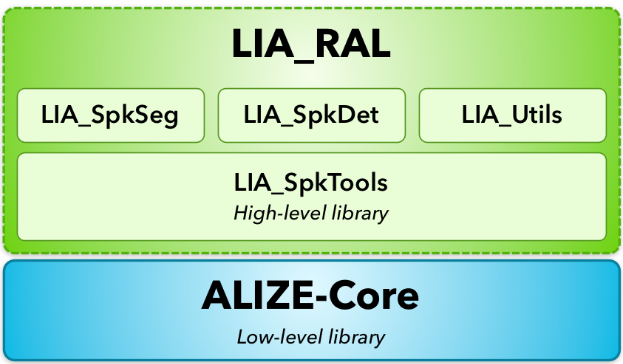
ALIZE là phần mềm miễn phí, có thể tải xuống tại địa chỉ: **http://alize.univ-avignon.fr/#download*.***

2.1.2. Giới thiệu về ALIZE

ALIZE được phát triển với kiến trúc đa tầng dựa trên mô hình UML. Với thư viện lõi là ALIZE-Core (là một thư viện mức thấp bao gồm tất cả các công cụ cần thiết đề sử dụng mô hình hỗn hợp Gauss – GMM), các chức năng mức cao dành cho việc nhận diện giọng nói được cài đặt trong gói LIA-RAL. Chính vì điều này hầu hết các hoạt động của người sử dụng đều thông qua LIA-RAL. Mã trong gói LIA-RAL được chia thành hai loại:

* Thứ nhất là thư viện mức cao (có tên LIA\_SpkTools), chứa các giải thuật huấn luyện và các hàm thống kê được áp dụng trên các véc tơ đặc trưng.
* Thứ hai là các bộ công cụ LIA\_SpkSeg, LIA\_SpkDet và LIA\_Utils.

Bộ thư viện ALIZE chỉ hỗ trợ các công cụ cần thiết để xây dựng một hệ thống xác minh/xác định, phân lớp,… không phải là một hệ thống hoàn chỉnh. Vì vậy, việc xây dựng nên hệ thống phân lớp đòi hỏi việc nghiên cứu tài liệu, tập tin cấu hình có sẵn của thư viện, xác định tính năng của mỗi chương trình con để từ đó giải quyết các bài toán cụ thể.

****

Hình 2.1. Thành phần của gói công cụ ALIZE

**Cụ thể về các bộ công cụ trong gói LIA\_RAL**:

* *LIA\_Utils:* công cụ cần thiết để xử lý tất cả các định dạng được sử dụng trong ALIZE: mô hình GMM, các thông số, ...
* *LIA\_SpkDet:* một tập hợp các công cụ để làm tất cả các nhiệm vụ được yêu cầu bởi một hệ thống xác thực sinh trắc học: huấn luyện mô hình (tiếng nói / từ), các thông số tiêu chuẩn hóa, chuẩn hóa điểm số, …
* *LIA\_SpkSeg:* một công cụ mới để phát hiện tiếng nói.
  1. Sử dụng SPro và ALIZE trong phân lớp làn điệu

**Tính kết quả**

**nhận dạng**

Huấn luyện mô hình đích (gmm)

**Train**

**Target**



Huấn luyện mô hình nền

Chuẩn hóa đặc trưng

Loại bỏ

khoảng lặng

Trích chọn đặc trưng

**NormFeat**

**TrainWorld**

**Compute**

**Test**

**Energy Detector**

**sfbcep**

**SPro**

**LIA\_SpkDet**

Hình 2.2. Sơ đồ sử dụng các công cụ của SPro và ALIZE trong phân lớp làn điệu[6]

Hình 2.2 mô tả các bước và công cụ liên kết tại mỗi bước, dưới đây là trình bày các bước như sau:

* + 1. Trích chọn đặc trưng

Bộ công cụ SPro sẽ đọc một file âm thanh đầu vào, xử lý nó và trích xuất các vectơ đặc trưng. Kết quả đầu ra sẽ là đầu vào cho hệ thống LIA\_SpkDet (ở trong đồ án này em sử dụng bộ công cụ LIA\_SpkDet trong gói LIA\_RAL của ALIZE)

Các đặc trưng được trích chọn thực hiện bởi công cụ dựa trên phân tích filter –bank, sử dụng *sfbcep* trong *SPro*. Việc thực hiện dựa trên câu lệnh sau:

*sfbcep [option] inputfile outputfile*

Các tùy chọn(option) như sau:

*-F, --format=str*

Xác định sóng đầu vào của file. Định dạng hiển thị là ‘PCM16’, ‘wave’ hoặc ‘sphere’. Mặc định là: ‘PCM16’

*-f, --sample-rate=f*

Thiết lập sóng đầu vào tỉ lệ tần sô cho ‘PCM16’. Mặc định là 8kHz

*-x, --channel=n*

Thiết lập đặc trưng cho các kênh. Mặc định là 1

*-B, --swap*

Trao đổi các mẫu sóng đầu vào

*-I, --input-bufsize=n*

Thiết lập bộ đệm đầu vào kích cỡ tới n kbytes. Mặc định: 10MB

*-O, --output-bufsize=n*

Thiết lập bộ đệm đầu vào kích cỡ tới n kbytes. Mặc định: 10MB

*-k, --pre-emphasis=f*

Thiết lập hệ số tiền nhấn mạnh đến f. Mặc định: 0.95

*-l --length=f*

Thiết lập các phân tích dài khung tới *f ms.* Mặc định: 20.0 ms.

*-d, --shift=f*

Thiết lập khoảng thời gian giữa hai khung liên tiếp tới ms.

Mặc định là: 10ms

*-p, --num-ceps=n*

Thiết lập số lượng đầu ra hệ số Cepstral đến n. Mặc định là 12.

*-e, --energy*

Thêm năng lượng cho vectơ đặc trưng

*-w, --window=str*

Xác định các cửa sổ dạng sóng. Cửa sổ có sẵn là ‘Hamming’, ‘Hanning', `Blackman' hay không có gì. Mặc định là của số ‘Hamming’

*-n, --num-filters=n*

Thiết lập số kênh trong filter bank. Mặc định là: 24

*-a, --alpha=f*

Thiết lập ràng buộc tần số thấp đến f Hz. Mặc đinh: 0 Hz

*-u, --freq-max=f*

Thiết lập tần số giới hạn trên đến f Hz. Mặc định: Niquist.

*-b, --fft-length=n*

Thiết lập độ dài FFT tới n mẫu. Mặc đinh: 512

*-D, --delta*

Thêm các dẫn xuất đầu tiên đến các vectơ đặc trưng

*-A, --acceleration*

Thêm các dẫn xuất tự thứ hai với vectơ đặc trưng. Yêu cầu *`--delta'.*

*-v, --verbose*

Bật chế độ verbose

*-h, --help*

In thông báo và giúp các công cụ thoát

*-V, --version*

In thông tin phiên bản và thoát..

*Inputfile*: Đầu vào là đường dẫn đến file tín hiệu âm thanh

*Outputfile:* Đầu ra của file âm thanh

* + 1. Loại bỏ khoảng lặng

Khi tất cả các vec-tơ đặc trưng được tính toán, một bước rất quan trọng là quyết định vectơ đó sử dụng được hay không. Một cách để phát hiện ra vấn đề đó là so sánh với những khung có khoảng lặng. Vì vậy, để chọn các khung tương ứng với tín hiệu, sử dụng một phương pháp tiếp cận dựa trên sự phân bố năng lượng của mỗi tín hiệu. Để chuẩn hóa các hệ số năng lượng, chạy lệnh *NormFeat* từ thư viện *LIA SpkDet*. Câu lệnh thực hiện như sau:

*NormFeat [option] --inputFeatureFileName inputfile*

*NormFeat* nhằm mục đích xử lý tín hiệu đầu vào liên quan đến đặc trưng bằng cách áp dụng bất kỳ tiêu chuẩn hóa

*inputfile:* là tên của file cần làm việc, là một danh sách, có phần mở rộng là *.lst*

*options:* là các tùy chọn.

Giá trị tùy chọn đượcmô tả ở bảng dưới đây:

Bảng 2.1: Mô tả tùy chọn sử dụng chuẩn hóa năng lượng

|  |  |  |
| --- | --- | --- |
| **Tùy chọn** | **Giá trị** | **Mô tả** |
| mode | .norm | Xác định các loại tiêu chuẩn hóa( Định mức đặc trưng bình thường là phân bố gaussion trên 1,0) |
| vectSize | 60 | Xác định kích thước của các đặc trưng |
| loadFeatureFileFormat  saveFeatureFileFormat | SPRO4 | Định dạng gọi, lưu các đặc trưng của file |
| loadFeatureFileExtension | .prm | Định dạng phần mở rộng để gọi đặc trưng của file (tương ứng với đầu ra của các bước trích chọn đặc trưng) |
| saveFeatureFileExtension | .enr.prm | Định dạng phần mở rộng để lưu đặc trưng của file |
| featureFilesPath | ./prm/ | Định dạng đường dẫn để lưu và gọi file đặc trưng |
| writeAllFeatures | True | Khi thiết lập đúng, tất cả các tính năng về tập tin ban đầu được ghi |

Sau đó, để chọn các khung với năng lượng cao nhất, sử dụng lệnh *EnergyDetector* của thư viện *LIA SpkDet*. Câu lệnh thực hiện như sau:

*EnergyDetector.exe [options] --inputFeatureFileName inputfile*

*EnergyDetector* nhằm mục đích phân tích thành phần năng lượng đặc trưng đầu vào bằng cách cung cấp nhãn năng lượng đầu ra ở mức cao. Đây là một kiểu phát hiện có tiếng nói/ không có tiếng nói

*Inputfile*: là tên của đặc trưng mà năng lượng đã được tiêu chuẩn hóa.

*options:* là các tùy chọn.

Giá trị tùy chọn đượcmô tả ở bảng dưới đây:

Bảng 2.2: Mô tả tùy chọn sử dụng phát hiện tín hiệu tiếng nói

|  |  |  |
| --- | --- | --- |
| **Tùy chọn** | **Giá trị** | **Mô tả** |
| labelOutputFrames | Speech | Nhãn ảnh hưởng đến khung được chọn |
| LabelFilesPath | ./lbl | Xác định đường dẫn lưu trữ file |
| saveLabelFileExtension | .lbl | Lưu trữ các file mở rộng |
| Alpha | 0.0 | Tỷ lệ khung hình được chọn trong gaussian trung tâm |
| nbTrainIt | 10 | Số bước lặp của EM để ước lượng năng lượng phân phối |
| baggedFrameProbabilityInit | 0.001 | Xác định tỷ lệ giữa các khung hình được lựa chọn và tổng số các khung được sử dụng để khởi tạo |

* + 1. Chuẩn hóa đặc trưng

Các vectơ tham số được chuẩn hóa để phù hợp với trung bình bằng không và các thuộc tính không phù hợp. Giá trị trung bình và phương sai được sử dụng cho chuẩn hóa được tính bằng tập tin trên tổng số các khung sau khi xử lý loại bỏ khung.

Việc chuẩn hóa đặc trưng cũng giống như sử dụng chuẩn hóa năng lượng. Chỉ khác một điểm khi chạy lệnh *NormFeat* là gọi nhãn file. Vì vậy, để thực hiện các chuẩn hóa năng lượng, *NormFeat* chỉ sử dụng các khung với năng lượng cao nhất. Bảng dưới đề cập đến những thay đổi so với chuẩn hóa năng lượng:

Bảng 2.3: Mô tả tùy chọn chuẩn hóa đặc trưng

|  |  |  |
| --- | --- | --- |
| **Tùy chọn** | **Giá trị** | **Mô tả** |
| saveFeatureFileExtension | .norm.prm | Xác định định dạng để lưu file |
| labelSelectFrames | speech | Xác định nhãn để làm việc |
| LabelFilePath | ./prm | Đường dẫn để lưu file |

* + 1. Huấn luyện mô hình nền

Mục đích của bước này là mô hình nền phổ quát (UBM) được tạo ra, sử dụng tất cả các dữ liệu mà chúng ta đã chuẩn hóa trước đó. Mô hình GMM được ước tính bằng cách sử dụng thuật toán EM (tối đa hóa kỳ vọng).

Huấn luyện mô hình nền, thực thi bởi lệnh *TrainWorld* của thư viện *LIA\_SpkDet.* Câu lệnh được thực hiện như sau:

*TrainWorld [options] --inputFeatureFileName inputfile*

*--outputWorldFilename worldfile*

*TrainWorld:* học mô hình GMM thông qua thuật toán EM.

*inputfile:* là tên của các chuẩn hóa đặc trưng để làm việc với nó và nó có thể là một danh sách mở rộng, có đuôi là .*lst*

*worldfile:* là tên file kết quả từ mô hình.

*options:* là các tùy chọn.

Giá trị tùy chọn đượcmô tả ở bảng dưới đây:

Bảng 2.4: Mô tả tùy chọn sử dụng trong huấn luyện mô hình nền

|  |  |  |
| --- | --- | --- |
| **Tùy chọn** | **Ví dụ** | **Mô tả** |
| distribType | GD | Xác định kiểu của thuộc tính |
| saveLabelFileExtension | .gmm | Định dạng để lưu thuộc tính |
| loadFeatureFileExtension | .norm.prm | Định dạng đầu vào của file |
| MixtureDistribCount | 256 | Xác định số thuộc tính gaussion trong hỗn hợp |
| mixtureFilePath | ./gmm | Đường dẫn để lưu thư mục file |
| baggedFrameProbabilityInit | 0.0005 | Xác định tỷ lệ giữa các khung được lựa chọn và tổng số các khung được sử dụng để khởi tạo. |
| baggedFrameProbability | 0.01 | Xác định tỷ lệ giữa các khung hình được lựa chọn và tổng số các khung được sử dụng để huấn luyện |
| nbTrainIt | 25 | Số lần lặp của EM liên quan đến baggedFrameProbabilityInit |
| nbTrainFinalIt | 4 | Số lần lặp của EM không có baggedFrameProbability |
| featureServerMask | 0-59 | Được sử dụng để chọn một tập hợp các đặc trưng trong vectơ |
| vectSize | 60 | Xác định kích thước của các đặc trưng |
| NormalizeModel | true | Áp dụng N(0, 1) sắp xếp ở cuối quá trình huấn luyện |

* + 1. Huấn luyện mô hình đích

Huấn luyện mô hình đích, chạy lệnh *TrainTarget* của gói *LIA\_SpkDet.* Sử dụng câu lệnh như sau:

*TrainTarget [options] --targetIdList inputfile*

*--inputWorldFilename worldfile*

*TrainTarget* nhằm mục đích huấn luyện đích người nói bằng cách làm phù hợp mô hình từ thông qua phương pháp MAP.

*Inputfile* là danh sách đầu, chính là danh sách các tên file đưa vào huấn luyện

*Worldfile: world -- debug false –verbose true*

*options:* là các tùy chọn.

Giá trị tùy chọn đượcmô tả ở bảng dưới đây:

Bảng 2.5: Mô tả tùy chọn trong huấn luyện mô hình đích

|  |  |  |
| --- | --- | --- |
| **Tùy chọn** | **Giá trị** | **Mô tả** |
| baggedFrameProbability | 0.1 | Xác định tỷ lệ giữa các khung được lựa chọn và tổng số các khung được sử dụng để thích ứng |
| nbTrainIt | 5 | Số lần lặp EM liên quan tới baggedFrameProbability |
| nbTrainFinalIt | 1 | Số lần lặp EM không liên quan tới baggedFrameProbability |
| targetIdList | Ndx | Danh sách đầu vào: mô hình trên cột đầu tiên, nhập vào tên tập tin đặc trưng về những người khác nhau |
| MAPAlgo | MAPOccDep | Xác định phương pháp thích ứng để sử dụng: MAPOccDep hoặc MAPConst |
| MAPRegFactor | 10 | Tham số được sử dụng bởi những kỹ thuật thích ứng MAPOccDep |

Phương thức MAP sử dụng là phương pháp tiếp cận MAPOCCDep: biến ngẫu nhiên để ước lượng được tính bằng sự kết hợp tuyến tính các giá trị trong mô hình nền với dữ liệu trên thuật toán EM. Phương pháp này sẽ đưa vào một xác suất hậu nghiệm cho mỗi Gaussion. Trọng số của sự kết hợp này được cung cấp bởi các tùy chọn MAPREGFactor r

* + 1. Thử nghiệm

Mục tiêu của bước này là tính toán điểm số cho từng thử nghiệm của các đặc trưng vectơ để đưa ra một mô hình đích và mô hình nền (số điểm là xác suất tính được qua thử nghiệm từng file âm thanh). Để tính toán số điểm này, chỉ xem xét 10 thuộc tính Gaussian đầu tiên của mô hình. Câu lệnh được thực hiện như sau:

*ComputeTest [options] --ndxFileName inputfile*

*--worldModelName worldfile*

*--outputFile outputfile*

*ComputeTest:* đưa ra điểm số liên quan đến một phần thử nghiệm và mô hình đích.

*Inputfile:* là đưa ra danh sách các file thử nghiệm

*Worldfile:* là tên của mô hình nền

*Outputfile:* là kết quả ghi ra tập tin

*options:* là các tùy chọn.

Giá trị tùy chọn đượcmô tả ở bảng dưới đây:

Bảng 2.6: Mô tả tùy chọn tính điểm số của mô hình GMM

|  |  |  |
| --- | --- | --- |
| **Tùy chọn** | **Giá trị** | **Mô tả** |
| topDistribsCount | 10 | Số Gaussians sử dụng để tính toán điểm số |
| computeLLKWithTopDistribs | COMPLETE | Tính toán với phân phối đầu |
| Gender | M | Giới tính của tập tin ndx |
| loadMixtureFileExtension | .gmm | Xác định phần mở rộng để gọi mô hình |
| loadFeatureFileExtension | .norm.prm | Xác định phần mở rộng để gọi đặc trưng |
| mixtureFilesPath | ./gmm | Xác định đường dẫn để gọi hỗn hợp |
| outputFile | test1.res | Kết quả điểm số tập tin |

CHƯƠNG 3. TRIỂN KHAI HỆ THỐNG THỬ NGHIỆM PHÂN LỚP LÀN ĐIỆU CHÈO VÀ DÂN CA QUAN HỌ

1. 1. Sơ đồ tổng quan quá trình xây dựng hệ thống phân lớp làn điệu chèo và dân ca quan họ

Bước đầu tiên của việc xây dựng hệ thống thử nghiệm là việc chuẩn bị dữ liệu, dữ liệu này là các file âm thanh có định dạng là .wav, mono, với tần số 16000Hz, được cắt bỏ phần nhạc dạo bằng phần mềm Wavepad Sound Editor. Danh sách các làn điệu sẽ được trình bày cụ thể ở mục 3.2 của chương này. Sau khi chạy chương trình thử nghiệm trên bộ công cụ ALIZE sẽ cho ra số (Score) của file âm thanh đưa vào thử nghiệm với danh sách các làn điệu đã được huấn luyện. File âm thanh này sẽ được xếp vào làn điệu nào được đánh điểm cao nhất.

Trong đồ án này việc triển khai hệ thống thử nghiệm sẽ được tiến hành theo 2 phương pháp nhưng điểm khác biệt chủ yếu của 2 phương pháp này nằm ở tập dữ liệu huấn luyện và thử nghiệm.

* + 1. Tiến hành thử nghiệm với dữ liệu không nằm trong tập dữ liệu huấn luyện

|  |
| --- |
| tổng hợp  kết quả  Tổng hợp kết quả  Đưa những file thử nghiệm có kết quả phân lớp theo thể loại đúng vào nhận dạng theo làn điêụ cụ thể  chuẩn hóa  Chia dữ liệu thành 5 phần bao gồm 4 phần để huấn luyện và 1 phần để thử nghiệm  Chuẩn bị dữ liệu  Tiến hành huấn luyện với 4 phần dữ liệu bên trên  huấn luyện  Thử nghiệm với 1 phần dữ liệu còn lại |

Hình 3.1. Hệ thống thử nghiệm với dữ liệu không nằm trong tập huấn luyện

* + 1. Tiến hành thử nghiệm với dữ liệu lấy từ tập huấn luyện

Hệ thống thử nghiệm này được xây dựng dựa trên ý tưởng, khi chúng ta khi đã biết về một bài hát thì không cần nghe hết bài hát vẫn có thể nhận biết ra được tên của bài hát đó.

Điểm khác của phương pháp này so với phương pháp đầu tiên đó là với tập dữ liệu mẫu ban đầu sẽ không chia ra thành 2 phần dữ liệu huấn luyện và thử nghiệm mà thay vào đó tất cả sẽ được đưa vào huấn luyện.

Dữ liệu thử nghiệm sẽ được tạo ra bằng việc cắt nhỏ từng file âm thanh của tập dữ liệu mẫu ban đầu theo độ dài định trước. Cụ thể, trong đồ án này em sẽ tiến hành thử nghiệm với các độ dài thời gian lần lượt là 4s, 6s, 8s, 10s, 12s, 14s, 16s và để nguyên toàn bộ file gốc.

Hệ thống được mô tả như hình vẽ dưới đây:

chuẩn hóa

Tiến hành huấn luyện với toàn bộ tập dữ liệu mẫu

Tập dữ liệu mẫu

cắt ngắn từng file theo độ dài định trước

chuẩn hóa

Tiến hành thử nghiệm

Tập dữ liệu thử nghiệm

tổng hợp

kết quả

Đưa những file thử nghiệm có kết quả phân lớp sơ bộ đúng vào nhận dạng theo làn điêụ cụ thể

Tổng hợp kết quả

Hình 3.2. Thử nghiệm hệ thống phân lớp với các dữ liệu dùng để thử nghiệm lấy từ tập huấn luyện

* 1. Chuẩn bị dữ liệu

File nhạc sau khi được tải về từ trên mạng internet sẽ cắt và chuyển về định dạng file Wav. Với việc tiến hành phân lớp sơ bộ (xác định một làn điệu là chèo hay quan họ) thì mỗi thể loại sẽ được huấn luyện với 25\*16 = 400 file âm thanh, sau đó khi tiến hành phân lớp cụ thể cho từng làn điệu thì mỗi làn điệu sẽ được huấn luyện với 16 file âm thanh.

Kết quả đã tạo được cơ sở dữ liệu âm nhạc phục vụ cho quá trình phân lớp làn điệu chèo và dân ca quan họ. Cơ sở dữ liệu này được đặt trong thư mục wav: gồm có 1000 file âm thanh với tổng kích thước là 1,8 GB.

Với mỗi làn điệu, định dạng file được được lưu dựa theo tên viết tắt được đề cập trong hai bảng liệt kê danh sách làn điệu chèo (bảng 3.1) và làn điệu dân ca quan họ (bảng 3.2). Định dạng file được tổ chức như sau:

CH/QH\_Chữ cái bắt đầu của từng chữ trong tên làn điệu-Thứ tự trong của file gốc trong danh sách các file của một làn điệu-Vị trí của phần được cắt ra trong file gốc.

Trong đó:

- CH: Ký hiệu là file chèo

- QH: Ký hiệu là file dân ca quan họ

Ví dụ: làn điệu Bình Thảo có các file mp3 gốc là:

- 1\_BinhThao-ThuHuyen\_ThanhTam.mp3

- 2\_BinhThao-MinhToan.mp3

- 3\_BinhThao-DoanThanhBinh.mp3

Từ file 1\_BinhThao-ThuHuyen\_ThanhTam.mp3, ta cắt ra được 3 file wav, thì định dạng tên của từng file sẽ là: CH\_BT-01-01, CH\_BT-01-02, CH\_BT-01-03. Tương tự với các file khác ta sẽ được là CH\_BT-02-01, CH\_BT-02-02,…

Đối với hệ thống có dữ liệu thử nghiệm không nằm trong tập huấn luyện thì dữ liệu được mô tả ở trên được chia thành 2 tập:

- Tập dữ liệu để huấn luyện: gồm dữ liệu của 50 làn điệu (25 làn điệu chèo, 25 làn điệu quan họ), mỗi làn điệu sẽ có 16 file được đưa vào huấn luyện. tổng cộng là 800 file.

- Tập dữ liệu để thử nghiệm: Lấy 4 file trong mỗi làn điệu để làm dữ liệu thử nghiệm, tổng cộng là 200 file

Các file được dùng để huấn luyện và thử nghiệm được luân phiên đổi chỗ cho nhau. Ví dụ 20 file của một làn điệu sẽ được chia ra làm 5 phần: phần 1, phần 2, phần 3, phần 4, phần 5. Đầu tiên sẽ lấy phần 1 làm dữ liệu thử nghiệm, còn các phần 2, 3, 4, 5 sẽ đưa vào huấn luyện, sau khi thu được kết quả thì lại lấy phần 2 làm test, các phần 1, 3, 4, 5 đưa vào huấn luyện,… Thực hiện liên tục cho đến khi tất cả các file đều đã được thử nghiệm, thì tiến hành tổng hợp kết quả.

Thông tin của toàn bộ các làn điệu được mô tả trong bảng dưới đây:

Bảng 3.1: Thông tin về 25 làn điệu chèo

|  |  |  |
| --- | --- | --- |
| STT | Tên làn điệu | Tên viết tắt |
| 01 | Bình thảo | CH\_BT |
| 02 | Cấm giá | CH\_CG |
| 03 | Chinh phụ | CH\_CP |
| 04 | Chức cẩm hồi văn | CH\_CCHV |
| 05 | Con nhện giăng mùng | CH\_CNGM |
| 06 | Đò đưa | CH\_DD |
| 07 | Đào liễu | CH\_DL |
| 08 | Đào lý một cành | CH\_DLMC |
| 09 | Duyên phận phải chiều | CH\_DPPC |
| 10 | Đường trường bắn thước | CH\_DTBT |
| 11 | Đường trường tiếng đàn | CH\_DTTD |
| 12 | Đường trường thu không | CH\_DTTK |
| 13 | Đường trường trên non | CH\_DTTN |
| 14 | Đường trường vị thủy | CH\_DTVT |
| 15 | Du xuân | CH\_DXN |
| 16 | Dương xuân | CH\_DX |
| 17 | Lới lơ | CH\_LL |
| 18 | Luyện năm cung | CH\_LNC |
| 19 | Nón thúng quai thao | CH\_NTQT |
| 20 | Quá giang | CH\_QG |
| 21 | Quạt màn | CH\_QM |
| 22 | Quân tử vu dịch | CH\_QTVD |
| 23 | Sắp qua cầu | CH\_SQC |
| 24 | Tò vò | CH\_TV |
| 25 | Xẩm xoan | CH\_XX |

Bảng 3.2: Thông tin về 25 làn điệu quan họ

|  |  |  |
| --- | --- | --- |
| STT | Tên làn điệu | Tên viết tắt |
| 01 | Buôn bấc buôn dầu | QH\_BBBD |
| 02 | Bèo dạt mây trôi | QH\_BDMT |
| 03 | Ba quan | QH\_BQ |
| 04 | Còn duyên | QH\_CD |
| 05 | Cổ tay vừa trắng vừa tròn | QH\_CTVTVT |
| 06 | Cây trúc xinh | QH\_CTX |
| 07 | Chuông vàng gác cửa tam quan | QH\_CVGCTQ |
| 08 | Đôi bên bác mẹ cùng già | QH\_DBBMCG |
| 09 | Đối ca sông cầu | QH\_DCSC |
| 10 | Đêm qua nhớ bạn | QH\_DQNB |
| 11 | Gió đưa cây cải | QH\_GDCC |
| 12 | Giăng thanh gió mát | QH\_GTGM |
| 13 | Lý giao duyên | QH\_LGD |
| 14 | Lòng vẫn đợi chờ | QH\_LVDC |
| 15 | Mười nhớ | QH\_MN |
| 16 | Nguyệt gác mái đình | QH\_NGMD |
| 17 | Nhớ mãi khôn nguôi | QH\_NMKN |
| 18 | Người ơi đến hẹn lại về | QH\_NODHLV |
| 19 | Nhất quế nhị lan | QH\_NQNL |
| 20 | Se chỉ luồn kim | QH\_SCLK |
| 21 | Súc miệng ấm đồng | QH\_SMAD |
| 22 | Tìm em trong chiều hội Lim | QH\_TETCHL |
| 23 | Tương phùng tương ngộ | QH\_TPTN |
| 24 | Trên rừng ba mươi sáu thứ chim | QH\_TRBMSTC |
| 25 | Vui bốn mùa | QH\_VBM |

* 1. Phân lớp làn điệu sử dụng ALIZE

Dù là việc phân lớp sơ bộ hay phân lớp cụ thể theo từng làn điệu thì cũng cần thực hiện các bước được mô tả bên dưới đây:

Tạo thư mục làm việc

Tính tham số MFCC

Dò tìm năng lượng

Phát hiện âm thanh trong tín hiệu

Chuẩn hóa các tham số của tín hiệu

Chuẩn hóa TrainWorldInit

Chuẩn hóa TrainWorldFinal

Huấn luyện GMM cho từng thể loại/làn điệu

Tính điểm số của từng file đưa vào thử nghiệm

Hình 3.3. Các bước xây dựng hệ phân lớp sử dụng SPro & ALIZE

*Sau đây là hướng dẫn chi tiết các bước thực hiện lệnh chạy của chương trình:*

* + 1. Bước 1: Tạo thư mục làm việc

Tạo thư mục Alize chứa các file và thư mục làm việc. Bên trong thư mục này ta tạo các thư mục sau:

* cfg: chứa những file config.
* gmm, gmm\_ch, gmm\_qh: chứa những file huấn luyên GMM cho từng thể loại và từng làn điệu.
* lbl: chứa những file đánh dấu những đoạn có âm thanh trong dữ liệu đầu vào(bao gồm cả dữ liệu huấn luyện và thử nghiệm).
* lst, lst\_ch, lst\_qh: chứa danh sách các file đầu vào.
* ndx, ndx\_ch, ndx\_qh: chứa tên của những file sẽ được đưa vào huấn luyện GMM.
* prm: chứa những file đã được trích chọn đặc trưng.
* test, test\_ch, test\_qh: chứa những file sẽ đưa vào thử nghiệm.
* res, res\_ch, res\_qh: chứa kết quả của những file thử nghiệm.
* rpt: nơi lưu file báo cáo của toàn bộ quá trình thử nghiệm.
* wav: nơi lưu trữ toàn bộ những file âm thanh đầu vào.
  + 1. Bước 2: Tính tham số MFCC

Tạo ra các vectơ đặc trưng từ các tín hiệu âm thanh, sử dụng công cụ *sfbcep* để tính các tham số MFCC. Điều kiện cần thiết để tạo được tham số MFCC là phải chuẩn hóa các file .wav thành các file .prm. Tạo thư mục có tên là “prm“ để chương trình chạy ghi lại các file .prm

* Đầu vào: file âm thanh có định dạng \*.wav
* Đầu ra: file prm có chứa tham số MFCC

Sử dụng câu lệnh:

*./sfbcep.exe -F PCM16 -f 16000 -p 19 -e -D -A wav/CH\_BT-01-01.wav prm/CH\_BT-01-01.prm*

Ý nghĩa của câu lệnh như sau:

*-FPCM16*: sử dụng định dạng hiển thị sóng đầu vào của file là : PCM16

*-f 16000*: sử dụng tần số cho sóng đầu vào là: 16000Hz

*-p 19*: có hệ số Cepstral là 19

*-e*: sử dụng thêm năng lượng cho vectơ đặc trưng

*-D*: Thêm các dẫn xuất đầu tiên đến các vectơ đặc trưng

*-A*: Thêm các dẫn xuất thứ hai với vectơ đặc trưng

🡺 Kết quả : sinh ra file *CH\_BT-01-01.prm*

* + 1. Bước 3: Dò tìm năng lượng
* Đầu vào: là những file .prm tạo được từ bước 2
* Đầu ra: file *.enr.prm* được lưu trong thư mục *prm*

Sử dụng câu lệnh:

*./NormFeat.exe --config ./cfg/NormFeat\_energy.cfg --inputFeatureFilename “CH\_BT-01-01”*

🡺 Kết quả : sinh ra file *CH\_BT-01-01.enr.prm*

* + 1. Bước 4: Phát hiện âm thanh trong tín hiệu

Mục đích để phát hiện những đoạn có âm thanh

* Đầu vào: là những file .prm được tạo ra từ bước 2
* Đầu ra: file *.lbl* được lưu trong thư mục *lbl*

Sử dụng câu lệnh:

*./EnergyDetector.exe --config ./cfg/EnergyDetector.cfg --inputFeatureFilename “CH\_BT-01-01”*

🡺 Kết quả : sinh ra file *CH\_BT-01-01.lbl*

Nội dung của file *CH\_BT-01-01.lbl*:

*0 0 speech*

*0.18 0.18 speech*

*0.2 0.21 speech*

*0.23 0.25 speech*

*…*

*43.87 43.88 speech*

*43.97 43.97 speech*

*44.02 44.02 speech*

* + 1. Bước 5: Chuẩn hóa các tham số của tín hiệu
* Đầu vào: là những file .prm tạo được từ bước 2
* Đầu ra: file *.norm.prm* được lưu trong thư mục prm

Sử dụng câu lệnh:

*./NormFeat.exe --config ./cfg/NormFeat.cfg --inputFeatureFilename “CH\_BT-01-01”*

🡺 Kết quả : sinh ra file *CH\_BT-01-01.norm.prm*

* + 1. Bước 6: Tạo sanh sách file wav
* Đầu vào: là những file trong thư mục wav
* Đầu ra: tạo ra trong thư mục lst 2 file: file all.lst chứa đầy đủ thông tin của file âm thanh, all1.lst chỉ chứa tên của file âm thanh

Ví dụ: Nội dung của file all.lst

*CH\_BT-01-01.wav*

*CH\_BT-01-02.wav*

*CH\_BT-01-03.wav*

*CH\_BT-02\_01.wav*

*……………*

*QH\_VBM-05-04.wav*

*QH\_VBM-06-02.wav*

*QH\_VBM-06\_03.wav*

*QH\_VBM -06\_04.wav*

Nội dung của file all1.lst:

*CH\_BT-01-01*

*CH\_BT-01-02*

*CH\_BT-01-03*

*CH\_BT-02\_01*

*……………*

*QH\_VBM-05-04*

*QH\_VBM-06-02*

* + 1. Bước 7: Phân chia dữ liệu thành hai tập huấn luyện và thử nghiệm
* Đầu vào: là file all1.lst tạo được ở Bước 6.
* Đầu ra: là những file có định dạng TEN\_THE\_LOAItrain.lst và TEN\_THE\_LOAItest.lst trong thư mục lst với mục đích thực hiện phân lớp theo thể loại hoặc TEN\_LAN\_DIEUtrain.lst và TEN\_LAN\_DIEUtest.lst với mục đích thực hiện phân lớp theo làn điệu.

Ví dụ: File *CHtrain.lst* trong thư mục lst sẽ có nội dung như sau:

*CH\_BT-02-02*

*CH\_BT-02-03*

*CH\_BT-03-01*

*CH\_BT-03-02*

*CH\_BT-03-03*

*…*

*CH\_XX-10-01*

*CH\_XX-10-02*

* + 1. Bước 8: Tạo file World.lst và World.lenght
* ***World.lst:*** nội dung của file này là chỉ đường dẫn đến tất cả các danh sách file âm thanh được dùng để huấn luyện (chính là file *TEN\_THE\_LOAItrain.lst* hoặc TEN\_LAN\_DIEUtrain.lst được tạo ra ở Bước 7). Mỗi tên file được lưu trên một dòng.

Ví dụ: Nội dung của file World.lst trong thư mục lst

*D:\Alize\lst\CHtrain.lst*

*D:\Alize\lst\QHtrain.lst*

* ***world.weight***: file trọng số, với việc phân theo 2 lớp(chèo hoặc quan họ) thì trọng số sẽ là 0.5.

Ví dụ: file World.weight trong thư mục lst

*0.5*

*0.5*

* + 1. Bước 9: Chuẩn hóa TrainWorldInit
* Đầu vào: file *world.lst* và file *world.weight*.
* Đầu ra: file *worl\_init.gmm, world\_initinit.gmm.*

*Sử dụng câu lệnh:*

*./TrainWorld.exe --config ./cfg/TrainWorldInit.cfg --inputStreamList ./lst/world.lst --weightStreamList ./lst/world.weight --outputWorldFilename world\_init --debug false --verbose true*

* + 1. Bước 10: Chuẩn hóa TrainWorldFinal
* Đầu vào: file *world.lst* và file *world.weight*
* Đầu ra: file *world.gmm*

Sử dụng câu lệnh:

*./TrainWorld.exe --config ./cfg/TrainWorldFinal.cfg --inputStreamList ./lst/world.lst --weightStreamList ./lst/world.weight --outputWorldFilename world --inputWorldFilename world\_init --debug false --verbose true*

* + 1. Bước 11: Tạo file chứa danh sách các file được huấn luyện
* Đầu vào: là những file có định dạng TEN\_THE\_LOAItrain.lst trong thư mục lst đối với việc phân lớp sơ bộ hoặc TEN\_LAN\_DIEUtrain.lst trong thư mục lst\_ch và lst\_qh đối với việc phân lớp theo làn điệu cụ thể.
* Đầu ra: là những file có định dạng TEN\_THE\_LOAI.ndx trong thư mục ndx với việc phân lớp sơ bộ hoặc TEN\_LAN\_DIEU.ndx trong thư mục ndx\_ch và ndx\_qh đối với việc phân lớp theo làn điệu cụ thể.

Ví dụ: file ch.ndx sẽ cấu trúc như sau:

*CH\_gmm CH\_BT-01-01 CH\_BT-01-02 CH\_BT-01-03 CH\_BT-02-01 CH\_BT-03-03 CH\_BT-04-02 CH\_BT-04-03 CH\_BT-05-02 CH\_BT-05-03 CH\_BT-06-01 CH\_BT-06-02 CH\_BT-07-01 CH\_BT-07-02 CH\_BT-08-01 CH\_BT-08-02 CH\_BT-08-03 CH\_CCHV-03-04 CH\_CCHV-04-02 CH\_CCHV-04-03 CH\_CCHV-05-02 CH\_CCHV-06-04*

*........* *CH\_XX-07-01 CH\_XX-07-02 CH\_XX-08-01 CH\_XX-08-02 CH\_XX-09-01 CH\_XX-09-02 CH\_XX-10-01 CH\_XX-10-02*

* + 1. Bước 12: Huấn luyện GMM
* Đầu vào: file *TEN\_THE\_LOAI.ndx* đối với việc phân lớp sơ bộ hoặc là file TEN\_LAN\_DIEU.ndx đối với việc phân lớp theo làn điệu cụ thể.
* Đầu ra: file *TEN\_THE\_LOAI\_gmm.gmm* trong thư mục gmm đối với việc phân lớp sơ bộ hoặc TEN\_LAN\_DIEU\_gmm.gmm trong thư mục gmm\_ch và gmm\_qh đối với việc phân lớp theo làn điệu cụ thể.

Câu lệnh thực hiện:

*./TrainTarget.exe --config ./cfg/target.cfg --targetIdList ./ndx/ch.ndx --inputWorldFilename world --debug false --verbose true*

🡺 Kết quả : sinh ra file *CH\_gmm*

* + 1. Bước 13: Tạo file thử nghiệm

Công việc này sẽ thực hiện khác nhau tùy thuộc vào từng bài toán là phân lớp sơ bộ hay phân lớp cụ thể theo từng làn điệu.

1. Phân lớp sơ bộ

* Đầu vào: những file có định dạng TEN\_THE\_LOAItest.lst trong thư mục lst và những file trong thư mục ndx.
* Đầu ra: là những file có định dạng TEN\_FILE\_AM\_THANH.ndx lưu trong thư mục test.

Ví dụ: nội dung của file CH\_BT-01-01.ndx trong quá trình tiến hành phân lớp sơ bộ là:

*CH\_BT-01-01 ch\_gmm qh\_gmm*

1. Phân lớp cụ thể

* Đầu vào: là file report.txt tổng hợp kết quả từ việc phân lớp theo thể loại, những file kết quả trong thư mục res và những file trong 2 thư mục ndx\_ch và ndx\_qh.
* Đầu ra: là những file có định dạng TEN\_FILE\_AM\_THANH.ndx lưu trong thư mục test\_ch hoặc test\_qh.

Ví dụ: nội dung của file CH-BT-01-01.ndx lưu trong thư mục test\_ch là:

*CH\_BT-01-01 ch\_bt\_gmm ch\_cchv\_gmm ch\_cg\_gmm ch\_cngm\_gmm ch\_cp\_gmm ch\_dd\_gmm ch\_dl\_gmm ch\_dlmc\_gmm ch\_dppc\_gmm ch\_dtbt\_gmm ch\_dttd\_gmm ch\_dttk\_gmm ch\_dttn\_gmm ch\_dx\_gmm ch\_dxn\_gmm ch\_ll\_gmm ch\_lnc\_gmm ch\_ntqt\_gmm ch\_qg\_gmm ch\_qm\_gmm ch\_qtvd\_gmm ch\_slc\_gmm ch\_sqc\_gmm ch\_tv\_gmm ch\_xx\_gmm*

* + 1. Bước 14: Tiến hành thử nghiệm
* Đầu vào: là những file trong thư mục test đối với phân lớp sơ bộ, là những file trong thư mục test\_ch và test\_qh đối với việc phân lớp theo làn điệu cụ thể.
* Đầu ra : ra những file có định dạng FILE\_AM\_THANH.res lưu trong thư mục res đối với phân lớp sơ bộ hoặc res\_ch và res\_qh đối với việc phân lớp theo từng làn điệu cụ thể.

Câu lệnh thực hiện:

*ComputeTest --config cfg/target\_seg\_male.cfg --ndxFilename ./test/CH\_BT-01-01.ndx --inputWorldFilename world --outputFilename ./res/CH\_BT-01-01.res*

🡺 Kết quả : sinh ra file *CH\_BT-01-01.res trong thư mục res*

Ví dụ: file kết quả *CH\_BT-01-01.res* trong thư mục res như sau:

*M ch\_gmm 1 CH\_BT-01-01 0.0751762*

*M qh\_gmm 0 CH\_BT-01-01 -0.103114*

Mỗi một file âm thanh sẽ ghi ra điểm số (score) tương ứng với N thể loại hoặc làn điệu. Điểm số của thể loại/làn điệu nào lớn nhất trong N thể loại/làn điệu chính là thể loại/làn điệu mà hệ thống cho ra kết quả cần tìm.

CHƯƠNG 4. KẾT QUẢ THỬ NGHIỆM

Hệ thống thử nghiệm được chạy trong hệ thống giả lập Linux – Linux Bash Shell của Window 10, trên máy tính Laptop 2.10 GHz Intel Pentium CPU với 6GB RAM. Các kết quả của đồ án dựa trên thử nghiệm với tập dữ liệu bao gồm 1000 file âm thanh, tương ứng với 25 làn điệu chèo và 25 làn điệu dân ca quan họ (mỗi làn điệu có 20 file âm thanh).

Với tập dữ liệu như trên em sử dụng hai phương pháp đánh giá:

* Thử nghiệm với dữ liệu không nằm trong tập dữ luyện huấn luyện: với tập dữ liệu ban đầu em sẽ tiến hành chia thành 5 phần, sau đó sẽ lấy 4 phần dữ liệu để đưa vào huấn luyện còn 1 phần sẽ dùng để thử nghiệm, như vậy sẽ có 800 file âm thanh dùng để huấn luyện và 200 file âm thanh dùng để thử nghiệm. Sau mỗi lần thử nghiệm em sẽ lại tiến hành đổi chỗ những file dùng để huấn luyện và thử nghiệm cho nhau, sao cho sau 5 lần thử nghiệm tất cả các file trong bộ dữ liệu mẫu đều được đưa vào huấn luyện và thử nghiệm. Kết quả cuối cùng là kết quả tổng hợp của 5 lần thử nghiệm đánh giá chéo bên trên.
* Thử nghiệm với dữ liệu lấy từ tập huấn luyện: khác với phương pháp trên, phương pháp này em sẽ đưa tất cả các file trong dữ liệu mẫu vào để huấn luyện còn dữ liệu thử nghiệm sẽ được tạo ra bằng cách cắt ngắn theo độ dài cố định từng file âm thanh đã đưa vào huấn luyện. Như vậy, dữ liệu huấn luyện sẽ bao gồm 1000 file âm thanh gốc, dữ liệu thử nghiệm sẽ bao gồm 1000 file âm thanh đã được cắt ngắn ra từ file gốc. Sau mỗi lần đánh giá chúng ta sẽ thu được kết quả đo độ chính xác của hệ thống đối với mỗi độ dài nhất định của file thử nghiệm.

Với mỗi lần thử nghiệm chúng ta sẽ thu được hai kết quả:

* *Kết quả phân lớp sơ bộ*: thể hiện độ chính xác của hệ thống khi xác định một làn điệu là làn điệu chèo hay quan họ.
* *Kết quả phân lớp cụ thể*: thể hiện độ chính xác của hệ thống khi xác định cụ thể tên của làn điệu.

Ngoài ra, để tăng hiệu quả của hệ thống phân lớp em đã tiến hành thử nghiệm trên nhiều hệ số thành phần Gauss khác nhau để lựa chọn ra hệ số đem lại kết quả phân lớp tốt nhất. Cụ thể, những giá trị của số thành phần Gauss lần lượt đưa vào thử nghiệm là 16, 32, 64, 128, 256, 512, 1024, 2048 và 4096.

Sau đây sẽ là kết quả thực nghiệm của hệ thống trong một số trường hợp:

* 1. Kết quả thử nghiệm với dữ liệu không nằm trong tập huấn luyện
     1. Kết quả thử nghiệm phân lớp sơ bộ

Tiến hành thử nghiệm với hệ số thành phần Gauss (M) là 512 ta thu được kết quả như sau:



Bảng 4.1. Ma trận nhầm lẫn phân lớp sơ bộ với dữ liệu thử nghiệm không nằm trong tập huấn luyện (M = 512)

Từ bảng 4.1 ta tính được độ chính xác của từng thể loại:

* Thể loại Chèo có độ chính xác là:
* Thể loại Dân ca quan họ có độ chính xác là: .
* Từ đó ta tính được độ chính xác trung bình :

**Nhận xét:**

* Với M = 512 thì độ chính xác của bài toán phân lớp sơ bộ đạt được tương đối cao, cụ thể là 87,70%.
* Các làn điệu chèo thì có độ nhận dạng chính xác trung bình trong bài toán sơ bộ tốt hơn so với các làn điệu quan họ.

Dưới đây sẽ là kết quả thử nghiệm với số các thành phần Gauss khác nhau:

Hình 4.1. Kết quả phân lớp sơ bộ khi thay đổi hệ số M với dữ liệu thử nghiệm không nằm trong tập huấn luyện

**Nhận xét:**

* Đối với việc phân lớp sơ bộ thì kết quả trả về với độ chính xác cao nhất với hệ thống có thành phần Gauss là 4096 (). Tuy nhiên sự chênh lệch về độ chính xác khi thử nghiệm với M = 4096 và M = 2048 là không quá nhiều, chỉ khác biệt
* Kết quả trả về có độ chính xác thấp nhất với hệ thống có số thành phần Gauss là 16 ().
  + 1. Kết quả thử nghiệm phân lớp cụ thể

1. Kết quả phân lớp làn điệu chèo

Tiến hành thử nghiệm với hệ số thành phần Gauss (M) bằng 512 ta thu được kết quả như sau:



Bảng 4.2. Ma trận nhầm lẫn nhận nhận dạng cụ thể làn điệu chèo với dữ liệu thử nghiệm không nằm trong tập huấn luyện (với M = 512)

Từ bảng 4.2 ta tính được độ chính xác của từng làn điệu như sau:

* Làn điệu “CH\_BT” có độ nhận dạng chính xác là:
* Làn điệu “CH\_CCHV” có độ nhận dạng chính xác là:
* Làn điệu “CH\_CG” có độ nhận dạng chính xác là:
* Làn điệu “CH\_CNGM” có độ nhận dạng chính xác là:
* Làn điệu “CH\_CP” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DD” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DL” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DLMC” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DPPC” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DTBT” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DTTD” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DTTK” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DTTN” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DX” có độ nhận dạng chính xác là:
* Làn điệu “CH\_DXN” có độ nhận dạng chính xác là:
* Làn điệu “CH\_LL” có độ nhận dạng chính xác là:
* Làn điệu “CH\_LNC” có độ nhận dạng chính xác là:
* Làn điệu “CH\_NTQT” có độ nhận dạng chính xác là:
* Làn điệu “CH\_QG” có độ nhận dạng chính xác là:
* Làn điệu “CH\_QM” có độ nhận dạng chính xác là:
* Làn điệu “CH\_QTVD” có độ nhận dạng chính xác là:
* Làn điệu “CH\_SLC” có độ nhận dạng chính xác là:
* Làn điệu “CH\_SQC” có độ nhận dạng chính xác là:
* Làn điệu “CH\_TV” có độ nhận dạng chính xác là:
* Làn điệu “CH\_XX” có độ nhận dạng chính xác là:
* Từ đó ta tính được độ chính xác trung bình là:

**Nhận xét:**

* Làn điệu đạt được độ chính xác cao nhất là làn điệu “Xẩm xoan” (CH\_XX), với độ chính xác .
* Làn điệu có độ chính xác thấp nhất là làn điệu “Đường trường tiếng đàn” (CH\_DTTD), có độ chính xác .
* Một số làn điệu có hiện tượng bị nhận nhầm lẫn nhau, ví dụ như làn điệu “Bình thảo” và “Cấm giá”, “Đào Liễu” và “Chức cẩm hồi văn”,…

Ma trận nhầm lẫn nhận dạng cụ thể từng làn điệu chèo với các hệ số thành phần Gauss khác nhau có thể tra cứu ở phần A.2 của Phụ lục.

Dưới đây sẽ là kết quả tính theo độ chính xác trung bình của bài toán phân lớp cụ thể làn điệu chèo với số các thành phần Gauss khác nhau:

Hình 4.2. Kết quả nhận dạng cụ thể làn điệu chèo khi thay đổi hệ số M với dữ liệu thử nghiệm không nằm trong tập huấn luyện

**Nhận xét:**

* Kết quả thu được có độ chính xác trung bình chưa được cao (kết quả cao nhất thu được là với số thành phần Gauss là 512 đạt được độ chính xác trung bình là 42,38%).
* Kết quả trả về có độ chính xác thấp nhất với hệ thống có số thành phần Gauss là 16 ().

1. Kết quả phân lớp làn điệu dân ca quan họ

Tiến hành thử nghiệm với hệ số thành phần Gauss (M) bằng 512 ta thu được kết quả như sau:



Bảng 4.3. Ma trận nhầm lẫn nhận nhận dạng cụ thể làn điệu dân ca quan họ với dữ liệu thử nghiệm không nằm trong tập huấn luyện (với M = 512)

Từ bảng 4.3 ta tính được độ chính xác của từng làn điệu như sau:

* Làn điệu “QH\_BBBD” có độ nhận dạng chính xác là:
* Làn điệu “QH\_BDMT” có độ nhận dạng chính xác là:
* Làn điệu “QH\_BQ” có độ nhận dạng chính xác là:
* Làn điệu “QH\_CD” có độ nhận dạng chính xác là:
* Làn điệu “QH\_CTVTVT” có độ nhận dạng chính xác là:
* Làn điệu “QH\_CTX” có độ nhận dạng chính xác là:
* Làn điệu “QH\_CVGCTQ” có độ nhận dạng chính xác là:
* Làn điệu “QH\_DBBMCG” có độ nhận dạng chính xác là:
* Làn điệu “QH\_DCSC” có độ nhận dạng chính xác là:
* Làn điệu “QH\_DQNB” có độ nhận dạng chính xác là:
* Làn điệu “QH\_GDCC” có độ nhận dạng chính xác là:
* Làn điệu “QH\_GTGM” có độ nhận dạng chính xác là:
* Làn điệu “QH\_LGD” có độ nhận dạng chính xác là:
* Làn điệu “QH\_LVDC” có độ nhận dạng chính xác là:
* Làn điệu “QH\_MN” có độ nhận dạng chính xác là:
* Làn điệu “QH\_NGMD” có độ nhận dạng chính xác là:
* Làn điệu “QH\_NMKN” có độ nhận dạng chính xác là:
* Làn điệu “QH\_NODHLV” có độ nhận dạng chính xác là:
* Làn điệu “QH\_NQNL” có độ nhận dạng chính xác là:
* Làn điệu “QH\_SCLK” có độ nhận dạng chính xác là:
* Làn điệu “QH\_SMAD” có độ nhận dạng chính xác là:
* Làn điệu “QH\_TETCHL” có độ nhận dạng chính xác là:
* Làn điệu “QH\_TPTN” có độ nhận dạng chính xác là:
* Làn điệu “QH\_TRBMSTC” có độ nhận dạng chính xác là:
* Làn điệu “QH\_VBM” có độ nhận dạng chính xác là:
* Từ đó ta tính được độ nhận dạng chính xác trung bình là:

**Nhận xét:**

* Với cùng số thành phần Gauss là 512 thì so với phân lớp cụ thể làn điệu chèo thì việc phân lớp cụ thể làn điệu dân ca quan họ có độ chính xác trung bình cao hơn (độ chính xác trung bình của phân lớp cụ thể làn điệu chèo là , của làn điệu dân ca quan họ là )
* Làn điệu đạt được độ nhận dạng chính xác cao nhất là làn điệu “Tìm em trong chiều hội Lim” (QH\_TETCHL) với độ nhận dạng chính xác . Ngoài ra, còn có một số làn điệu có độ chính xác trên như: làn điệu “Đối ca sông Cầu” (QH\_DCSC) – , làn điệu “Vui bốn mùa” (QH\_VBM) –
* Làn điệu có độ nhận dạng chính xác thấp nhất là làn điệu “Cây trúc xinh” (QH\_CTX) – .

Ma trận nhầm lẫn nhận dạng cụ thể từng làn điệu dân ca quan họ với các hệ số thành phần Gauss khác nhau có thể tra cứu ở phần A.3 của Phụ lục.

Dưới đây sẽ là kết quả tính theo độ chính xác trung bình của bài toán phân lớp cụ thể với số các thành phần Gauss khác nhau:

Hình 4.3. Kết quả nhận dạng cụ thể làn điệu dân ca quan họ khi thay đổi hệ số M với dữ liệu thử nghiệm không nằm trong tập huấn luyện

**Nhận xét:**

* Với số thành phần Gauss bằng 16 thì hệ thống thử nghiệm có kết quả nhận dạng chính xác trung bình thấp nhất ().
* Với số thành phần Gauss bằng 1024 thì hệ thống thử nghiệm có kết qủa nhận dạng chính xác trung bình cao nhất ().
* Với số thành phần Gauss lớn hơn 128 thì độ nhận dạng chính xác trung bình so với kết quả cao nhất thì chênh lệch không quá lớn. Ví dụ: với M = 128 thì độ chính xác trung bình của hệ thống thử nghiệm là 61,70% chỉ thấp hơn 1,10% so với kết quả nhận dạng chính xác trung bình cao nhất.
  1. Kết quả thử nghiệm với dữ liệu lấy từ tập huấn luyện
     1. Kết quả thử nghiệm phân lớp sơ bộ

Tiến hành thử nghiệm với số thành phần Gauss (M) là 512 ta thu được kết quả như sau:



Bảng 4.4. Ma trận nhầm lẫn phân lớp sơ bộ với dữ liệu thử nghiệm lấy từ tập huấn luyện (M = 512)

Từ bảng 4.4 ta tính được độ chính xác trung bình của hệ thống phân lớp sơ bộ đối với từng độ dài nhất định theo thời gian của dữ liệu thử nghiệm.

* Với độ dài là “4s” thì độ chính xác trung bình:
* Với độ dài là “6s” thì độ chính xác trung bình:
* Với độ dài là “8s” thì độ chính xác trung bình:
* Với độ dài là “10s” thì độ chính xác trung bình:
* Với độ dài là “12s” thì độ chính xác trung bình:
* Với độ dài là “14s” thì độ chính xác trung bình:
* Với việc để nguyên cả file thì độ chính xác trung bình:

Dưới đây là kết quả thử nghiệm với số các thành phần Gauss khác nhau:



Bảng 4.5. Độ chính xác trung bình của hệ thống phân lớp sơ bộ có độ dài của file thử nghiệm và số thành phần Gauss thay đổi

Ma trận nhầm lẫn phân lớp sơ bộ có độ dài của file thử nghiệm và hệ số thành phần Gauss thay đổi có thể tra cứu ở phần B.1 của Phụ lục.

Hình 4.4 là đồ thị trực quan biểu diễn cho bảng 4.5

Hình 4.4. Biểu đồ kết quả của hệ thống phân lớp sơ bộ có độ dài của file thử nghiệm và số thành phần Gauss thay đổi

**Nhận xét:**

* Nhìn vào đồ thị ta thấy với cùng một hệ số M thì độ dài của file thử nghiệm càng dài thì độ chính xác lại càng cao.
* Với cùng một độ dài của file thử nghiệm thì nếu hệ số M tăng thì cũng kéo theo độ chính xác trung bình của hệ thống phân lớp sơ bộ cũng tăng theo.
* Độ chính xác của việc nhận dạng thể loại được tăng lên đáng kể khi độ dài file thử nghiệm tăng từ 4s thành 6s và từ 6s tăng lên thành 8s. Với các độ dài sau như 10s, 12s, 14s, 16s kết quả nhận dạng có tăng nhưng tăng không đáng kể.
  + 1. Kết quả thử nghiệm phân lớp cụ thể

1. Kết quả phân lớp làn điệu chèo

Tiến hành thử nghiệm với hệ số thành phần Gauss (M) bằng 512 ta thu được kết quả tương ứng với từng độ dài của file thử nghiệm như sau:



Bảng 4.6. Độ chính xác trung bình của hệ thống phân lớp cụ thể làn điệu chèo tùy theo độ dài của file thử nghiệm với M = 512

Ma trận nhầm lẫn nhận dạng cụ thể từng làn điệu chèo tùy theo độ dài của file thử nghiệm với số thành phần Gauss bằng 512 có thể tra cứu ở phần B.2 của Phụ lục.

**Nhận xét:**

* So với việc thử nghiệm với dữ liệu không nằm trong tập huấn luyện thì việc thử nghiệm với dữ liệu lấy từ tập huấn luyện cho độ chính xác khá cao ngay với cả độ dài của file thử nghiệm rất ngắn (chỉ 4s nhưng độ nhận dạng chính xác trung bình đã là ).
* Độ dài file thử nghiệm càng dài thì độ nhận dạng chính xác trung bình càng cao.
* Với độ dài file thử nghiệm trên 8s thì độ nhận dạng chính xác trung bình đã đạt được trên và từ 14s trở lên đạt được độ chính xác trung bình trên .

Dưới đây là kết quả thử nghiệm với số các thành phần Gauss khác nhau:



Bảng 4.7. Độ chính xác trung bình của hệ thống phân lớp cụ thể làn điệu chèo có độ dài của file thử nghiệm và số thành phần Gauss thay đổi

Hình 4.5 là đồ thị trực quan biểu diễn cho bảng 4.7

Hình 4.5. Biểu đồ kết quả phân lớp cụ thể làn điệu chèo có độ dài file thử nghiệm và số thành phần Gauss thay đổi

**Nhận xét:**

* Số thành phần Gauss tăng lên thì kéo theo độ chính xác của hệ thống phân lớp cũng tăng theo, với các hệ số thành phần Gauss thay đổi từ 16 đến 256 thì sự biến đổi này thể hiện rất rõ rệt. Ví dụ như với độ dài file thử nghiệm là 4s thì khi tăng hệ số M từ 16 thành 32 thì độ chính xác cũng tăng từ lên thành , còn hệ số M tăng từ 256 lên thành 512 thì độ chính xác chỉ tăng từ lên thành .
* Với số thành phần Gauss từ 256 trở lên thì với độ dài file thử nghiệm từ chỉ từ 14s đã cho kết quả nhận dạng trên . Điều này chứng tỏ với làn điệu Chèo đã từng được huấn luyện thì không cần nghe hết cả bài mà chỉ cần tối đa là 16s thì hệ thống đã có thể trả về kết quả xấp xỉ chính xác

1. Kết quả phân lớp làn điệu dân ca quan họ

Tiến hành thử nghiệm với hệ số thành phần Gauss (M) bằng 512 ta thu được kết quả tương ứng với từng độ dài của file thử nghiệm như sau:



Bảng 4.8. Độ chính xác trung bình của hệ thống phân lớp cụ thể làn điệu dân ca quan họ tùy theo độ dài của file thử nghiệm với M = 512

Ma trận nhầm lẫn nhận dạng cụ thể từng làn điệu dân ca quan họ tùy theo độ dài của file thử nghiệm với số thành phần Gauss bằng 512 có thể tra cứu ở phần B.2 của Phụ lục.

**Nhận xét:**

* Với cùng số thành phần Gauss thì kết quả nhận dạng chính xác trung bình của phân lớp cụ thể làn điệu dân ca quan họ thấp hơn so với kết quả phân lớp cụ thể làn điệu chèo.
* Với độ dài của file thử nghiệm trên 8s thì cho kết quả nhận dạng trung bình cao trên 90%.
* Độ dài file thử nghiệm càng dài thì độ nhận dạng chính xác trung bình càng cao.

Dưới đây là kết quả thử nghiệm với số các thành phần Gauss khác nhau:



Bảng 4.9. Độ chính xác trung bình của hệ thống phân lớp cụ thể làn điệu dân ca quan họ có độ dài của file thử nghiệm và số thành phần Gauss thay đổi

Hình 4.6 là đồ thị trực quan biểu diễn cho bảng 4.9

Hình 4.6. Biểu đồ kết quả phân lớp cụ thể làn điệu dân ca quan họ có độ dài file thử nghiệm và số thành phần Gauss thay đổi

**Nhận xét:**

* Nhìn vào biểu đồ trên ta thấy thì cũng giống như phân lớp cụ thể làn điệu chèo, độ chính xác trung bình của phân lớp cụ thể làn điệu dân ca quan họ đều tăng khi ta tăng độ dài file thử nghiệm hoặc số thành phần Gauss.
* Độ chính xác có xu hướng tăng mạnh nhất khi độ dài file thử nghiệm tăng từ 4s lên thành 6s. Ví dụ, như từ độ dài file thử nghiệm là 4s tăng lên thành 6s thì độ chính xác của hệ thống thử nghiệm tăng từ 33,96% lên thành 49,58%.
* Với số thành phần Gauss nhỏ hơn 256 thì việc thay đổi độ dài dài file thử nghiệm theo bước nhảy 2s/lần thì độ chính xác đều có sự tăng lên rõ rệt. Còn với số thành phần Gauss lớn hơn 256 thì sự thay đổi về độ chính xác này chỉ nhì thấy rõ rệt nhất ở độ dài file thử nghiệm nhỏ hơn 10s vì từ 10s trở lên thì hệ thống đã đạt được độ chính xác là trên 95% nên sự biến động sẽ không còn nhiều với các file có độ dài lớn hơn.

CHƯƠNG 5. KẾT LUẬN

1. 1. Các kết quả đạt được

Thông qua nghiên cứu đề tài: “**Phân lớp một số làn điệu chèo và dân ca quan họ**” em đã đạt được một số kết quả sau:

* Tìm hiểu được các kiến thức liên quan về học máy nói chung và bài toán phân lớp trong học máy nói riêng.
* Tìm hiểu được nguyên lý hoạt động của một số mô hình phân lớp.
* Nghiên cứu được cách cài đặt và sử dụng hai bộ công cụ ALIZE và SPro trong giải quyết bài toán phân lớp.
* Xây dựng được cơ sở dữ liệu các làn điệu chèo và dân ca quan họ.
* Cải thiện khả năng đọc hiểu thuật toán và nghiên cứu tài liệu bằng tiếng Anh.
* Nâng cao khả năng lập trình Python.
* Xây dựng được bộ chương trình dùng để thử nghiệm việc phân lớp làn điệu chèo và dân ca quan họ bằng Python.
  1. Các vấn đề chưa đạt được

Kết quả thu được của đề tài mới chỉ góp phần vào việc nghiên cứu hiệu quả của mô hình cũng như bộ công cụ hỗ trợ trong việc phân lớp một số làn điệu chèo và dân ca quan họ, chưa ra được sản phẩm phần mềm có thể sử dụng được trong cuộc sống.

Chưa giải quyết được bài toán xác định tên của nhiều làn điệu chèo trong cùng một bài hát.

Kho dữ liệu thử nghiệm chưa bao phủ được hết tất cả các làn điệu chèo và dân ca quan họ hiện có của Việt Nam.

* 1. Hướng phát triển của đề tài

Với những kết quả đã đạt được, hướng phát triển của đề tài là:

* Xây dựng một ứng dụng thực tế về nhận dạng làn điệu chèo và dân ca quan họ để phục vụ nhu cầu của người sử dụng.
* Cải thiện cả về số lượng cũng như chất lượng của cơ sở dữ liệu các làn điệu chèo và dân ca quan họ.
* Dựa trên nền tảng đã có mở rộng bài toán không chỉ dừng lại ở phân lớp làn điệu chèo và dân ca quan họ mà sẽ là phân lớp các thể loại nhạc cổ truyền của Việt Nam.

TÀI LIỆU THAM KHẢO

[1] Hoàng Kiều, Hà Hoa (2007), *Những làn điệu chèo cổ chọn lọc*, NXB Văn hóa thông tin.

[2] Lâm Minh Đức (2005). *Dân ca quan họ Bắc Ninh – 100 bài hát lời cổ*, NXB Thanh niên

[3] TS. Ngô Hữu Phúc (2016). *Slide Lý thuyết nhận dạng.*

[4] Phan Anh Cang, Phan Thượng Cang. *Phân loại nhạc theo thể loại dùng phép biến đổi Wavelet rời rạc,* Kỷ yếu Hội nghị Khoa học Quốc gia lần thứ IX “Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR'9)”.

[5] Lê Nhật Thăng (2010). *Slide Xử lý âm thanh và hình ảnh*.

[6] Đào Thị Thu Diệp (2013). *Luận văn Nhận dạng người nói phụ thuộc vào từ khóa tiếng Việt*.

Website:

[7] <https://tech.3si.vn/2016/03/31/ml-classification-part-3> - *Quá trình xây dựng cây quyết định.*

[8] <http://bis.net.vn/forums/t/370.aspx> - *Thuật toán K-láng giềng gần nhất.*

[9] <https://www.irisa.fr/metiss/guig/spro/spro-4.0.1/spro_3.html> - *Tìm hiểu về bộ công cụ SPro.*

[10] <https://pdfs.semanticscholar.org/ca84/a0e8a34dffac72775eeeea88ca94ae60717d.pdf> - *Tìm hiểu về bộ công cụ ALIZE.*

[11] <https://vi.wikipedia.org/wiki/Ch%C3%A8o> – *Giới thiệu về chèo*

[12] <https://123doc.org//document/3370993-dac-diem-cua-quan-ho.htm> - *Đặc điểm của dân ca quan họ.*

[13]

<http://nganhkhoahocmaytinh.com/index.php/2016/12/02/cay-quyet-dinh-decision-tree-phan-1/> - *Ưu điểm của mô hình cây quyết định.*

[14] <http://doan.edu.vn/do-an/mo-hinh-markov-an-hop-gauss-29561/> - *Mô hình hỗn hợp Gauss*

PHỤ LỤC

1. Kết quả thử nghiệm với dữ liệu nằm không nằm trong tập huấn luyện.

A.1. Kết quả thử nghiệm phân lớp sơ bộ



Bảng A.1. Kết quả phân lớp sơ bộ đối của từng làn điệu với các hệ số M khác nhau

Với việc tiến hành phân lớp sơ bộ (xác định một làn điệu là chèo hay quan họ), thì bảng A.1 ghi lại kết quả thử nghiệm với mỗi làn điệu là 20 file âm thanh, cùng với các hệ số M khác nhau. Trong đó CH: kí hiệu cho đây là những file thuộc làn điệu chèo, QH: kí hiệu cho những file thuộc làn điệu dân ca quan họ.

Kết quả của bảng A.1 được tập hợp lại trong bảng A.2 dưới đây.



Bảng A.2. Ma trận nhầm lẫn phân lớp sơ bộ với các hệ số M khác nhau

A.2. Kết quả thử nghiệm phân lớp cụ thể làn điệu chèo



Bảng A.3. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 16



Bảng A.4. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 32



Bảng A.5. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 64



Bảng A.6. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 128



Bảng A.7. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 256



Bảng A.8. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 1024



Bảng A.9. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 2048



Bảng A.10. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo với M = 4096

A.3. Kết quả thử nghiệm phân lớp cụ thể làn điệu dân ca quan họ



Bảng A.11. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 16



Bảng A.12. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 32



Bảng A.13. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 64



Bảng A.14. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 128



Bảng A.15. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 256



Bảng A.16. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 1024



Bảng A.17. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 2048



Bảng A.18. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ với M = 4096

B. Kết quả thử nghiệm với dữ liệu lấy từ tập huấn luyện

B.1. Kết quả thử nghiệm phân lớp sơ bộ

* Với M = 16



Bảng B.1. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 16

* Với M = 32



Bảng B.2. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 32

* Với M = 64



Bảng B.3. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 64

* Với M = 128



Bảng B.4. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 128

* Với M = 256



Bảng B.5. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 256

* Với M = 1024



Bảng B.6. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 1024

* Với M = 2048



Bảng B.7. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 2048

* Với M = 4096



Bảng B.8. Ma trận nhầm lẫn phân lớp sơ bộ với các độ dài khác nhau của file thử nghiệm khi M = 4096

B.2. Kết quả thử nghiệm phân lớp cụ thể làn điệu chèo

B.2.1. Kết quả thử nghiệm với M = 512



Bảng B.9. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 4s



Bảng B.10. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 6s



Bảng B.11. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 8s



Bảng B.12. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 10s



Bảng B.13. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 12s



Bảng B.14. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 14s



Bảng B.15. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng 16s



Bảng B.16. Ma trận nhầm lẫn phân lớp cụ thể làn điệu chèo khi độ dài file thử nghiệm bằng độ dài của file huấn luyện

B.3. Kết quả thử nghiệm phân lớp cụ thể làn điệu dân ca quan họ

B.3.1. Kết quả thử nghiệm với M = 512



Bảng B.17. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 4s



Bảng B.18. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 6s



Bảng B.19. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 8s



Bảng B.20. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 10s



Bảng B.21. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 12s



Bảng B.22. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 14s



Bảng B.23. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng 16s



Bảng B.24. Ma trận nhầm lẫn phân lớp cụ thể làn điệu dân ca quan họ khi độ dài file thử nghiệm bằng độ dài của file huấn luyện