

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA ĐA PHƯƠNG TIỆN

BÀI GIẢNG MÔN

**KHAI PHÁ DỮ LIỆU  
ĐA PHƯƠNG TIỆN**

TS. Đỗ Thị Liên

**HÀ NỘI - 2022**

## MỤC LỤC

<b>MỤC LỤC</b> .....	i
<b>DANH MỤC HÌNH VẼ</b> .....	iii
<b>LỜI NÓI ĐẦU</b> .....	1
<b>CHƯƠNG 1</b> .....	3
<b>TỔNG QUAN VỀ KHAI PHÁ DỮ LIỆU ĐA PHƯƠNG TIỆN</b> .....	3
<b>1.1. Thông tin và dữ liệu</b> .....	3
1.1.1. <i>Thông tin</i> .....	3
1.1.2. <i>Dữ liệu, dữ liệu đa phương tiện</i> .....	4
1.1.3. <i>Các cách thức lưu trữ dữ liệu số</i> .....	6
<b>1.2. Khai phá dữ liệu</b> .....	6
1.2.1. <i>Khái niệm khai phá dữ liệu</i> .....	6
1.2.2. <i>Quy trình khai phá dữ liệu</i> .....	7
1.2.3. <i>Phân loại các bài toán khai phá dữ liệu</i> .....	10
1.2.4. <i>Một số nền tảng công nghệ hỗ trợ khai phá dữ liệu</i> .....	11
1.2.5. <i>Những thách thức trong ứng dụng và nghiên cứu về khai phá dữ liệu</i> .....	12
<b>1.3. Khai phá dữ liệu đa phương tiện</b> .....	12
1.3.1. <i>Khái niệm khai phá dữ liệu đa phương tiện</i> .....	12
1.3.2. <i>Phân loại khai phá dữ liệu đa phương tiện</i> .....	13
1.3.3. <i>Quy trình khai phá dữ liệu đa phương tiện</i> .....	15
1.3.4. <i>Ứng dụng của khai phá dữ liệu đa phương tiện</i> .....	15
<b>TỔNG KẾT CHƯƠNG 1</b> .....	16
<b>CÂU HỎI VÀ BÀI TẬP CHƯƠNG 1</b> .....	17
<b>CHƯƠNG 2</b> .....	18
<b>TIỀN XỬ LÝ DỮ LIỆU ĐA PHƯƠNG TIỆN</b> .....	18
<b>2.1. Biểu diễn và mô tả tập dữ liệu trên máy tính cho mục đích khai phá dữ liệu</b> .....	18
2.1.1. <i>Biểu diễn tập dữ liệu</i> .....	18
2.1.2. <i>Mô tả thông tin từ tập dữ liệu</i> .....	20
<b>2.2. Khái niệm tiền xử lý dữ liệu đa phương tiện</b> .....	21
<b>2.3. Tầm quan trọng của tiền xử lý dữ liệu đa phương tiện</b> .....	22
<b>2.4. Những nhiệm vụ chính của tiền xử lý dữ liệu đa phương tiện</b> .....	22
2.4.1. <i>Trích chọn đặc trưng dữ liệu</i> .....	23
2.4.2. <i>Dọn dẹp dữ liệu</i> .....	34
<b>2.5. Tiền xử lý dữ liệu đa phương tiện với Weka</b> .....	43
2.5.1. <i>Giới thiệu Weka</i> .....	43

2.5.2. <i>Mô tả dữ liệu trong Weka</i> .....	48
2.5.3. <i>Tiền xử lý dữ liệu với Weka</i> .....	50
<b>TỔNG KẾT CHƯƠNG 2 .....</b>	<b>69</b>
<b>CÂU HỎI VÀ BÀI TẬP CHƯƠNG 2 .....</b>	<b>69</b>
<b>CHƯƠNG 3 .....</b>	<b>71</b>
<b>MÔ HÌNH KHAI PHÁ DỮ LIỆU ĐA PHƯƠNG TIỆN .....</b>	<b>71</b>
<b>    3.1. Phân lớp dữ liệu.....</b>	<b>71</b>
3.1.1. <i>Bài toán phân lớp dữ liệu</i> .....	71
3.1.2. <i>Chuẩn bị dữ liệu cho quá trình phân lớp</i> .....	74
3.1.3. <i>Phương pháp phân lớp dữ liệu</i> .....	78
3.1.4. <i>Đánh giá mô hình phân lớp dữ liệu</i> .....	84
3.1.5. <i>Một số ứng dụng của bài toán phân lớp:</i> .....	88
<b>    3.2. Phân cụm dữ liệu.....</b>	<b>88</b>
3.2.1. <i>Bài toán phân cụm dữ liệu .....</i>	88
3.2.2. <i>Phương pháp phân cụm dữ liệu .....</i>	90
3.2.3. <i>Đánh giá mô hình phân cụm dữ liệu .....</i>	95
3.2.4. <i>Một số ứng dụng của bài toán phân cụm .....</i>	97
<b>    3.3. Khai phá luật kết hợp .....</b>	<b>97</b>
3.3.1. <i>Bài toán khai phá luật kết hợp .....</i>	97
3.3.2. <i>Phương pháp khai phá luật kết hợp .....</i>	100
3.3.3. <i>Một số ứng dụng của bài toán khai phá luật kết hợp .....</i>	104
<b>    TỔNG KẾT CHƯƠNG 3 .....</b>	<b>105</b>
<b>    CÂU HỎI VÀ BÀI TẬP CHƯƠNG 3 .....</b>	<b>105</b>
<b>    TÀI LIỆU THAM KHẢO .....</b>	<b>107</b>

## DANH MỤC HÌNH VẼ

	Trang
Hình 1.1. Mối quan hệ giữa dữ liệu, thông tin và tri thức .....	3
Hình 1.2. Minh họa dữ liệu dạng in ấn.....	4
Hình 1.3. Minh họa các thiết bị trao đổi dữ liệu dạng quảng bá .....	5
Hình 1.4. Minh họa nền tảng thiết bị và ứng dụng trao đổi dữ liệu số.....	5
Hình 1.5. Minh họa các sản phẩm đa phương tiện số tương tác .....	6
Hình 1.6. Quy trình phát hiện tri thức từ dữ liệu KDD [FPS96].....	8
Hình 1.7. Quy trình KDD dưới góc nhìn từ học máy thống kê.....	9
Hình 1.8. Quy trình khai phá dữ liệu lặp CRIP-DM .....	9
Hình 1.9. Phân loại khai phá dữ liệu theo mục đích sử dụng .....	11
Hình 1.10. Phân loại khai phá dữ liệu theo kỹ thuật thực thi .....	11
Hình 1.11. Phân loại khai phá dữ liệu đa phương tiện theo loại dữ liệu khai phá .....	13
Hình 1.12. Quy trình khai phá dữ liệu đa phương tiện.....	15
Hình 2.1. Minh họa một số cách biểu diễn tập dữ liệu.....	19
Hình 2.2. Minh họa biểu diễn tập dữ liệu hoa văn Iris dưới dạng bản ghi .....	19
Hình 2.3. Ví dụ mô tả thông tin từ tập dữ liệu qua một số công thức độ đo .....	20
Hình 2.4. Minh họa mô tả thông tin từ tập dữ liệu qua biểu đồ histogram .....	21
Hình 2.5. Minh họa mô tả thông tin từ tập dữ liệu qua biểu đồ scatter plot .....	21
Hình 2.6. Thông kê tỉ trọng các công việc cần thực hiện trong quá trình khai phá dữ liệu .....	22
Hình 2.7. Nguyên tắc hoạt động của trích chọn đặc trưng văn bản .....	23
Hình 2.8. Phân loại các kỹ thuật vector hóa văn bản .....	24
Hình 2.9. Các bước trích chọn đặc trưng văn bản theo kỹ thuật Bag of words .....	25
Hình 2.10. Nguyên tắc hoạt động của trích chọn đặc trưng hình ảnh .....	27
Hình 2.11. Ba loại đặc trưng của hình ảnh và các kỹ thuật thực hiện chính .....	28
Hình 2.12. Minh họa quá trình trích chọn đặc trưng ảnh theo phương pháp k-color histogram .....	29
Hình 2.13. Nguyên tắc hoạt động của trích chọn đặc trưng âm thanh .....	29
Hình 2.14. Phân loại tín hiệu âm thanh nghe được .....	30
Hình 2.15. Quá trình trích chọn đặc trưng âm thanh theo phương pháp MFCC.....	31
Hình 2.16. Cấu trúc của video .....	33
Hình 2.17. Ví dụ minh họa về cấu trúc của 1 video .....	33
Hình 2.18. Nguyên tắc hoạt động của trích chọn đặc trưng video .....	34
Hình 2.19. Chu trình dọn dẹp dữ liệu .....	35
Hình 2.20. Tích hợp dữ liệu .....	36
Hình 2.21. Minh họa về tích hợp dữ liệu về lương nhân viên công ty qua các năm.....	37
Hình 2.22. Tiếp cận liên kết chặt chẽ để tích hợp dữ liệu (Tight coupling) .....	37
Hình 2.23. Tiếp cận liên kết lỏng lẻo để tích hợp dữ liệu (Loose Coupling).....	38

Hình 2.24. Minh họa quá trình chuyển đổi dữ liệu tuổi và lương trong khai phá dữ liệu nhân viên.....	39
Hình 2.25. Các hướng tiếp cận giảm chiều dữ liệu .....	40
Hình 2.26. Nguyên tắc hoạt động của hướng tiếp cận giảm bớt chiều dữ liệu .....	40
Hình 2.27. Các kỹ thuật giảm bớt chiều dữ liệu.....	41
Hình 2.28. Nguyên tắc hoạt động của hướng tiếp cận giảm bớt số lượng bộ dữ liệu..	41
Hình 2.29. Các kỹ thuật giảm bớt số lượng bộ dữ liệu dữ liệu .....	42
Hình 2.30. Minh họa dữ liệu cần được làm sạch.....	42
Hình 2.31. Các bước phổ biến thực hiện để làm sạch dữ liệu .....	43
Hình 2.32. Giao diện phần mềm Weka .....	44
Hình 2.33. Giao diện chức năng Weka Explorer.....	44
Hình 2.34. Luồng xử lý dữ liệu trong Weka Explorer .....	45
Hình 2.35. Giao diện chức năng Weka Experimenter.....	46
Hình 2.36. Giao diện chức năng Weka Knowledge Flow .....	46
Hình 2.37. Giao diện chức năng Workbench .....	47
Hình 2.38. Giao diện chức năng Simple CLI .....	47
Hình 2.39. Ví dụ và giải thích về tệp tin ARFF làm việc trong môi trường Weka.....	48
<del>Hình 3.1. Minh họa các hướng tiếp cận học máy cơ bản dựa trên phương thức học dữ liệu .....</del>	<del>72</del>
Hình 3.2. Quá trình phân lớp dữ liệu theo hướng tiếp cận học máy có giám sát.....	72
Hình 3.3. Quá trình phân lớp dữ liệu theo hướng tiếp cận học máy bán giám sát.....	73
Hình 3.4. Một số phương pháp phân lớp dữ liệu.....	78
Hình 3.5. Minh họa thuật toán phân lớp kNN .....	79
Hình 3.6. Chia dữ liệu theo phương pháp Holdout .....	80
Hình 3.7. Chia dữ liệu theo phương pháp Cross-validation.....	81
Hình 3.8. Chia dữ liệu theo phương pháp Bootstrap .....	81
Hình 3.9. Minh họa so sánh độ chính xác của các mô hình phân lớp thông qua đường cong ROC .....	87
Hình 3.10. Minh họa bài toán phân cụm dữ liệu .....	89
Hình 3.11. Quá trình phân cụm dữ liệu theo hướng tiếp cận học máy không giám sát	89
Hình 3.12. Một số phương pháp phân cụm dữ liệu .....	91
Hình 3.13. Minh họa thuật toán phân cụm k-means .....	92
Hình 3.14. Minh họa bài toán phân tích giỏ hàng .....	98
Hình 3.15. Quá trình khai phá luật kết hợp .....	100
Hình 3.16. Quá trình khai phá luật kết hợp cho bài toán phân tích giỏ hàng.....	100

## LỜI NÓI ĐẦU

Do sự phổ biến, độ đa dạng cũng nhu cầu sử dụng dữ liệu đa phương tiện ngày càng lớn nên việc lưu trữ, khai phá và sử dụng nguồn dữ liệu này cũng ngày càng gia tăng. Tận dụng dữ liệu đa phương tiện sẽ tạo điều kiện thuận lợi nhằm hỗ trợ cá nhân hóa thông tin phù hợp với người dùng, cụ thể một số ứng dụng của việc khai phá dữ liệu đa phương tiện như:

- Tìm kiếm hình ảnh, âm thanh
- Dự báo thời tiết
- Y học từ xa
- Nhận diện giọng nói, hình ảnh
- Xây dựng hệ thống đề xuất và gợi ý
- Quảng cáo
- ...

Một trong những nhiệm vụ trọng tâm của các hệ thống đa phương tiện trên chính là việc hiểu, làm chủ được quá trình cũng như đánh giá được mức độ phù hợp của việc khai phá dữ liệu cho tập dữ liệu cũng như mục tiêu mà bài toán hướng tới.

Bài giảng này nhằm cung cấp cho sinh viên có một cái nhìn tổng quan về khai phá dữ liệu nói chung và khai phá dữ liệu đa phương tiện nói riêng; các khái niệm có liên quan, ý nghĩa và tầm quan trọng. Trên cơ sở những hiểu biết cơ bản về quy trình khai phá dữ liệu đa phương tiện, môn học đi sâu vào cung cấp kiến thức và kỹ năng trong tiền xử lý và xây dựng mô hình khai phá dữ liệu đa phương tiện cùng các độ đo và phương pháp đánh giá liên quan. Trong quá trình thực hành, môn học hướng dẫn sinh viên tiếp cận mã nguồn mở Weka để hỗ trợ trong quá trình cài đặt và thử nghiệm hệ thống khai phá dữ liệu đa phương tiện.

Nội dung của bài giảng bao gồm ba chương:

- **Chương 1: Tổng quan về khai phá dữ liệu đa phương tiện.** những kiến thức cơ bản về khai phá dữ liệu nói chung và khai phá dữ liệu đa phương tiện nói riêng; Các khái niệm có liên quan, ý nghĩa và tầm quan trọng.
- **Chương 2: Tiền xử lý dữ liệu đa phương tiện.** Trình bày cụ thể về pha đầu tiên trong quy trình khai phá dữ liệu đa phương tiện được đề cập trong chương 1, đó là tiền xử lý dữ liệu và một số công việc liên quan.

- **Chương 3: Mô hình khai phá dữ liệu đa phương tiện.** Trình bày cụ thể về việc xây dựng và triển khai một số mô hình khai phá dữ liệu điển hình, áp dụng cho các dữ liệu đa phương tiện

Mặc dù tác giả đã có nhiều cố gắng trong quá trình biên soạn bài giảng này, song không thể tránh khỏi những thiếu sót. Rất mong nhận được sự đóng góp ý kiến của sinh viên và các bạn đồng nghiệp.

THƯ VIỆN PTIT

## CHƯƠNG 1

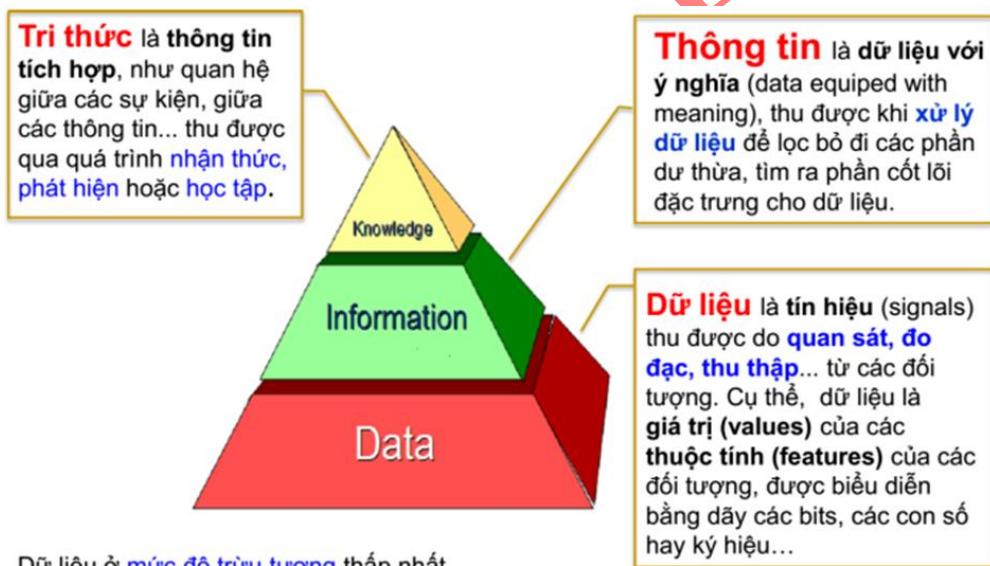
### TỔNG QUAN VỀ KHAI PHÁ DỮ LIỆU ĐA PHƯƠNG TIỆN

Nội dung chương này nhằm giới thiệu một cách tổng quát những kiến thức cơ bản về khai phá dữ liệu nói chung và khai phá dữ liệu đa phương tiện nói riêng; Các khái niệm có liên quan, ý nghĩa và tầm quan trọng. Nội dung trình bày bao gồm:

- Giới thiệu về thông tin, dữ liệu và dữ liệu đa phương tiện.
- Trình bày những kiến thức cơ bản về khai phá dữ liệu và khai phá dữ liệu đa phương tiện: Khái niệm, quy trình thực hiện, phân loại các bài toán khai phá dữ liệu, các thách thức trong nghiên cứu và ứng dụng khai phá dữ liệu đa phương tiện.

#### 1.1. Thông tin và dữ liệu

Mối quan hệ giữa dữ liệu, thông tin và tri thức được thể hiện qua hình vẽ sau:



Hình 1.1. Mối quan hệ giữa dữ liệu, thông tin và tri thức

Nội dung dưới đây sẽ cung cấp thông tin chi tiết về từng khái niệm này.

##### 1.1.1. Thông tin

- Khái niệm: Thông tin là các thông báo nhằm mang lại một sự hiểu biết nào đó cho đối tượng nhận tin.
- Thông tin là 1 loại nguồn lực đặc biệt quan trọng trong tổ chức, có vai trò:
  - Hoạch định, điều khiển tất cả các tiến trình trong tổ chức.

- Giúp hiểu rõ thị trường để cải tiến tổ chức và các hoạt động sản xuất kinh doanh, cũng như định hướng sản phẩm mới.
- Các hệ thống thông tin dựa trên máy tính giúp tự động hóa xử lý công việc nhanh gọn, hiệu quả.

### 1.1.2. *Dữ liệu, dữ liệu đa phương tiện*

#### - *Dữ liệu*

Theo điều 4 luật Giao dịch điện tử ban hành ngày 29 tháng 11 năm 2005, dữ liệu là thông tin dưới dạng ký hiệu, chữ viết, chữ số, hình ảnh, âm thanh hoặc tín hiệu dạng tương tự.

Trải qua quá trình phát triển của các phương tiện truyền thông, dữ liệu được phân chia thành 3 loại: 1/ Dữ liệu dạng in ấn (Print media); 2/ Dữ liệu dạng quảng bá (Broadcast media); 3/ Dữ liệu số (Digital media / New media). Trong đó:

- Dữ liệu dạng in ấn được thể hiện qua các sản phẩm trên giấy như báo giấy, sách vở, tạp chí giấy, truyện, sản phẩm quảng cáo in.



Newspaper



Books



Magazines



Comics



Brochures

Hình 1.2. Minh họa dữ liệu dạng in ấn

- Dữ liệu dạng quảng bá được truyền thông qua các sóng vô tuyến hoặc sóng điện từ trong môi trường không khí. Các thiết bị sử dụng dữ liệu dạng này có thể kể đến như: ti vi, radio, vệ tinh, điện thoại,...



**Hình 1.3. Minh họa các thiết bị trao đổi dữ liệu dạng quảng bá**

- Dữ liệu số: Ngày nay, mọi hoạt động đều được thực hiện thông qua các thiết bị thông minh như máy tính, điện thoại, máy tính bảng... Những công cụ đó đòi hỏi dạng dữ liệu mà chúng có thể đọc hiểu được. Các dữ liệu dạng văn bản, hình ảnh, âm thanh, video... được biểu diễn bằng hệ số nhị phân dựa trên các bit 1 và 0 mà máy tính chấp nhận gọi chung là dữ liệu số.



**Hình 1.4. Minh họa nền tảng thiết bị và ứng dụng trao đổi dữ liệu số**

Thế giới đang bước vào “kỷ nguyên số” với đặc điểm, tính chất và sự tác động sâu rộng chưa từng có, theo đó loại hình dữ liệu số đang dần chiếm lĩnh mọi mặt của đời sống và thay thế các loại hình dữ liệu khác trong việc truyền tải thông tin tới người dùng.

#### *- Dữ liệu đa phương tiện*

Dữ liệu đa phương tiện nhằm tới các kiểu thông tin được thể hiện thông qua việc kết hợp đa dạng các dữ liệu số như âm thanh, hình ảnh, văn bản, video.

Hiện nay dữ liệu đa phương tiện đang được việc tích hợp rất phổ biến trong các sản phẩm số (phần mềm, tài liệu số). Trong các sản phẩm số này ngoài mục tiêu truyền tải thông tin hiệu quả thì tính tương tác hai chiều cũng là một yếu tố đã và đang được khai thác để tăng trải nghiệm và khai thác thông tin hiệu quả từ người dùng.



Hình 1.5. Minh họa các sản phẩm đa phương tiện số tương tác

### **1.1.3. Các cách thức lưu trữ dữ liệu số**

Đứng trước vấn đề bùng nổ dữ liệu số như hiện nay thì vấn đề lưu trữ và xử lý dữ liệu số là một vấn đề đặc biệt được quan tâm thực hiện và nghiên cứu. Về cơ bản có một số cách thức lưu trữ dữ liệu số phổ biến sau:

- Tổ chức lưu trữ dữ liệu trong tệp tin (File) lưu trữ nội bộ tại các thiết bị số.
- Dữ liệu được tổ chức trong các cơ sở dữ liệu (Database) được quản lý bởi các hệ quản trị cơ sở dữ liệu.
- Tổ chức lưu trữ trong kho dữ liệu (Data warehouse), nơi tích hợp dữ liệu từ nhiều nguồn.
- Lưu trữ trên các kho dữ liệu đám mây (Cloud storage) như : Google driver, Dropbox, iCloud,...

### **1.2. Khai phá dữ liệu**

#### **1.2.1. Khái niệm khai phá dữ liệu**

Khai phá dữ liệu (Data mining) là quá trình khai thác mô hình hay tri thức thú vị (không tầm thường, tiềm ẩn, chưa từng được biết và có khả năng hữu ích) từ số lượng rất lớn của dữ liệu.

Một số bài toán tình huống dưới đây minh họa cho yêu cầu về việc cần áp dụng khai phá dữ liệu trong việc đưa ra tri thức mới:

- Tình huống 1: Trong việc vận hành hệ thống các máy ATM của ngân hàng, mỗi ngày có rất nhiều người dùng thẻ ATM với ID khác nhau tới thực hiện các giao

dịch liên quan tới tiền. Tuy nhiên việc sử dụng thẻ ATM này có trường hợp không được phép như thẻ bị ăn trộm, nếu ai dùng thẻ cũng có thể thực hiện giao dịch thì không ổn. Do vậy nhiệm vụ của hệ thống là phải có khả năng dự đoán xem người đang sử dụng thẻ với ID đưa vào có thật sự là chủ nhân của thẻ hay là một tên trộm ? Thông tin dự đoán được này từ hệ thống chính là tri thức mới được đưa ra và cần phải khai phá.

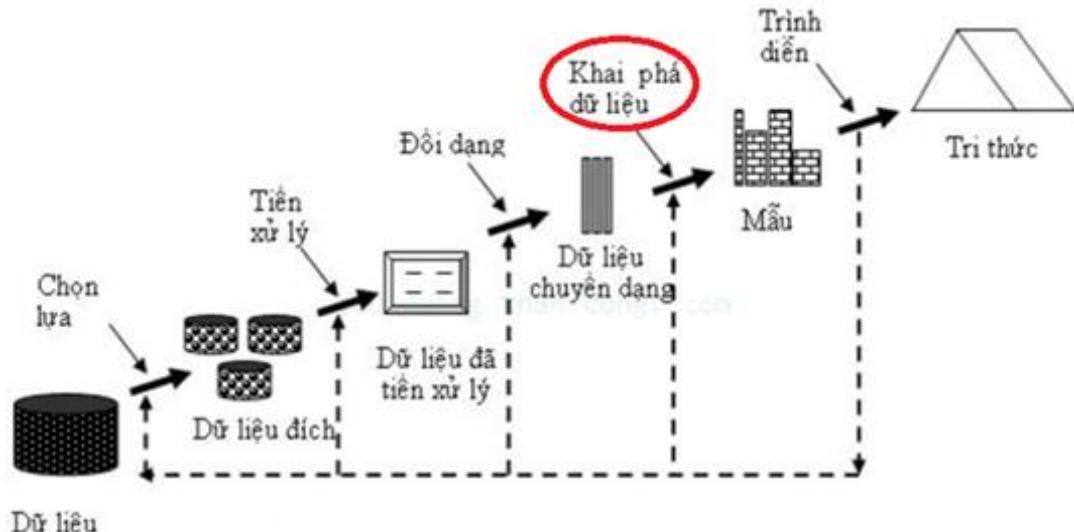
- Tình huống 2: Cơ quan quản lý thuế định kỳ đều cập nhật tình trạng đóng thuế của người dân, doanh nghiệp. Tuy nhiên không phải lúc nào mọi người dân hoặc tổ chức đều tự nguyện đóng thuế theo quy định mà có những trường hợp trốn thuế, điều này sẽ gây thất thoát nguồn thu ngân sách lớn cho nước nhà. Để trước tình trạng này thì cần thiết phải có một hệ thống có khả năng dự đoán khả năng trốn thuế của người dân và doanh nghiệp để kịp thời đưa ra những biện pháp xử lý. Khai phá dữ liệu phù hợp để xử lý trong tình huống này.
- Tình huống 3: Để đưa ra đầu tư hiệu quả trong lĩnh vực thị trường chứng khoán, người tham gia rất cần thiết dự đoán được trước giá cổ phiếu cho ngày hôm sau. Đây là tri thức mới chưa xảy ra.
- Tình huống 4: Các cơ sở giáo dục như các trường đại học mong muốn xác định trước khả năng tốt nghiệp đúng hạn của sinh viên đang trong tiến trình học tập tại nhà trường để có những tác động kịp thời.
- Tình huống 5: Trong lĩnh vực thương mại điện tử, người ta mong muốn ngày càng tăng doanh số bán hàng. Để làm được điều này thì các hệ thống thương mại điện tử đã có cần tích hợp tính năng tư vấn hỗ trợ mua bán sản phẩm cá nhân hóa tốt tới người dân, có như vậy sẽ giữ chân được họ và tăng doanh số bán hàng là điều tất yếu xảy ra.
- Tình huống 6: Việc dự đoán số lượng bệnh nhân sẽ nhập viện trong năm tới là điều được các bệnh viện quan tâm nhằm có phương án chuẩn bị tốt nhất về đội ngũ nhân sự, thuốc thang và cơ sở vật chất. Điều này có thể thực hiện thông qua quá trình khai phá từ dữ liệu bệnh nhân trước đây của bệnh viện.

### **1.2.2. Quy trình khai phá dữ liệu**

Trải qua lịch sử hình thành và phát triển trong lĩnh vực khai phá dữ liệu đã có một số quy trình được đưa ra, trong đó điển hình có thể kể đến 2 quy trình chính: 1/ Quy trình khai phá tri thức từ cơ sở dữ liệu KDD và 2/ Quy trình khai phá dữ liệu lặp CRIP-DM. Nội dung phần trình bày dưới đây sẽ đưa ra góc nhìn theo 2 quy trình này để hiểu về quá trình xử lý trong khi thực hiện khai phá dữ liệu.

- *Quy trình khai phá tri thức từ cơ sở dữ liệu KDD*

Theo quy trình khai phá tri thức từ cơ sở dữ liệu KDD(Knowledge Discovery in Database) thì khai phá dữ liệu là một giai đoạn có mối quan hệ với các giai đoạn khác trong toàn bộ quy trình như sau:



[FPS96] Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth (1996). From Data Mining to Knowledge Discovery: An Overview, *Advances in Knowledge Discovery and Data Mining* 1996: 1-34

Hình 1.6. Quy trình phát hiện tri thức từ dữ liệu KDD [FPS96]

Trong đó các bước của quy trình KDD trên có thể được mô tả qua các bước như sau:

Bước 1: Tìm hiểu về mục đích của bài toán ứng dụng để tiến hành lựa chọn thu thập dữ liệu đích phù hợp từ tập hợp đa dạng dữ liệu ban đầu.

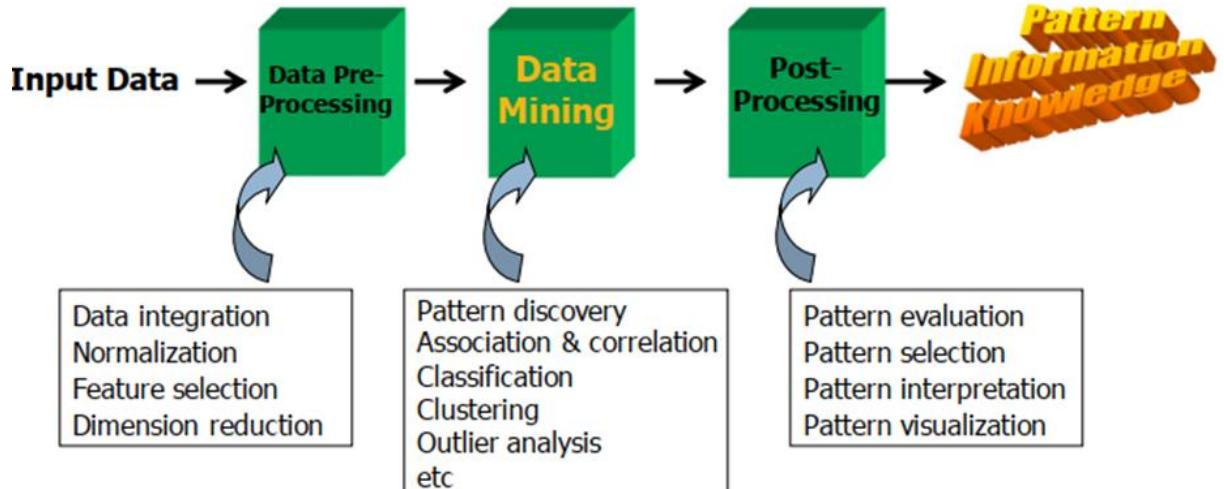
Bước 2: Tiền xử lý dữ liệu nhằm cải thiện chất lượng dữ liệu qua một số những phần việc cấu thành như làm sạch, tích hợp dữ liệu, biến đổi và giảm chiều dữ liệu.

Bước 3: Đổi dạng dữ liệu là bước liền trước giai đoạn khai phá dữ liệu với mục tiêu chuyển đổi dữ liệu đã tiền xử lý về dạng biểu diễn phù hợp với từng kỹ thuật khai phá dữ liệu khác nhau được lựa chọn.

Bước 4: Tiến hành quá trình khai phá dữ liệu bằng việc lựa chọn phương pháp và kỹ thuật phù hợp cho bài toán tiếp cận.

Bước 5: Đánh giá mẫu thu được và trực quan hóa trong biểu diễn tri thức.

Dưới góc nhìn học máy thống kê thì quá trình KDD được quy hoạch gọn thành ba giai đoạn: 1/ Tiền xử lý dữ liệu (Data pre-processing); 2/ Khai phá dữ liệu (Data mining); 3/ Hậu xử lý dữ liệu (Post-processing). Ba giai đoạn này được xâu chuỗi với nhau như sau:

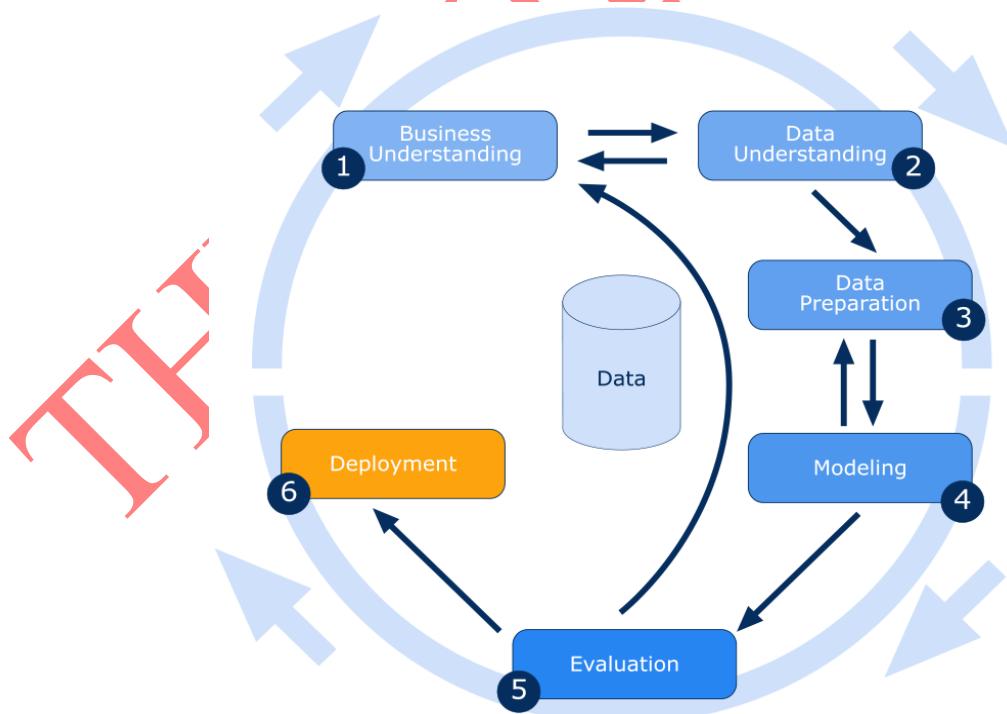


Hình 1.7. Quy trình KDD dưới góc nhìn từ học máy thống kê

Trong đó giai đoạn 1 của quá trình KDD dưới góc nhìn từ học máy thống kê này sẽ tương ứng với các bước 1, 2, 3 từ quy trình KDD [FPS96] đã đề xuất trước đó.

- *Quy trình khai phá dữ liệu lặp CRIP-DM*

Quy trình khai phá dữ liệu theo tiêu chuẩn công nghiệp CRIP-DM (Cross Industry Standard Process for Data Mining) coi khai phá dữ liệu là một chuỗi lặp và tương tác gồm các bước bắt đầu với dữ liệu thô (raw data) và kết thúc với tri thức tìm được nhằm đáp ứng được yêu cầu của người sử dụng.



Hình 1.8. Quy trình khai phá dữ liệu lặp CRIP-DM

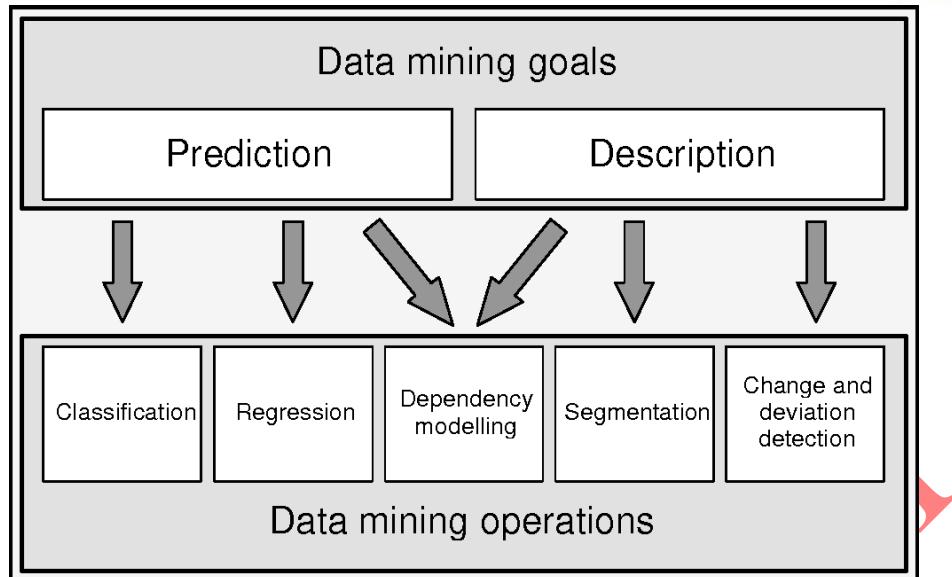
CRISP – DM chia quá trình khai phá dữ liệu thành 6 giai đoạn. Mỗi giai đoạn có một nhiệm vụ khác nhau:

- Giai đoạn 1 (Business understanding): Đây là bước đầu tiên nhằm tìm hiểu nghiệp vụ bài toán và mục tiêu khai phá dữ liệu.
- Giai đoạn 2 (Data understanding): Hiểu dữ liệu để có thể đưa ra cách thức thu thập dữ liệu phù hợp.
- Giai đoạn 3 (Data preparation): Chuẩn bị dữ liệu gồm những nhiệm vụ cơ bản như chọn lọc, làm sạch, tích hợp, biến đổi và giảm chiều dữ liệu
- Giai đoạn 4 (Data modeling): Nhiệm vụ của giai đoạn này là chọn lựa mô hình dữ liệu để đáp ứng được nhiệm vụ khai phá dữ liệu theo mục tiêu đưa ra.
- Giai đoạn 5 (Evaluation): Vì có khá nhiều mô hình khai phá dữ liệu khác nhau có thể được sử dụng cho bài toán hướng tới, tuy nhiên không phải mô hình nào cũng cho hiệu quả cao. Đồng thời, bản thân khi thay đổi các tham số trong một mô hình cũng khiến cho hiệu quả mang lại của mô hình đó khác nhau. Do vậy việc đánh giá mô hình khai phá dữ liệu là đặc biệt cần thiết.
- Giai đoạn 6 (Deployment): Tri thức có được thông qua quá trình khai phá dữ liệu cần được trình bày theo cách mà các bên liên quan có thể sử dụng khi họ muốn, bảo trì và giám sát để thực hiện các hỗ trợ cho tương lai.

#### **1.2.3. Phân loại các bài toán khai phá dữ liệu**

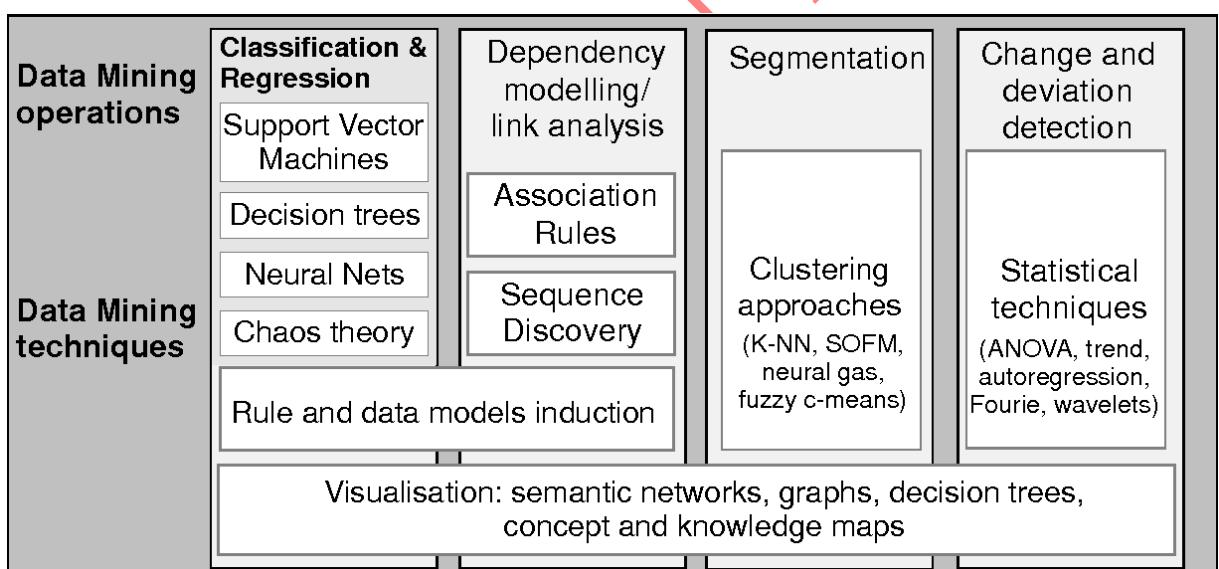
Hai cách phân loại khai phá dữ liệu phổ biến thường đề cập tới đó là 1/ Theo mục đích sử dụng khai phá dữ liệu; 2/ Theo kỹ thuật khai phá dữ liệu.

- Theo mục đích sử dụng thì khai phá dữ liệu được phân thành 2 bài toán chính là khai phá dữ liệu dạng dự đoán và khai phá dữ liệu dạng mô tả. Với mỗi bài toán sẽ được hỗ trợ bởi một tập hợp các kỹ thuật khai phá dữ liệu khác nhau. Cụ thể:
  - Khai phá dữ liệu dạng dự đoán là đưa ra các dự đoán tri thức chưa biết dựa vào các suy diễn trên cơ sở dữ liệu hiện thời. Một số hướng tiếp cận học máy phù hợp cho bài toán khai phá dữ liệu dạng này bao gồm: Phân lớp (Classification), hồi quy (Regression), ...
  - Khai phá dữ liệu mô tả có nhiệm vụ mô tả về các tính chất hoặc đặc tính chung của dữ liệu từ nguồn dữ liệu hiện có. Các kỹ thuật khai phá dữ liệu phù hợp cho bài toán này có thể kể đến là mô hình phụ thuộc (Dependency modelling), phân đoạn (Segmentation), phân cụm (Clustering), khai phá luật kết hợp (Association Rules), ...



Hình 1.9. Phân loại khai phá dữ liệu theo mục đích sử dụng

- Theo kỹ thuật thực thi thì sẽ có rất nhiều thuật toán khai phá dữ liệu được sử dụng nhằm đáp ứng mục tiêu đầu ra của bài toán.



Hình 1.10. Phân loại khai phá dữ liệu theo kỹ thuật thực thi

#### 1.2.4. Một số nền tảng công nghệ hỗ trợ khai phá dữ liệu

Về cơ bản có 3 hướng để khai thác các nền tảng công nghệ đã có trong việc khai phá dữ liệu, đó là:

- Sử dụng các công cụ tích hợp sẵn các chức năng khai phá dữ liệu. Khi đó nhiệm vụ của người dùng là nhập dữ liệu đầu vào cần thiết, chọn chức năng khai phá và nhận kết quả đầu ra. Theo hướng này, người dùng sẽ không phải xử lý về mặt kỹ thuật khai phá nên sẽ dễ dàng tiếp cận và thực hiện khai phá dữ liệu, tuy nhiên hướng này cũng gặp phải một số nhược điểm như một số phần mềm yêu cầu bản

quyền tính phí và lệ thuộc kỹ thuật vào phần mềm, cũng như khó khăn trong việc muốn thay đổi kỹ thuật khai phá mà phần mềm không hỗ trợ. Một số công cụ tích hợp có thể kể đến như:

- Phần mềm SAS, SPSS, MiniTab
- Intelligent Miner (IBM)
- Microsoft data mining tools
- Oracle data mining
- Enterprise miner (SAS institue)
- ...

- Sử dụng trực tiếp các ngôn ngữ lập trình để xây dựng công cụ khai phá dữ liệu (Ví dụ: Matlab, C, Java, Python, R...). Theo hướng này, lập trình viên sẽ là người trực tiếp thực hiện nhiệm vụ khai phá dữ liệu. Ưu điểm là việc tùy chỉnh các giải pháp khai phá dữ liệu về mặt kỹ thuật không còn là trở ngại, tuy nhiên việc này là khó tiếp cận với người dùng không có chuyên môn về công nghệ thông tin.
- Sử dụng các nền tảng mã nguồn mở là một hướng tiếp cận có tính chất lai ghép giữa hai hướng tiếp cận trên. Một mặt hỗ trợ người dùng thực hiện khai phá dữ liệu dựa trên mã nguồn và công cụ đã có được dễ dàng mà chưa cần am hiểu về kỹ thuật xử lý. Mặt khác khi người dùng có nhu cầu muốn tùy chỉnh các giải pháp khai phá dữ liệu về mặt kỹ thuật vẫn hoàn toàn khả thi do mã nguồn mở cho phép người dùng học hỏi và can thiệp trực tiếp vào mã lệnh. Ngoài ra cộng đồng mã nguồn mở cũng hỗ trợ cho người dùng khi có vấn đề xung quanh. Trong các mã nguồn mở phục vụ cho khai phá dữ liệu thì Weka là một nền tảng đã và đang rất phổ biến cho hướng tiếp cận này. Đây cũng là nền tảng mà các chương tiếp theo bài giảng sẽ khai thác cho mục đích minh họa xử lý khai phá dữ liệu.

### **1.3. Khai phá dữ liệu đa phương tiện**

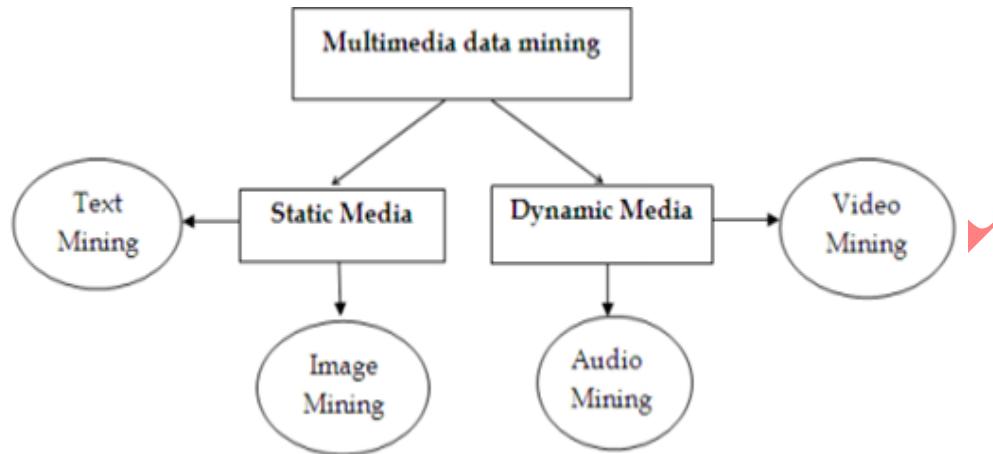
#### ***1.3.1. Khái niệm khai phá dữ liệu đa phương tiện***

Khai phá dữ liệu đa phương tiện (Multimedia data mining) đề cập đến phân tích một lượng lớn dữ liệu đa phương tiện để trích xuất các mẫu dựa trên mối quan hệ của dữ liệu nhằm đưa ra tri thức mới.

Có thể nói, khái niệm khai phá dữ liệu đa phương tiện được kế thừa từ khái niệm khai phá dữ liệu nói chung, được đề cập trong phần trước. Điểm chú ý ở đây là tập trung vào từng loại hình dữ liệu đa phương tiện như văn bản, hình ảnh, âm thanh, video cho mục đích khai phá dữ liệu.

### 1.3.2. Phân loại khai phá dữ liệu đa phương tiện

Việc áp dụng tiêu chí phân loại cho khai phá dữ liệu nói chung hoàn toàn kế thừa áp dụng cho khai phá dữ liệu đa phương tiện nói riêng. Ngoài ra, có một cách tiếp cận phổ biến trong việc phân loại khai phá dữ liệu đa phương tiện đó là dựa theo loại hình dữ liệu đa phương tiện như sau:



Hình 1.11. Phân loại khai phá dữ liệu đa phương tiện theo loại dữ liệu khai phá

Nội dung dưới đây sẽ trình bày tổng quan về các hướng tiếp cận khai phá dữ liệu đa phương tiện theo loại dữ liệu khai phá

#### 1.3.2.1. Khai phá dữ liệu văn bản (Text mining)

- Khái niệm: Khai phá dữ liệu văn bản là việc trích ra các thông tin có ích, chưa được biết đến còn tiềm ẩn trong kho dữ liệu văn bản lớn. Khai phá dữ liệu văn bản là việc thu thập và phân tích dữ liệu bằng các công cụ tự động hoặc bán tự động từ các nguồn tài liệu đã có khác nhau để có được các tri thức mới, chưa được biết đến trước đó.
- Một số kỹ thuật khai phá dữ liệu văn bản: Phân lớp văn bản, phân cụm văn bản, tóm tắt văn bản, truy xuất thông tin văn bản...
- Ứng dụng của khai phá dữ liệu văn bản:
  - Phân tích các câu trả lời, khảo sát.
  - Tự động xử lý tin nhắn, email: Trả lời tự động, lọc thư rác...
  - Phân tích yêu cầu bảo hiểm, phỏng vấn tự động, chuẩn đoán bệnh.
  - Phân tích các đối thủ cạnh tranh trên thị trường bằng cách thu thập dữ liệu từ trang web của họ.

...

#### 1.3.2.2. Khai phá dữ liệu hình ảnh (Image mining)

- Khái niệm: Khai phá dữ liệu hình ảnh nhằm phát hiện tri thức trong cơ sở dữ liệu hình ảnh. Khai phá dữ liệu hình ảnh không chỉ là một phần mở rộng của khai phá dữ liệu đa phương tiện nó còn là một nỗ lực liên ngành dựa trên chuyên môn về thị giác máy tính, khai phá dữ liệu, học máy, cơ sở dữ liệu và cơ sở dữ liệu nhân tạo.
- Một số kỹ thuật khai phá dữ liệu hình ảnh: Phân lớp hình ảnh, phân cụm hình ảnh, khai phá luật kết hợp...
- Ứng dụng của khai phá dữ liệu hình ảnh:
  - o Trong y tế: Phát hiện khối u, chuẩn đoán bệnh,...
  - o Nhận diện khuôn mặt
  - o Xây dựng bản đồ vệ tinh
  - ...

#### *1.3.2.3. Khai phá dữ liệu âm thanh (Audio mining)*

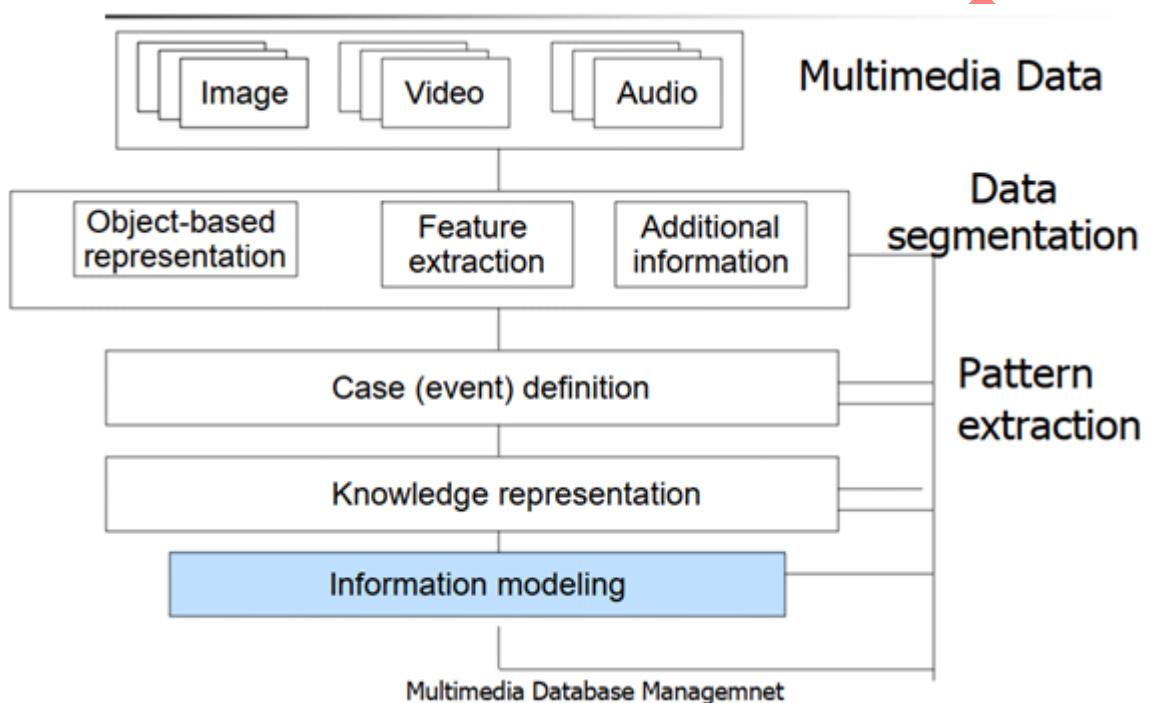
- Khái niệm: Khai phá dữ liệu âm thanh nhằm phát hiện tri thức từ tập hợp các file âm thanh.
- Một số kỹ thuật khai phá dữ liệu âm thanh: phân loại âm thanh, phân cụm âm thanh, ...
- Một số bài toán phổ biến của khai phá dữ liệu âm thanh như: phân loại lời nói, tìm kiếm lời nói, phân tích lời nói, phân tích ngữ âm, nhận dạng giọng nói, lập chỉ mục âm thanh,....

#### *1.3.2.4. Khai phá dữ liệu video (Video mining)*

- Khái niệm: Khai phá dữ liệu video nhằm phát hiện các tri thức thú vị từ nội dung trong video.
- Một số kỹ thuật khai phá dữ liệu video: Phân cụm video, phân lớp video, khai phá luật kết hợp, khai phá xu hướng của video...
- Ứng dụng của khai phá dữ liệu video:
  - o Phân tích tự động video nhằm: giám sát trộm cắp, chăm sóc bệnh nhân và trẻ nhỏ, phát hiện văn hóa giao thông,...
  - o Kiểm soát chất lượng sản xuất tự động: Phát hiện quy trình sản xuất kém chất lượng.
  - o Thư viện số: phân loại video theo chủ đề để cải thiện tìm kiếm hoặc truy xuất thông tin.
  - o ...

### 1.3.3. Quy trình khai phá dữ liệu đa phương tiện

Quy trình khai phá dữ liệu đa phương tiện nói riêng kể thừa từ quy trình khai phá dữ liệu tổng quát nói chung được trình bày trong mục 1.2.2. Điểm khác biệt cụ thể ở đây là tập trung vào khai phá dữ liệu cho từng loại hình dữ liệu đa phương tiện nên dữ liệu cần được trích chọn đặc trưng (Feature extraction) nhằm chuyển đổi dữ liệu đa phương tiện ban đầu sang biểu diễn thông qua các đặc trưng để tổ chức lưu trữ và xử lý tính toán bởi máy tính. Điều này đặc biệt có ý nghĩa với các dạng dữ liệu như hình ảnh, âm thanh, video.



Hình 1.12. Quy trình khai phá dữ liệu đa phương tiện

### 1.3.4. Ứng dụng của khai phá dữ liệu đa phương tiện

Một số ứng dụng phổ biến của khai phá dữ liệu đa phương tiện như sau:

- Thư viện số (Digital Library): Bộ sưu tập dữ liệu số được lưu trữ và duy trì trong thư viện số dưới dạng văn bản hình ảnh, video, âm thanh,... có thể được khai phá nhằm đưa ra những gợi ý đầu sách phù hợp cho độc giả.
- Nhận dạng phương tiện và lưu lượng giao thông và dự đoán thời gian xếp hàng tại các ngã tư từ chuỗi các video (Traffic Video Sequence) thu thập được thông qua camera giám sát là một cách tiếp cận kinh tế cho quy trình giám sát giao thông.
- Phân loại hình ảnh trong lĩnh vực y tế là một nhóm bài toán điển hình mà khai phá dữ liệu đa phương tiện hướng tới. Ví dụ: phân định 3D tự động các khối u não, tự động định vị và xác định các đốt sống trong quét CT,...

- Trong lĩnh vực thương mại điện tử các phản hồi, ý kiến của khách hàng về sản phẩm hoặc dịch vụ là những nguồn dữ liệu được các phương pháp khai phá dữ liệu sử dụng để đưa dự đoán hành vi khách hàng trong tương lai, tư vấn gợi ý hay quảng cáo sản phẩm phù hợp với khách hàng. Các ý kiến, phản hồi của khách hàng có thể ở các dạng dữ liệu khác nhau như âm thanh, văn bản, hình ảnh, video.
- Trong lĩnh vực truyền thông, các đài phát thanh và truyền hình có thể cải thiện chất lượng phát sóng khi áp dụng khai phá dữ liệu đa phương tiện lên các nội dung dữ liệu cần được giám sát.
- Các tổ chức chính phủ, các công ty đa quốc gia, ngân hàng, trung tâm mua sắm,... có rất nhiều dữ liệu lưu trữ dưới dạng đa phương tiện, các nguồn dữ liệu số này nếu được khai thác hợp lý sẽ mang lại rất nhiều giá trị.

#### **1.3.5. Những thách thức trong ứng dụng và nghiên cứu về khai phá dữ liệu**

Trong kỷ nguyên số, chúng ta biết rằng dữ liệu là đối tượng thường xuyên biến động theo thời gian. Khi dữ liệu thay đổi cả về số lượng bộ lẩn tăng giảm thuộc tính cần xử lý sẽ dẫn tới khá nhiều việc cần phải thực hiện theo quy trình khai phá dữ liệu, trong đó ảnh hưởng nhất tới việc đánh giá hiệu quả của các mô hình khai phá dữ liệu cần phải học lại để nâng cao chất lượng khai phá đưa ra tri thức mới.

Bên cạnh đó, khi các dịch vụ số càng phát triển thì nhu cầu gia tăng tính thông minh trong các dịch vụ này càng cần thiết, dẫn tới yêu cầu bổ sung xây dựng những tính năng khai phá dữ liệu nâng cao tiếp theo cho hệ thống nhằm đạt được những tri thức mới thú vị cho các dịch vụ số này. Khi yêu cầu bổ sung thì chúng ta lại cần thực hiện lại toàn bộ quy trình khai phá dữ liệu, cũng là việc cần thời gian, công sức và kinh phí.

Một mặt khác, vấn đề đảm bảo tính an ninh, toàn vẹn, riêng tư trong khai phá dữ liệu cũng là một thách thức khiến việc tiếp cận nguồn dữ liệu cũng gặp khó khăn ngay từ bước thu thập dữ liệu của quy trình khai phá dữ liệu.

### **TỔNG KẾT CHƯƠNG 1**

Nội dung chương này đã trình bày các vấn đề tổng quan những kiến thức cơ bản về khai phá dữ liệu nói chung và khai phá dữ liệu đa phương tiện nói riêng; Các khái niệm có liên quan, ý nghĩa và tầm quan trọng. Đó là:

- Thông tin, dữ liệu và dữ liệu đa phương tiện.
- Tổng quan về khai phá dữ liệu
- Tổng quan về khai phá dữ liệu đa phương tiện.

Trong các chương tiếp theo bài giảng sẽ đi sâu vào trình bày về một số pha chính trong quy trình khai phá dữ liệu gồm : Tiền xử lý dữ liệu đa phương tiện (Chương 2) và Xây dựng mô hình khai phá dữ liệu đa phương tiện (Chương 3).

## CÂU HỎI VÀ BÀI TẬP CHƯƠNG 1

1. Dữ liệu đa phương tiện là gì? Nêu mối quan hệ giữa dữ liệu, thông tin và tri thức.
2. Khai phá dữ liệu là gì? Quá trình khai phá dữ liệu được thực hiện như thế nào?  
Phân loại các bài toán khai phá dữ liệu.
3. Có các giải pháp công nghệ triển khai khai phá dữ liệu nào mà em biết?
4. Các khó khăn và thuận lợi của việc triển khai khai phá dữ liệu trong các sản phẩm số thực tế là gì?
5. Khai phá dữ liệu đa phương tiện là gì? Mối quan hệ giữa khai phá dữ liệu đa phương tiện và khai phá dữ liệu.
6. Theo loại hình dữ liệu đa phương tiện được sử dụng để khai phá thì có những loại bài toán khai phá dữ liệu đa phương tiện nào.

THƯ VIỆT

## CHƯƠNG 2

### TIỀN XỬ LÝ DỮ LIỆU ĐA PHƯƠNG TIỆN

Nội dung chương này tập trung vào trình bày cụ thể về pha đầu tiên trong quy trình khai phá dữ liệu đa phương tiện được đề cập trong chương 1, đó là tiền xử lý dữ liệu và một số công việc liên quan. Nội dung trình bày bao gồm:

- Hiểu về cách biểu diễn và mô tả dữ liệu trên máy tính cho mục đích khai phá dữ liệu
- Khái niệm về tiền xử lý dữ liệu đa phương tiện
- Tầm quan trọng của tiền xử lý dữ liệu đa phương tiện
- Các nhiệm vụ chính của tiền xử lý dữ liệu đa phương tiện: trích chọn đặc trưng, làm sạch, tích hợp, biến đổi, thu giảm dữ liệu
- Tiền xử lý dữ liệu đa phương tiện với Weka

#### 2.1. Biểu diễn và mô tả tập dữ liệu trên máy tính cho mục đích khai phá dữ liệu

##### 2.1.1. Biểu diễn tập dữ liệu

Các loại hình dữ liệu đa phương tiện như văn bản, hình ảnh, âm thanh, video phổ biến được tồn tại dưới dạng các tệp tin rời rạc trên máy tính. Để thực hiện khai phá dữ liệu được từ tập hợp các file này thì cần phải có một cách thức để biểu diễn dữ liệu cho máy tính. Một số cách biểu diễn tập dữ liệu phổ biến có thể kể đến như: Kiểu bản ghi (Record), kiểu đồ thị (Graph), biểu diễn tập dữ liệu có trật tự (Ordered)...

Tid	Refund	Marital Status	Taxable Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

(a) Record data.

TID	ITEMS
1	Bread, Soda, Milk
2	Beer, Bread
3	Beer, Soda, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Soda, Diaper, Milk

(b) Transaction data.

Projection of x Load	Projection of y Load	Distance	Load	Thickness
10.23	5.27	15.22	27	1.2
12.65	6.25	16.22	22	1.1
13.54	7.23	17.34	23	1.2
14.27	8.43	18.45	25	0.9

(c) Data matrix.

team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0
Document 2	0	7	0	2	1	0	0	3	0
Document 3	0	1	0	0	1	2	2	0	3

(d) Document-term matrix.

Hình 2.1. Minh họa một số cách biểu diễn tập dữ liệu

Trong số những cách này, bài giảng tiếp cận cách biểu diễn tập dữ liệu phổ biến nhất đó là kiểu bản ghi. Theo cách biểu diễn dữ liệu dưới dạng bản ghi, một tập dữ liệu (dataset) là một tập hợp các đối tượng (objects) và các thuộc tính (attributes) của chúng, được minh họa như sau:

$$\mathbf{D} = \left( \begin{array}{c|cccc} & X_1 & X_2 & \cdots & X_d \\ \hline \mathbf{x}_1 & x_{11} & x_{12} & \cdots & x_{1d} \\ \mathbf{x}_2 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_n & x_{n1} & x_{n2} & \cdots & x_{nd} \end{array} \right)$$

	Sepal length	Sepal width	Petal length	Petal width	Class
	X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>4</sub>	X <sub>5</sub>
x <sub>1</sub>	5.9	3.0	4.2	1.5	Iris-versicolor
x <sub>2</sub>	6.9	3.1	4.9	1.5	Iris-versicolor
x <sub>3</sub>	6.6	2.9	4.6	1.3	Iris-versicolor
x <sub>4</sub>	4.6	3.2	1.4	0.2	Iris-setosa
x <sub>5</sub>	6.0	2.2	4.0	1.0	Iris-versicolor
x <sub>6</sub>	4.7	3.2	1.3	0.2	Iris-setosa
x <sub>7</sub>	6.5	3.0	5.8	2.2	Iris-virginica
x <sub>8</sub>	5.8	2.7	5.1	1.9	Iris-virginica
:	:	:	:	:	:
x <sub>149</sub>	7.7	3.8	6.7	2.2	Iris-virginica
x <sub>150</sub>	5.1	3.4	1.5	0.2	Iris-setosa

Hình 2.2. Minh họa biểu diễn tập dữ liệu hoa diên vĩ Iris dưới dạng bản ghi

Trong đó:

- Mỗi thuộc tính mô tả một đặc điểm của một đối tượng (Ví dụ: Sepal length, Sepal width, Petal length, Petal width, Class).
- Một tập các giá trị của các thuộc tính mô tả một đối tượng (Ví dụ: x1, x2,...). Khái niệm “đối tượng” còn được tham chiếu đến với các tên gọi khác như bản ghi (record), điểm dữ liệu (data point), trường hợp (case), mẫu (sample), thực thể (entity) hoặc ví dụ /thể hiện (instance).

### 2.1.2. Mô tả thông tin từ tập dữ liệu

Mục đích của việc mô tả thông tin từ dữ liệu nhằm hiểu rõ về dữ liệu có được (chiều hướng chính/ trung tâm, sự biến thiên, sự phân bố). Điều này có thể được thực hiện thông qua các công thức độ đo toán học về mô tả dữ liệu như : Giá trị cực tiểu/ cực đại (min/max), giá trị xuất hiện nhiều nhất (mode), giá trị trung bình (mean), Giá trị trung vị (median), sự biến thiên (variance), độ lệch chuẩn (standard deviation), các ngoại lai (outliers)...

**Mean**

$$\bar{x} = \frac{\sum_{i=1}^N x_i}{N} = \frac{x_1 + x_2 + \dots + x_N}{N}$$
  

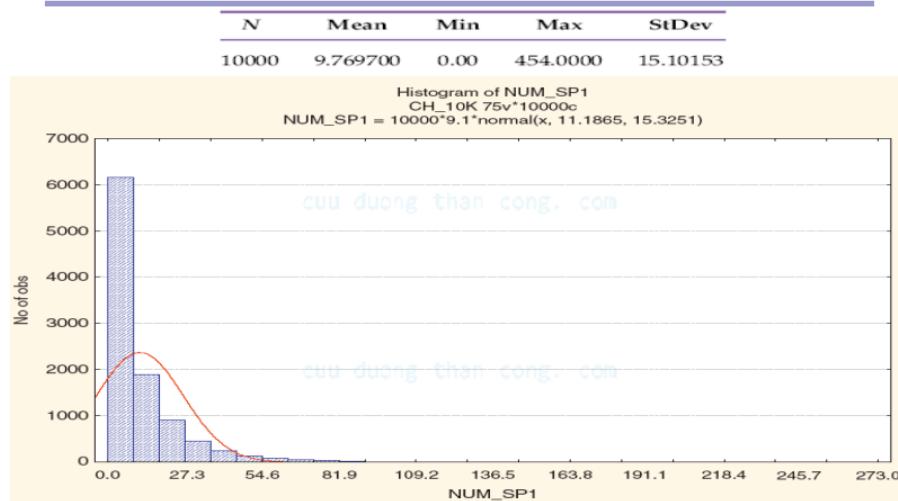
**Median**  $Median = \begin{cases} x_{\lceil N/2 \rceil} & \text{if } N \text{ odd} \\ (x_{N/2} + x_{N/2+1})/2 & \text{if } N \text{ even} \end{cases}$

Hình 2.3. Ví dụ mô tả thông tin từ tập dữ liệu qua một số công thức độ đo

Việc mô tả thông tin từ tập dữ liệu bằng việc thống kê các kết quả tính toán theo độ đo nhiều khi gây khó theo dõi, do vậy phương pháp mô tả dữ liệu bằng hiển thị hóa dữ liệu (visualization) dưới dạng biểu đồ được đưa ra. Có một số dạng biểu đồ điển hình như biểu đồ histogram, đồ thị rải rác (scatter plot) hay được sử dụng.

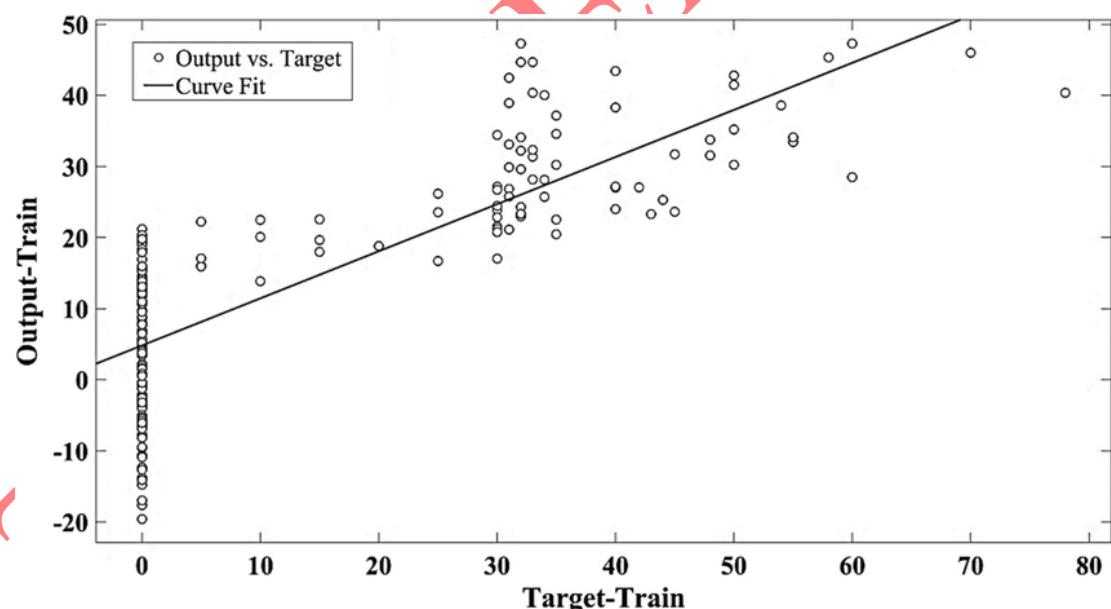
- Biểu đồ histogram: là cách biểu diễn dựa trên đồ thị, được sử dụng rất phổ biến nhằm hiển thị các mô tả thống kê xuất hiện theo một thuộc tính nào đó.

## Mô tả dữ liệu, so sánh với phân bố chuẩn (chủ yếu trong miền [0,10])



Hình 2.4. Minh họa mô tả thông tin từ tập dữ liệu qua biểu đồ histogram

- Biểu đồ đồ thị rải rác (scatter plot): Cho phép hiển thị quan hệ 2 chiều (giữa 2 thuộc tính) của dữ liệu nhằm quan sát trực quan các nhóm điểm, các ngoại lai... Trong đó mỗi cặp giá trị của 2 thuộc tính được xét tương ứng với 2 tọa độ của điểm được hiển thị trên mặt phẳng.



Hình 2.5. Minh họa mô tả thông tin từ tập dữ liệu qua biểu đồ scatter plot

### 2.2. Khái niệm tiền xử lý dữ liệu đa phương tiện

- Tiền xử lý dữ liệu là quá trình xử lý dữ liệu thô nhằm cải thiện chất lượng dữ liệu và từ đó cải thiện chất lượng của kết quả khai phá dữ liệu.

Dữ liệu thô có thể ở dạng có cấu trúc, bán cấu trúc hoặc phi cấu trúc được đưa vào từ nhiều nguồn như thu thập chủ động, các hệ thống xử lý tập tin hay các hệ thống cơ sở dữ liệu.

- Khái niệm tiền xử lý dữ liệu đa phương tiện kế thừa từ khái niệm tiền xử lý dữ liệu cơ sở, trong đó nguồn dữ liệu đầu vào đa dạng dưới dạng văn bản, hình ảnh, âm thanh, video.

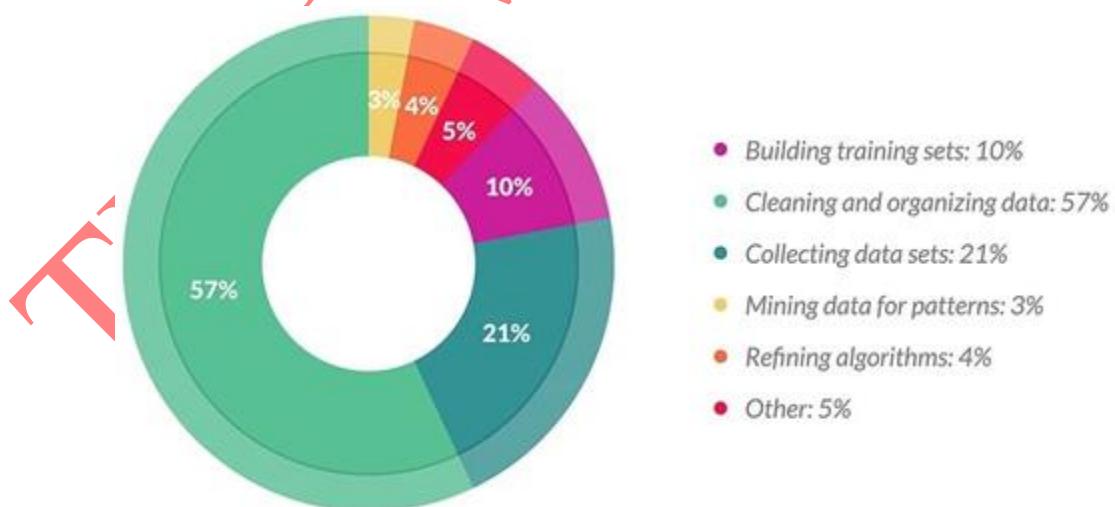
### 2.3. Tầm quan trọng của tiền xử lý dữ liệu đa phương tiện

Chất lượng của việc khai phá dữ liệu phụ thuộc chính vào chất lượng nguồn dữ liệu đầu vào. Nói cách khác nếu không có dữ liệu tốt thì không thể có kết quả khai phá tốt. Dữ liệu được cho là có chất lượng cao nếu phù hợp với mục đích sử dụng trong điều hành, ra quyết định và lập kế hoạch.

Một số tính chất điển hình của dữ liệu chất lượng:

- Tính chính xác (accuracy): Giá trị được ghi nhận đúng với giá trị thực.
- Tính hiện hành (Currency/ Timeliness): Giá trị được ghi nhận không bị lỗi thời.
- Tính toàn vẹn (Completeness): Tất cả các giá trị dành cho một biến/ thuộc tính đều được ghi nhận.
- Tính nhất quán (Consistency): Tất cả giá trị dữ liệu đều được biểu diễn như nhau trong tất cả trường hợp.

Theo các nghiên cứu đã có phần lớn công việc xây dựng một kho dữ liệu là trích chọn và dọn dẹp dữ liệu.



Hình 2.6. Thống kê tỉ trọng các công việc cần thực hiện trong quá trình khai phá dữ liệu

### 2.4. Những nhiệm vụ chính của tiền xử lý dữ liệu đa phương tiện

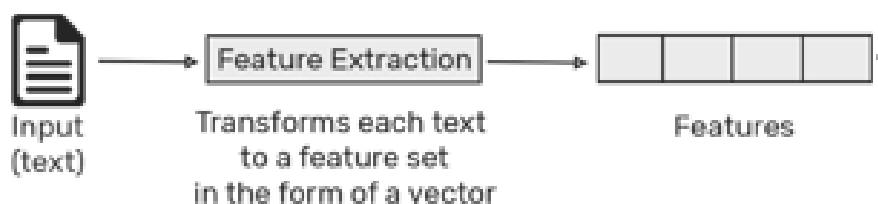
### 2.4.1. Trích chọn đặc trưng dữ liệu

Như đã đề cập trong mục 2.1.1, việc biểu diễn tập dữ liệu đa phương tiện tiếp cận trong bài giảng dưới dạng bản ghi. Tuy nhiên các loại hình dữ liệu đa phương tiện như văn bản, hình ảnh, âm thanh, video phổ biến được tồn tại dưới dạng các tệp tin rời rạc trên máy tính. Do vậy để thực hiện khai phá dữ liệu được từ tập hợp các tệp tin này thì cần phải có cách thức để biểu diễn tập dữ liệu về dạng bản ghi. Trích chọn đặc trưng chính là nhiệm vụ mấu chốt cho quá trình biểu diễn tập dữ liệu đa phương tiện dưới dạng bản ghi để có thể thực hiện khai phá dữ liệu sau này.

Tùy thuộc vào loại hình dữ liệu đa phương tiện tiếp cận là văn bản, âm thanh, hình ảnh hay video mà có các kỹ thuật trích chọn đặc trưng khác nhau. Nội dung sau đây sẽ chia việc làm này theo từng loại hình dữ liệu đa phương tiện.

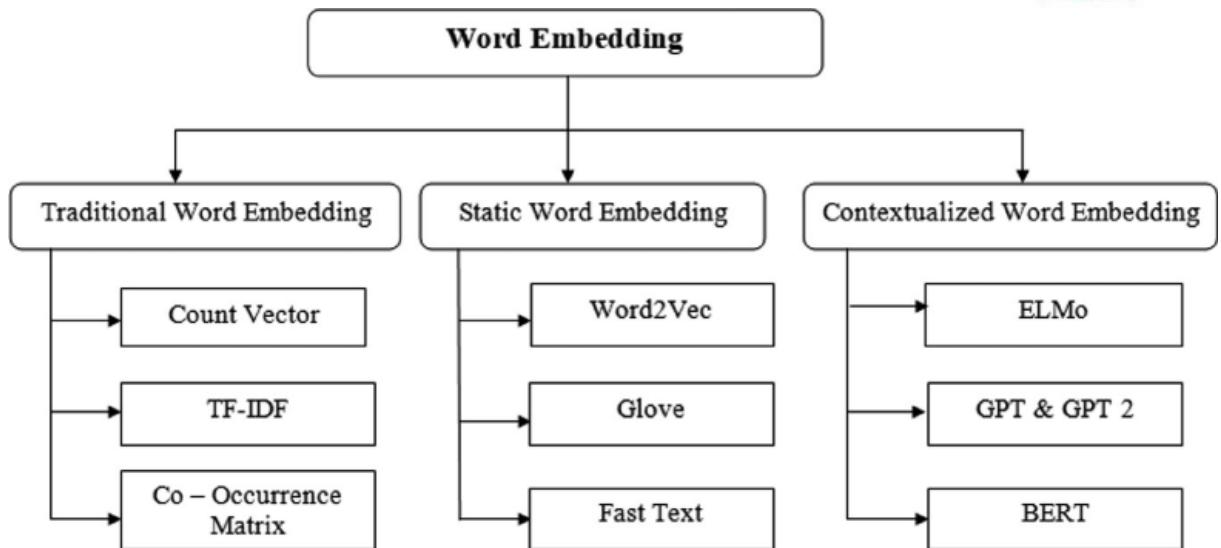
#### 2.4.1.1. Trích chọn đặc trưng văn bản (Text feature extraction)

Phương pháp tiêu chuẩn để biểu diễn văn bản đó là biểu diễn các văn bản về dạng một véc tơ đặc trưng, điều này còn được biết đến là vector hóa văn bản (vectorization). Trong đó mỗi đặc trưng đại diện cho một thông tin cần thiết để miêu tả về văn bản, mỗi văn bản khác nhau trong tập văn bản sẽ nhận một bộ giá trị khác nhau cho các đặc trưng.



Hình 2.7. Nguyên tắc hoạt động của trích chọn đặc trưng văn bản

Trong lĩnh vực xử lý ngôn ngữ tự nhiên NLP (Natural Language Processing), chúng ta dùng thuật ngữ nhúng từ (Word Embedding) để chỉ công việc vector hóa văn bản này. Các kỹ thuật trích chọn đặc trưng văn bản có thể được phân loại theo 3 nhóm phương pháp vector hóa văn bản đó là: 1/Phương pháp vector hóa văn bản cổ điển (Traditional Word Embedding); 2/Phương pháp vector hóa văn bản tĩnh (Static Word Embedding); 3/Phương pháp vector hóa văn bản theo ngữ cảnh (Contextualized Word Embedding).

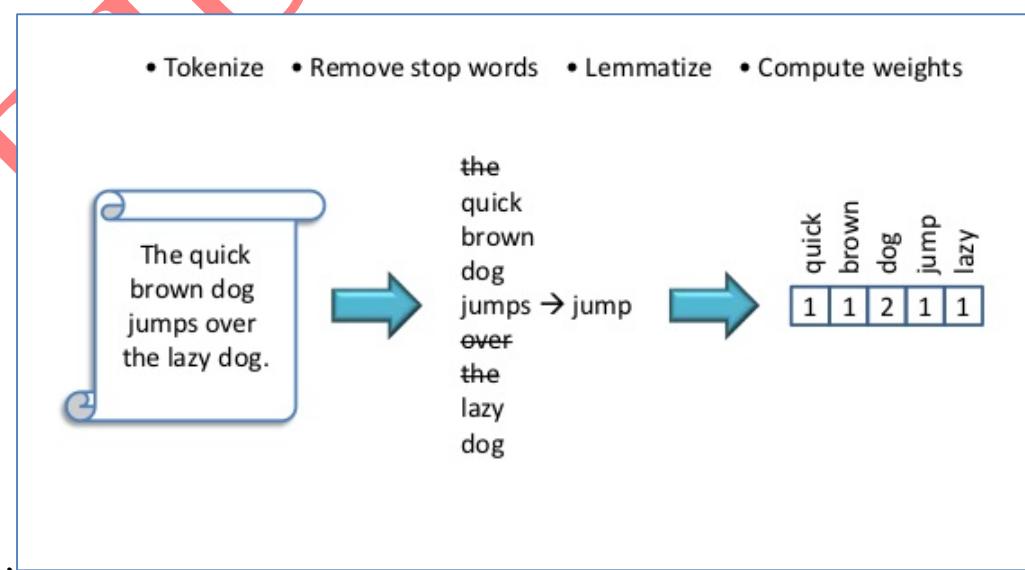


Hình 2.8. Phân loại các kỹ thuật vector hóa văn bản

Trong số các phương pháp cụ thể cho ba nhóm phương pháp kể trên, bài giảng sẽ tập trung trình bày một số phương pháp trích chọn đặc trưng văn bản nền tảng phổ biến: Bag of words, TF-IDF. Nội dung dưới đây sẽ trình bày khái quát về các phương pháp này.

- **Bag of words:**

Phương pháp này chuyển đổi biểu diễn thông tin từ dạng văn bản qua một véc tơ đặc trưng theo 4 bước : 1/ Tách văn bản thành các tập hợp các từ (Tokenize); 2/ Loại bỏ những từ dừng không có ý nghĩa về ngữ nghĩa (Remove stop words); 3/ Chuyển đổi tập hợp các từ còn lại về dạng thức từ gốc (Lemmatize) để tạo lập thành 1 túi từ đóng vai trò là tập các đặc trưng của văn bản; 4/ Tính trọng số cho các từ trong túi từ, khi đó mỗi từ nhận một giá trị. Giá trị này thể hiện tần suất xuất hiện của từ trong văn bản chứa nó, ký hiệu là  $TF(t, d)$  - Term Frequency.



### Hình 2.9. Các bước trích chọn đặc trưng văn bản theo kỹ thuật Bag of words

Trong đó:

- Bước 1 có thể lựa chọn tách từ trong văn bản theo dạng từ đơn (1-gram), từ ghép đôi (bi-gram), từ ghép 3 (tri-gram) hay từ ghép n (n-gram) tùy thuộc vào số lượng từ được tách để tạo lập tập đặc trưng.
- TF(t,d) có thể tính theo một trong số các công thức sau:

weighting scheme	TF weight
binary	0, 1
raw count	$f_{t,d}$
term frequency	$f_{t,d} / \sum_{t' \in d} f_{t',d}$
log normalization	$1 + \log(f_{t,d})$



Ưu và nhược điểm của phương pháp Bag of words:

- **Ưu điểm:** Đơn giản trong việc xây dựng véc tơ đặc trưng cho văn bản. Phương pháp này phù hợp với bài toán phân loại câu văn bản.
- **Nhược điểm :** 1/ Mặc dù khi tiến hành bước 1 ta có thể chọn các chiến lược khác nhau cho việc tách từ (n-gram) nhưng về cơ bản xét một cách tổng thể thì kỹ thuật này không quan tâm tới trật tự từ trong túi từ nên với mỗi cách lựa chọn thứ tự từ khác nhau sẽ tạo lập ra một véc tơ đặc trưng khác nhau cho văn bản. Việc không nhất quán trong hình thành véc tơ đặc trưng cho tập dữ liệu văn bản sẽ ảnh hưởng trực tiếp tới quá trình khai phá dữ liệu sau này; 2/ Đối với văn bản dài thì số lượng từ trong túi từ được xây dựng theo kỹ thuật này là rất lớn, điều này khiến cho vector đặc trưng sẽ có kích thước rất lớn. Do vậy, việc biểu diễn tập văn bản theo phương pháp bản ghi sử dụng kỹ thuật Bag of words này sẽ tạo lập một ma trận rất lớn là điều chúng ta cần phải cân nhắc về thời gian thực hiện khai phá dữ liệu; 3/ Vector đặc trưng của mỗi văn bản chỉ đánh giá được mức độ quan trọng của từng đặc trưng trong văn bản chứa chúng, mà chưa quan tâm tới mối tương quan trên toàn bộ tập văn bản.

#### - TF-IDF (Term Frequency–Inverse Document Frequency)

Phương pháp TF-IDF phát triển trên cơ sở kỹ thuật Bag of words. Tuy nhiên việc xác định vector đặc trưng của văn bản sẽ được hình thành trên 2 thông số : 1/ Mức độ

quan trọng của đặc trưng trong văn bản chứa nó – TF(t,d) và 2/ Mức độ quan trọng của đặc trưng đó trên tập hợp các văn bản của hệ thống - IDF(t,D).

$$\text{tfidf}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D)$$

	Doc 1	Doc 2	...	Doc n
Term(s) 1	12	2	...	1
Term(s) 2	0	1	...	0
...	...	...	...	
Term(s) n	0	6	...	3

Trong đó:

- TF(t,d) có thể tính theo một trong số các công thức như kỹ thuật Bag of words.
- IDF(t,D) được tính theo công thức sau:

$$\text{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

Như vậy TF-IDF thể hiện trọng số của mỗi từ / n-gram theo ngữ cảnh văn bản. TF-IDF sẽ có giá trị tăng tỷ lệ thuận với số lần xuất hiện của từ trong văn bản và số văn bản có chứa từ đó trên toàn bộ tập văn bản. Phương pháp loại này giúp cho TF-IDF có tính phân hóa cao hơn so với phương pháp trước, do vậy phương pháp này phù hợp với bài toán phân loại các tài liệu văn bản.

Ví dụ: Cho 3 tài liệu văn bản, mỗi tài liệu văn bản có 1 câu như sau:

A: This pasta is very tasty and affordable.

B: This pasta is not tasty and is affordable.

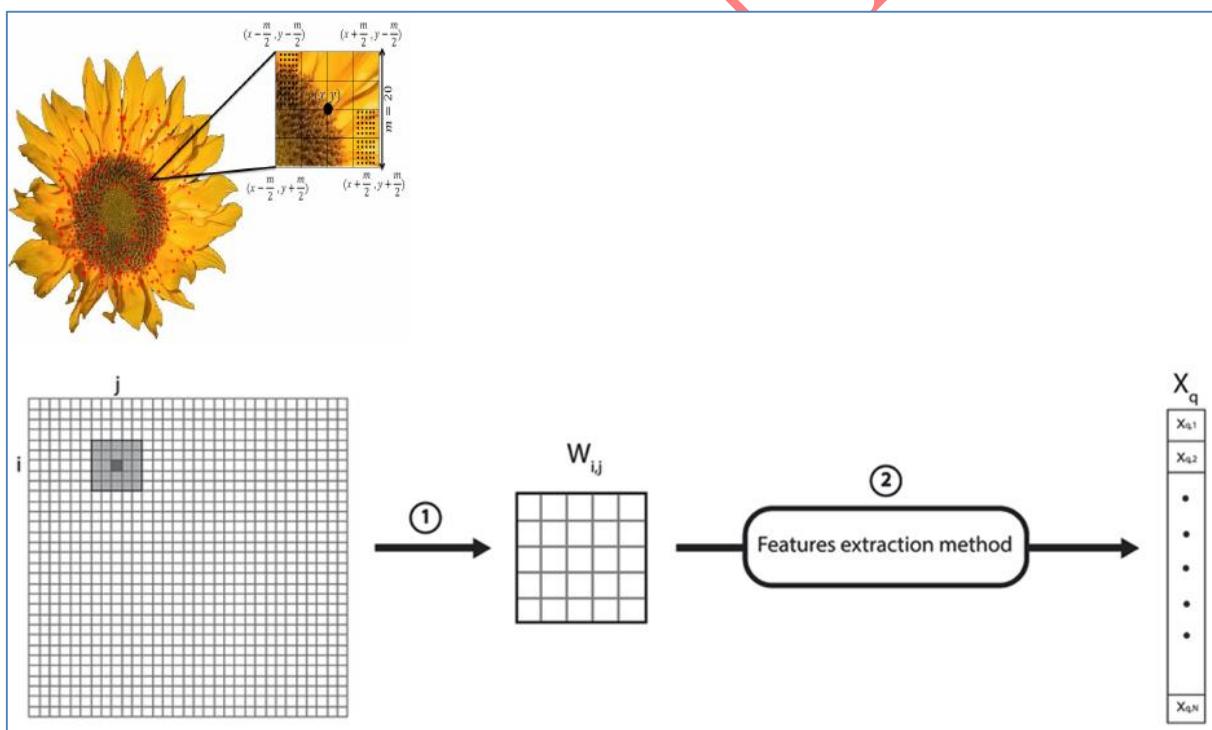
C: This pasta is very very delicious.

Áp dụng phương pháp TF-IDF sẽ tạo lập được vector đặc trưng cho mỗi tài liệu văn bản dưới đây:

Word	TF			TF * IDF		
	A	B	C	A	B	C
this	1/7	1/8	1/6	$\log(3/3)=0$	0	0
pasta	1/7	1/8	1/6	$\log(3/3)=0$	0	0
is	1/7	2/8	1/6	$\log(3/3)=0$	0	0
Very	1/7	0	2/6	$\log(3/2)=0.176$	0.025	0.058
tasty	1/7	1/8	0	$\log(3/2)=0.176$	0.025	0.022
and	1/7	1/8	0	$\log(3/2)=0.176$	0.025	0.022
affordable	1/7	1/8	0	$\log(3/2)=0.176$	0.025	0.022
not	0	1/8	0	$\log(3/1)=0.477$	0	0.0596
delicious	0	0	1/6	$\log(3/1)=0.477$	0	0.079

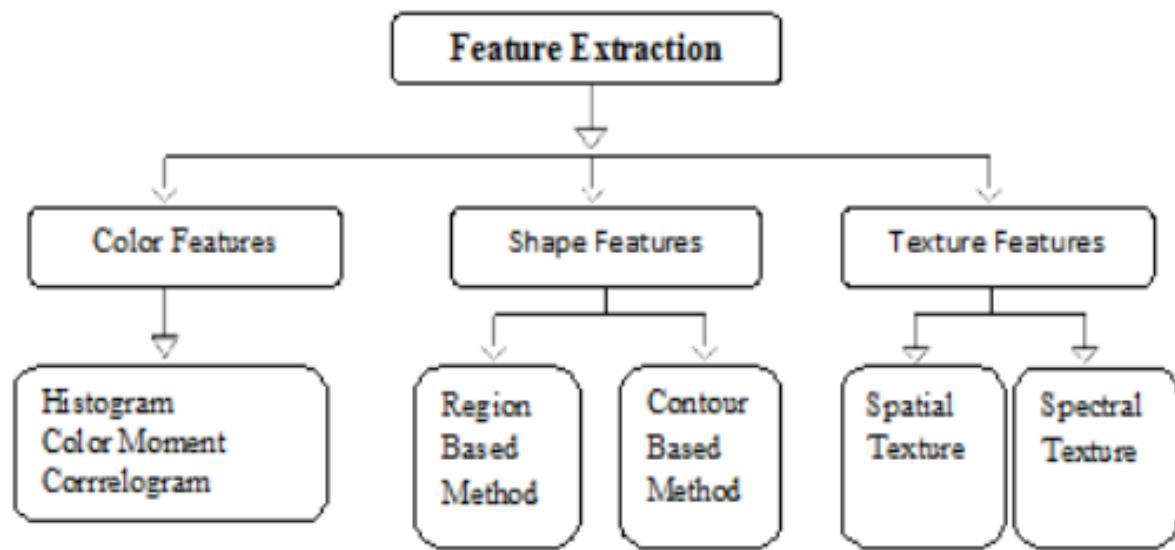
#### 2.4.1.2. Trích chọn đặc trưng hình ảnh (Image feature extraction)

Nguyên tắc hoạt động của trích chọn đặc trưng hình ảnh: Có cùng nguyên tắc hoạt động với trích chọn đặc trưng văn bản, điểm khác biệt duy nhất là dữ liệu nhận vào là hình ảnh biểu diễn dưới dạng ma trận điểm ảnh.



Hình 2.10. Nguyên tắc hoạt động của trích chọn đặc trưng hình ảnh

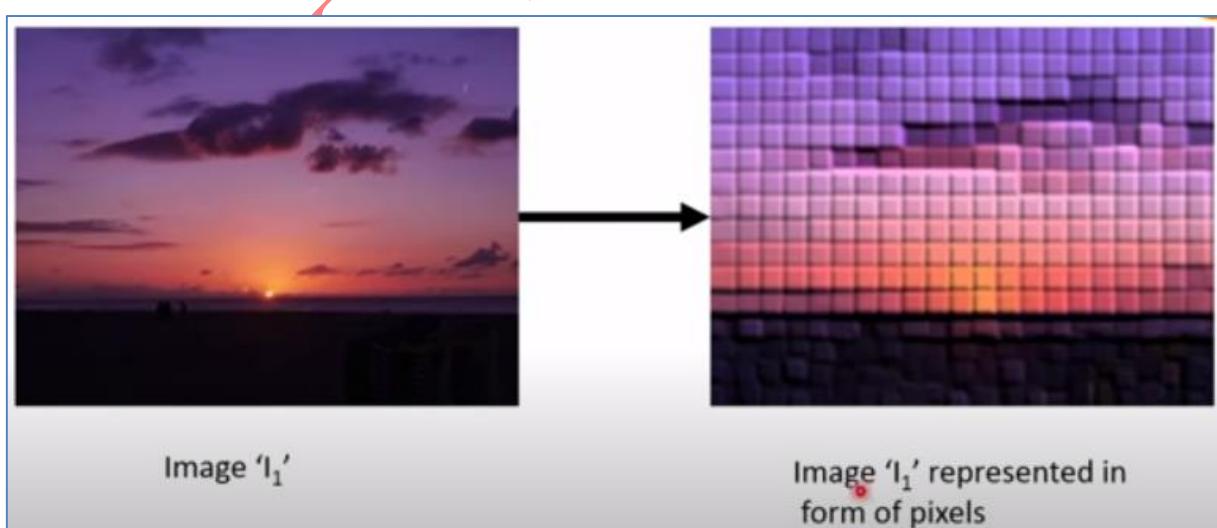
Có ba loại đặc trưng chính của hình ảnh thường được khai thác là: Các đặc trưng màu sắc (Color features), đặc trưng kết cấu (Texture features), đặc trưng hình dạng (Shape features). Mỗi loại đặc trưng sẽ có những phương pháp khác nhau để thực hiện.

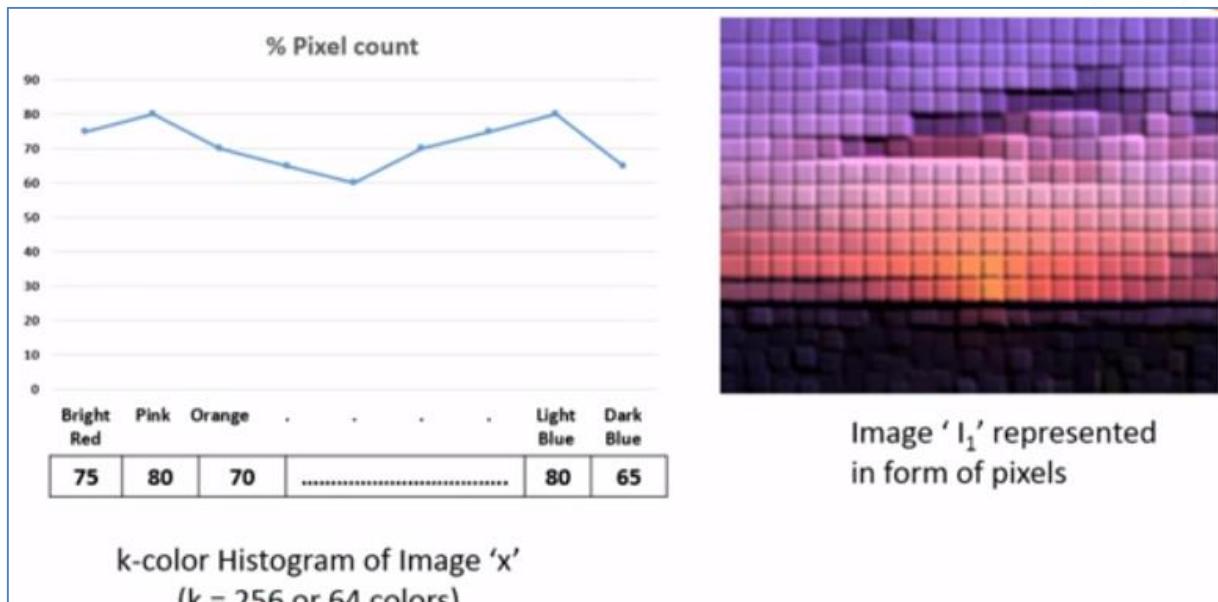


Hình 2.11. Ba loại đặc trưng của hình ảnh và các kỹ thuật thực hiện chính

Trong đó, đặc trưng về màu sắc thường được coi là dễ tiếp cận và khai thác phổ biến, đây cũng là nhóm đặc trưng được khai thác trong phạm vi bài giảng này. Phương pháp trích chọn đặc trưng hình ảnh cơ bản điển hình là k-Color histogram (Phổ biến nhất k = 64 hoặc 256).

Về cơ bản, phương pháp k-Color histogram cho phép biểu diễn phân phối của các màu trên ảnh bằng cách thống kê phần trăm số lượng các pixel có giá trị nằm trong một khoảng màu nhất định cho trước. Như vậy việc trích chọn đặc trưng theo phương pháp k-Color histogram sẽ thông qua 2 bước: 1/ Biểu diễn hình ảnh dưới dạng pixel; 2/ Tính phần trăm số lượng các pixel có giá trị nằm trong một mức sáng nhất định cho trước.





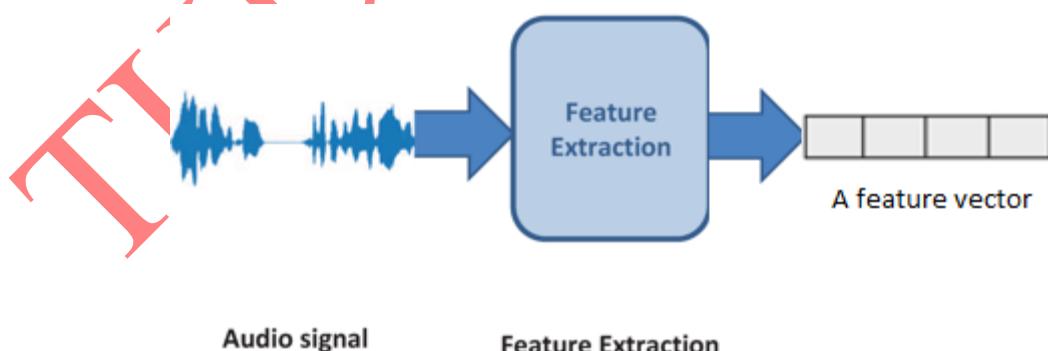
Hình 2.12. Minh họa quá trình trích chọn đặc trưng ảnh theo phương pháp k-color histogram

Biểu đồ k-color ở hình minh họa cho biết 75% điểm ảnh trong bức ảnh có màu Bright Red, tương tự lý giải cho các con số khác trong biểu đồ.

Tùy thuộc vào việc thiết lập giá trị  $k$  thì vector đặc trưng của bức ảnh được hình thành với số chiều tương ứng. Giá trị của mỗi phần tử trong vector đặc trưng sẽ tương ứng với phần trăm số lượng các pixel theo trực hoành trong biểu đồ k-color trên.

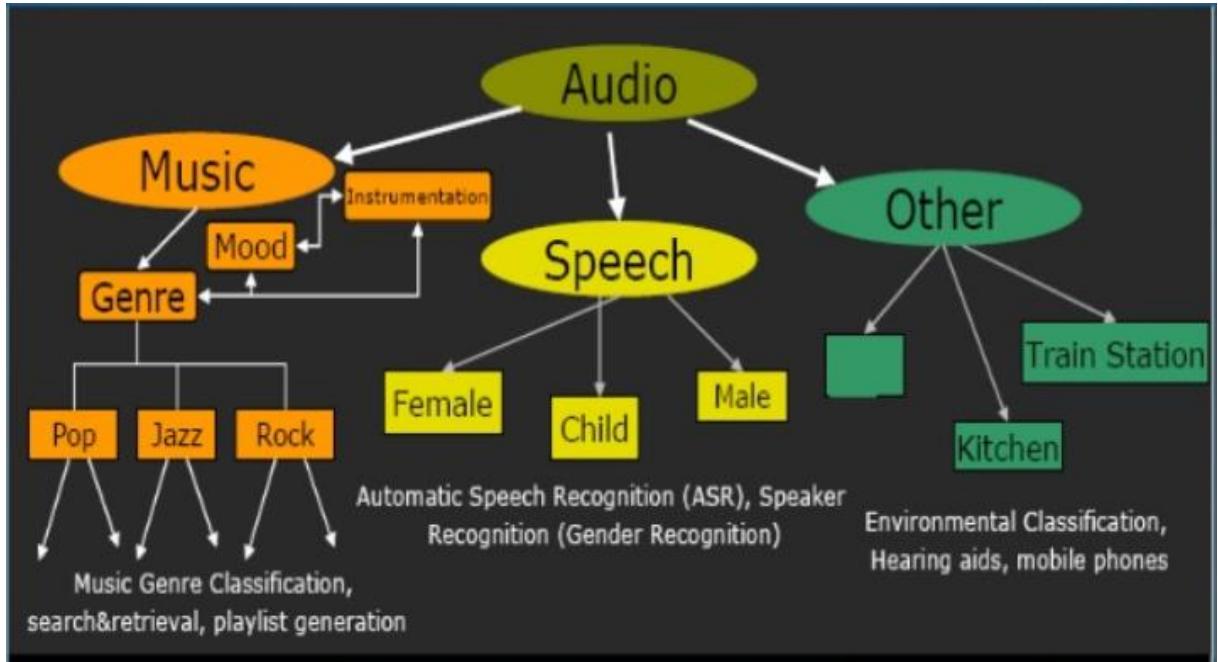
#### 2.4.1.3. Trích chọn đặc trưng âm thanh (Audio feature extraction)

Nguyên tắc hoạt động của trích chọn đặc trưng âm thanh: Có cùng nguyên tắc hoạt động với trích chọn đặc trưng văn bản, điểm khác biệt duy nhất là dữ liệu nhận vào là tín hiệu âm thanh.



Hình 2.13. Nguyên tắc hoạt động của trích chọn đặc trưng âm thanh

Trong đó, tín hiệu âm thanh nghe được cũng được phân thành một số loại khác nhau đó là: âm nhạc, lời nói hay những âm thanh từ môi trường.



Hình 2.14. Phân loại tín hiệu âm thanh nghe được

Các phương pháp trích chọn đặc trưng âm thanh được chia thành 2 hướng tiếp cận là: 1/ Dựa trên siêu dữ liệu (Metadata) của file âm thanh; 2/ Dựa vào nội dung file âm thanh. Tùy thuộc vào mỗi loại tín hiệu âm thanh khác nhau để lựa chọn hướng tiếp cận phù hợp.

- **Trích chọn đặc trưng âm thanh dựa trên metadata:** Các thông tin đi kèm được tạo ra theo file âm thanh chính là các siêu dữ liệu miêu tả về file âm thanh đó sẽ được dùng như các đặc trưng của âm thanh.

Ví dụ một số siêu dữ liệu điển hình cho bài hát bao gồm:

- Tên bài hát
- Tên tác giả
- Tên ca sĩ
- Nhạc sĩ sáng tác
- Năm sáng tác
- Thời lượng bài hát
- Người dùng bài hát
- Xếp hạng bài hát
- Số lượt xem bài hát

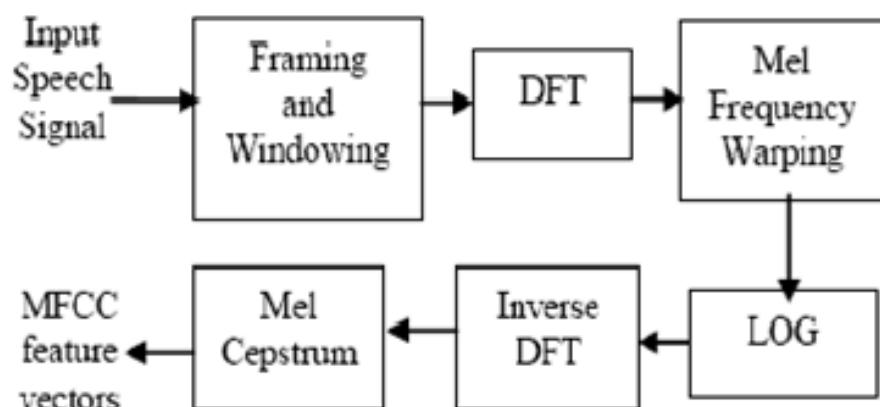
Như vậy tiếp cận phương pháp trích chọn đặc trưng âm thanh dựa trên metadata thì giá trị của các đặc trưng trong vector đặc trưng âm thanh sẽ do người tạo ra âm thanh thiết lập giá trị ban đầu.

- **Trích chọn đặc trưng âm thanh dựa vào nội dung:** Với hướng tiếp cận này, một số phương pháp xử lý tín hiệu âm thanh được sử dụng như :

- Mã hóa dự đoán tuyến tính ( Linear Predictive Coding – LPC)
- Hệ số dự đoán tuyến tính ( Linear Predictive Cepstral Coefficients – LPCC)
- Hệ số cepstrum tần số Mel ( Melfrequency Cepstrum Coefficients – MFCC)
- Bộ lọc năng lượng logarit (Log Energy Filter Coefficient – LEFC),

Nội dung dưới đây sẽ trình bày cụ thể về phương pháp MFCC, một phương pháp trích xuất đặc trưng phổ biến nhất được áp dụng cho tín hiệu âm thanh dưới dạng lời nói. Việc tìm hiểu các phương pháp trích chọn đặc trưng âm thanh dựa vào nội dung khác được coi như một bài tập của phần này.

Quá trình trích chọn đặc trưng với MFCC như sau:



**Hình 2.15. Quá trình trích chọn đặc trưng âm thanh theo phương pháp MFCC**

Các bước trong quá trình trích chọn đặc trưng MFCC được miêu tả như sau:

- Framing and Windowing : Cắt đoạn tín hiệu âm thanh đầu vào thành các mẫu tín hiệu có thời lượng nhỏ, gọi là các frame.
- DFT: biến đổi fourier rời rạc đối với từng frame. Qua phép biến đổi này, tín hiệu sẽ được đưa về không gian miền tần số. Kết quả của quá trình biến đổi Fourier rời rạc thể hiện năng lượng của tín hiệu ở những dải tần số khác nhau.
- Mel Frequency Warping: Do mức độ cảm nhận của tai người không có sự nhạy cảm như nhau đối với mọi dải tần số do vậy tại giai đoạn này sẽ chuyển đổi biểu

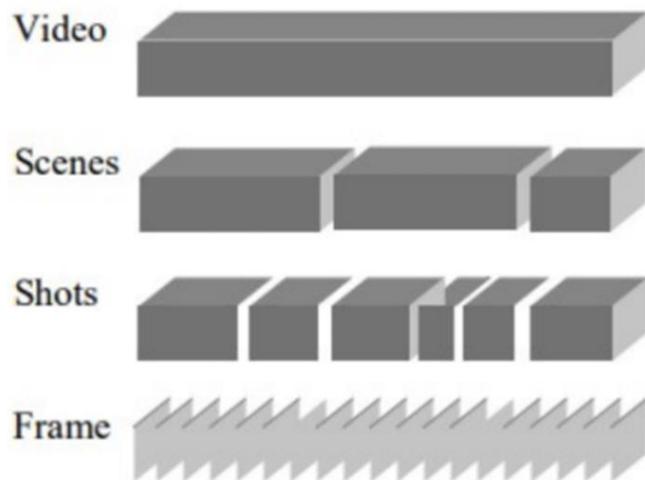
diễn tín hiệu từ không gian miền tần số trước đó san thang đo tần số Mel để chuyển về tín hiệu phù hợp với cảm nhận của tai người.

- LOG: Tại bước này tính toán logarit của bình phương độ lớn những hệ số tại công ra của bộ lọc. Bước này có hai nguyên nhân là do tai người nhạy cảm với âm thanh cường độ thấp hơn và làm các giá trị đặc trưng nhỏ đi, tiện cho việc tính toán.
- Inverse DFT: Sau khi thực hiện biến đổi Fourier rời rạc thì dãy tín hiệu theo thời gian đã được chuyển thành phổ tần số và việc áp dụng các băng lọc tần số mel giúp cô đọng phổ tần số về một hệ số nhất định (bằng với số băng lọc). Các hệ số này thể hiện các đặc trưng của nguồn âm thanh như tần số cơ bản, xung âm thanh,... Việc thực hiện biến đổi Fourier ngược giúp tách biệt các đặc trưng về nguồn âm và bộ máy phát âm từ các hệ số.
- Mel cepstrum: Giai đoạn này sẽ trích lọc ra từ các hệ số mel thu được ở bước trước nhằm tạo ra vector đặc trưng MFCC cho file âm thanh. Một số đặc trưng MFCC điển hình được trích lọc gồm: 12 giá trị đặc trưng phổ Mel được biến đổi Fourier ngược, 12 giá trị delta phổ, 12 giá trị double delta phổ, 1 giá trị delta mức năng lượng, 1 giá trị double delta mức năng lượng.

#### *2.4.1.4. Trích chọn đặc trưng video (Video feature extraction)*

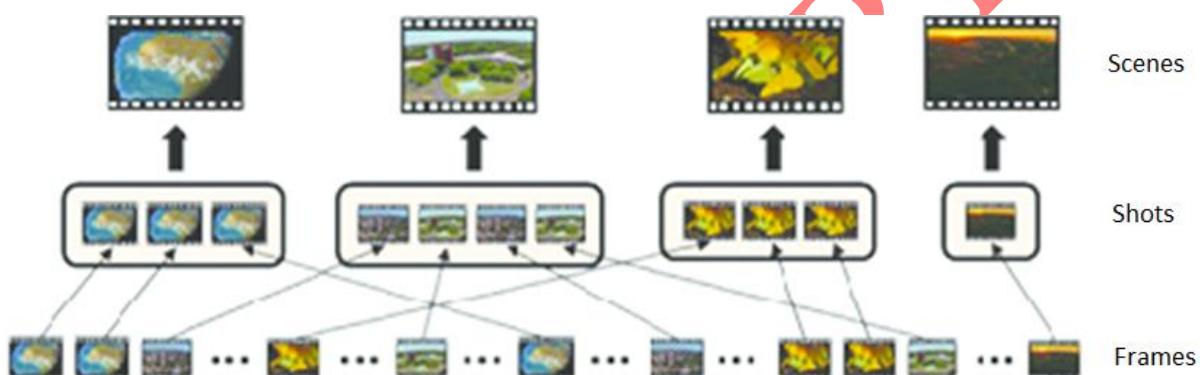
Để có thể trích chọn đặc trưng từ video ta cần hiểu nguyên lý tạo ra video, cụ thể mỗi video là sự kết hợp của kênh hình và kênh tiếng. Trong đó:

- **Kênh hình của video** được phân cấp cấu thành như sau: Mỗi video sẽ được cấu thành bởi sự ghép nối nhiều hoạt cảnh dựa trên nội dung khác nhau (Scenes). Mỗi scene sẽ được kết hợp bởi các shots (Shot là một dãy các khung hình liên tiếp được camera ghi nhận không có sự ngắt quãng nào xảy ra). Shot là một đơn vị cơ bản để xây dựng phân tích nội dung video. Mỗi shot sẽ bao gồm một tập liên tiếp các khung hình (frame), mỗi frame cho phép hiển thị một hình ảnh quan sát được của các sự kiện xảy ra tại một thời điểm. Để đoạn video có thể tạo cảm giác chuyển động, các khung hình phải được quay với tốc độ phù hợp. Vì mắt người chỉ có thể nhận được 24 hình/giây, nên nếu như trong một giây, lần lượt 24 hình hoặc nhiều hơn được phát thì mắt sẽ không nhận ra được sự rời rạc giữa những khung hình, mà chỉ thấy những cảnh liên tục.



Hình 2.16. Cấu trúc của video

Ví dụ minh họa về cấu trúc của 1 video:

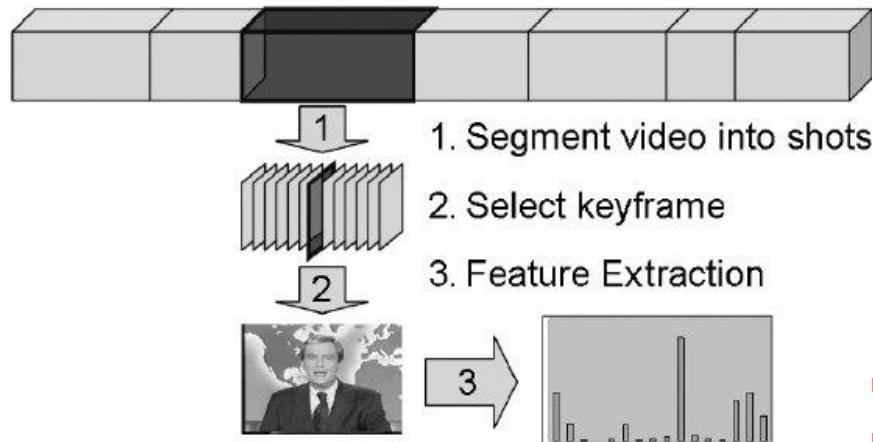


Hình 2.17. Ví dụ minh họa về cấu trúc của 1 video

- **Kênh tiếng** chính là âm thanh phát ra trong khi video đang phát.

Như vậy có một số hướng tiếp cận chính để trích chọn đặc trưng video đó là: 1/ Tiếp cận trích chọn vector đặc trưng của video dựa vào kênh hình; 2/ Tiếp cận trích chọn vector đặc trưng của video dựa vào kênh tiếng; 3/ Kết hợp trích chọn vector đặc trưng của video dựa vào cả kênh hình và kênh tiếng.

Theo hướng tiếp cận thứ nhất thì quá trình trích chọn đặc trưng video sẽ hoạt động theo quy trình dưới đây:



**Hình 2.18. Nguyên tắc hoạt động của trích chọn đặc trưng video**

Theo đó khi khai thác kênh hình để trích chọn đặc trưng cho video sẽ quy về trích chọn đặc trưng hình ảnh (Nội dung này đã được trình bày trong mục 2.4.1.2). Hình ảnh ở đây được lựa chọn trên keyframe đại diện cho các frame trong một shot.

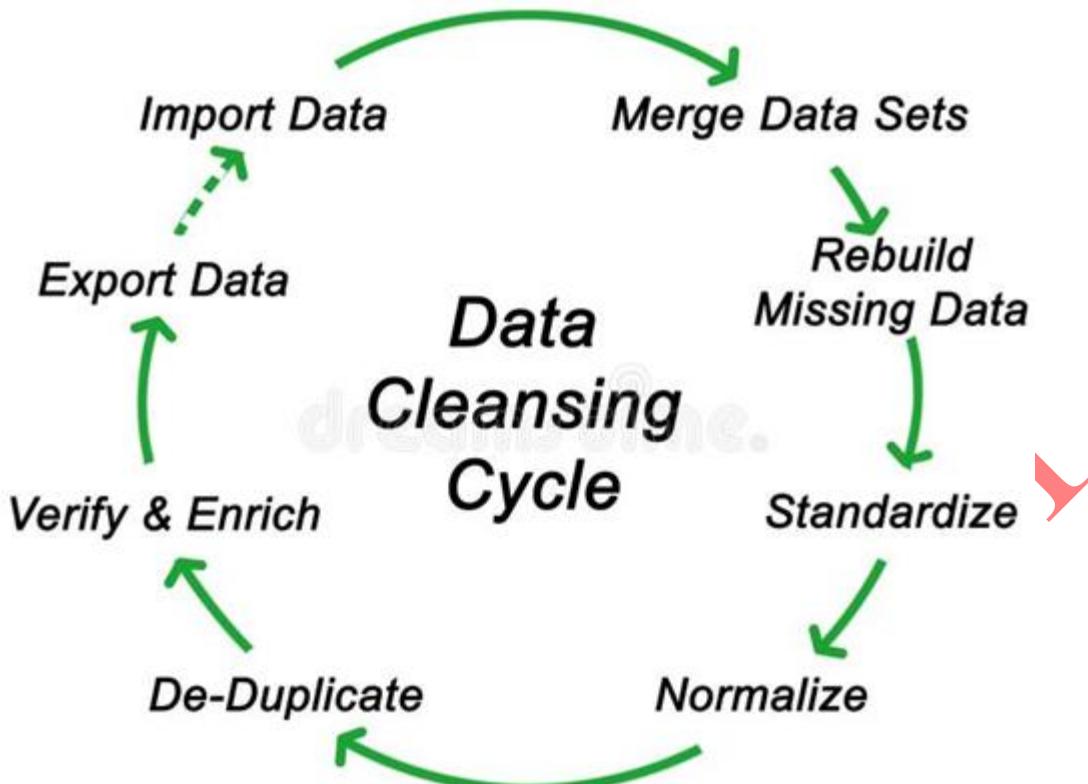
Theo hướng tiếp cận thứ hai thì quá trình trích chọn đặc trưng video sẽ quy về trích chọn đặc trưng âm thanh trong video ((Nội dung này đã được trình bày trong mục 2.4.1.3)).

Theo hướng tiếp cận thứ ba thì cần tìm ra một phương pháp kết hợp giữa hai loại đặc trưng hình ảnh và âm thanh trong việc hình thành vector đặc trưng cho video.

#### 2.4.2. Dọn dẹp dữ liệu

**Khái niệm:** Dọn dẹp dữ liệu (Data cleaning) là quá trình xác định, xóa và / hoặc thay thế thông tin không phù hợp hoặc không chính xác khỏi cơ sở dữ liệu. Kỹ thuật này đảm bảo tiền xử lý dữ liệu có chất lượng tốt phục vụ cho quá trình khai phá dữ liệu mang lại hiệu quả đáng tin cậy. Đây có thể coi là phần nền tảng của khoa học dữ liệu nói chung và khai phá dữ liệu nói riêng.

Việc thực hiện khai phá đối với dữ liệu đa phương tiện sẽ yêu cầu việc dọn dẹp dữ liệu cần thực hiện sau bước trích chọn đặc trưng dữ liệu. Chu trình dọn dẹp dữ liệu khi đó được cụ thể hóa các công việc sau:



Hình 2.19. Chu trình dọn dẹp dữ liệu

Trong đó:

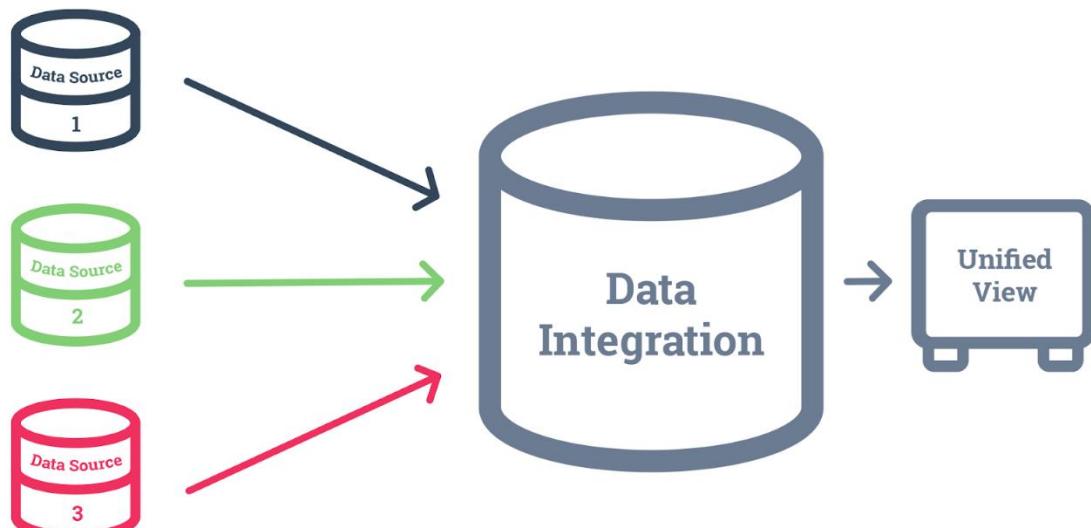
- **Import data**: Nhập dữ liệu đầu vào. Đối với dữ liệu đa phương tiện, đây chính là tập dữ liệu đầu ra của quá trình trích chọn đặc trưng dữ liệu trước đó
- **Merge data sets**: Hợp nhất các tập dữ liệu có liên quan. Quá trình này còn được gọi là tích hợp dữ liệu (Data integration), nội dung này sẽ được cụ thể hơn ở phần trình bày phía dưới.
- **Rebuild missing data**: Xử lý dữ liệu bị thiếu (Xóa, thay thế dữ liệu thiếu với dữ liệu khác phù hợp). Đây là một phần công việc trong quá trình làm sạch dữ liệu (Data cleaning), nội dung này sẽ được cụ thể hơn ở phần trình bày phía dưới.
- **Standardize** : Tiêu chuẩn hóa dữ liệu nhằm chuyển đổi thông tin được lưu trữ trong dữ liệu gốc thành một dạng nhất quán và được xác định rõ ràng.
- **Normalize**: Chuẩn hóa dữ liệu gốc về miền giá trị phù hợp cho tính toán. Standardize và Normalize là hai công việc thường đi cùng nhau trong quá trình chuyển đổi dữ liệu (Data transformation), nội dung này sẽ được cụ thể hơn ở phần trình bày phía dưới.
- **De-duplicate**: Xử lý những dữ liệu trùng lặp. Đây cũng là một phần công việc trong quá trình làm sạch dữ liệu

- **Verify & Enrich:** Xác thực độ chính xác của dữ liệu và làm giàu dữ liệu nhằm cung cấp thông tin đầy đủ và hữu ích. Đây cũng là một phần công việc trong quá trình làm sạch dữ liệu
- **Export data:** Xuất dữ liệu sau khi làm sạch ra để phục vụ cho giai đoạn khai phá dữ liệu sau đó.

Nội dung dưới đây sẽ trình bày cụ thể về một số công việc trong quy trình dọn dẹp dữ liệu, đó là : Tích hợp dữ liệu, chuyển đổi dữ liệu, thu giảm dữ liệu và làm sạch dữ liệu.

#### 2.4.2.1. Tích hợp dữ liệu ( Data integration )

~~Khái niệm: Tích hợp dữ liệu là công việc trộn dữ liệu (merge data) từ nhiều nguồn khác nhau vào một kho dữ liệu. Các nguồn khác nhau này có thể là từ cơ sở dữ liệu, khối dữ liệu hoặc tập tin dữ liệu.~~



Hình 2.20. Tích hợp dữ liệu

~~Ví dụ tích hợp dữ liệu về lương nhân viên trong công ty qua 3 năm 2008, 2009, 2010 để tạo thành một nguồn dữ liệu hợp nhất qua các năm.~~

Year 2010	
Year 2009	
Year 2008	
Quarter	Sales
Q1	\$224,000
Q2	\$408,000
Q3	\$350,000
Q4	\$586,000

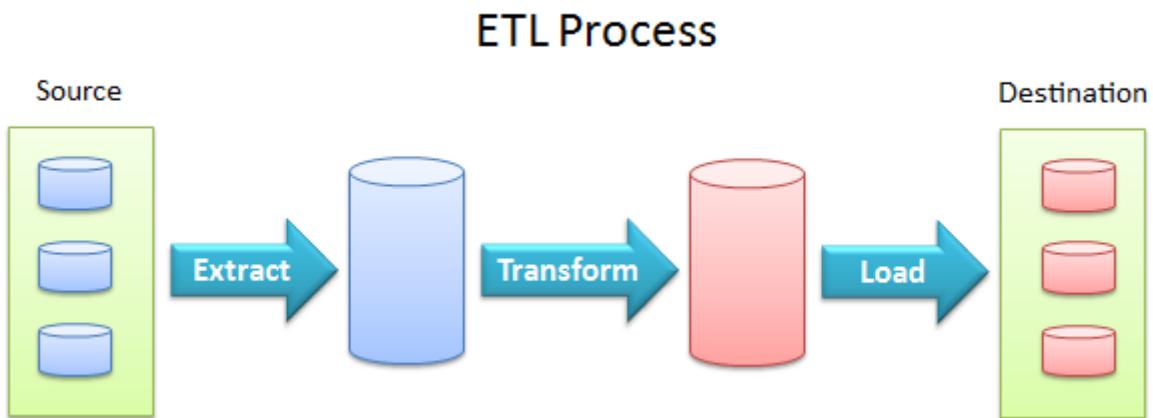
→

Year	Sales
2008	\$1,568,000
2009	\$2,356,000
2010	\$3,594,000

Hình 2.21. Minh họa về tích hợp dữ liệu về lương nhân viên công ty qua các năm

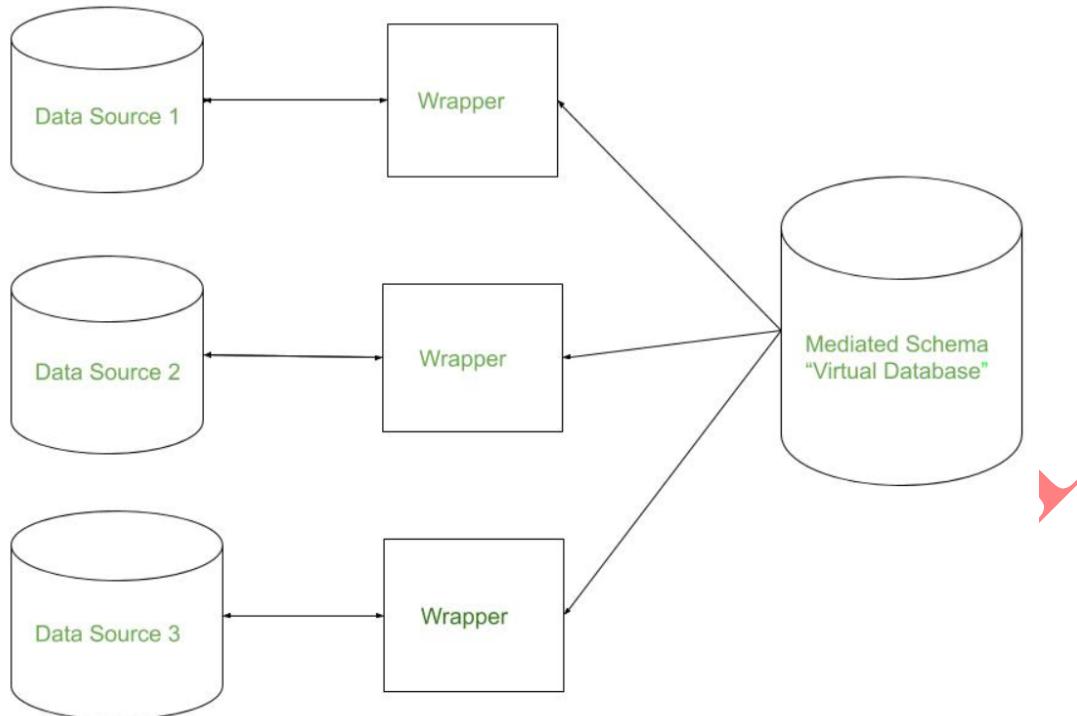
Có hai hướng tiếp cận chính để thực hiện tích hợp dữ liệu là cách tiếp cận liên kết chặt chẽ (Tight coupling) và cách tiếp cận liên kết lỏng lẻo (Loose Coupling).

Theo hướng tiếp cận liên kết chặt chẽ để tích hợp dữ liệu: Dữ liệu được kết hợp với nhau từ nhiều nguồn sẽ được tích hợp trong một kho dữ liệu duy nhất được đặt tại một vị trí vật lý xác định. Quá trình tích hợp này căn cứ theo tiến trình trích xuất, chuyển đổi và tải dữ liệu ETL (Extraction, Transformation and Loading).



Hình 2.22. Tiếp cận liên kết chặt chẽ để tích hợp dữ liệu (Tight coupling)

Ngược lại với hướng tiếp cận liên kết chặt chẽ là hướng tiếp cận liên kết lỏng lẻo. Theo hướng tiếp cận liên kết lỏng lẻo này thì chúng ta sẽ duy trì một giao tiếp để nhận yêu cầu dữ liệu từ người dùng và chuyển tiếp truy vấn tới cơ sở dữ liệu đích. Như vậy cơ sở dữ liệu đích mới là nơi lưu trữ dữ liệu gốc để đáp ứng yêu cầu dữ liệu.



Hình 2.23. Tiếp cận liên kết lỏng lẻo để tích hợp dữ liệu (Loose Coupling).

#### 2.4.2.2. Chuyển đổi dữ liệu (Data transformation)

**Khái niệm:** Chuyển đổi dữ liệu là quá trình biến đổi hay kết hợp dữ liệu vào những dạng thích hợp cho quá trình khai phá dữ liệu. Đây cũng là một giai đoạn trong quy trình ETL (được đề cập ở trên).

Về cơ bản quá trình này sẽ gồm hai công việc chính là tiêu chuẩn hóa dữ liệu (Standardize) và chuẩn hóa dữ liệu (Normalization). Về ý nghĩa của hai công việc này đã được miêu tả trong chương trình đơn dẹp dữ liệu. Để phân biệt hai công việc này chúng ta cùng xét ví dụ sau khi biểu diễn tập dữ liệu về tuổi và lương của nhân viên trong một công ty như dưới đây.

Standardisation		Max-Min Normalization			
	Age	Age	Salary		
0	0.758874	7.494733e-01	0	0.739130	0.685714
1	-1.711504	-1.438178e+00	1	0.000000	0.000000
2	-1.275555	-8.912655e-01	2	0.130435	0.171429
3	-0.113024	-2.532004e-01	3	0.478261	0.371429
4	0.177609	6.632192e-16	4	0.565217	0.450794
5	-0.548973	-5.266569e-01	5	0.347826	0.285714
6	0.000000	-1.073570e+00	6	0.512077	0.114286
7	1.340140	1.387538e+00	7	0.913043	0.885714
8	1.630773	1.752147e+00	8	1.000000	1.000000
9	-0.258340	2.937125e-01	9	0.434783	0.542857

Hình 2.24. Minh họa quá trình chuyển đổi dữ liệu tuổi và lương trong khai phá dữ liệu nhân viên

Theo ví dụ minh họa trên cho ta hiểu rõ hơn về quá trình Standardize sẽ giúp chuyển đổi thông tin được lưu trữ trong dữ liệu gốc thành một dạng nhất quán dưới dạng số cho cả thuộc tính age và salary của nhân viên, nhằm đáp ứng tính toán các số liệu được thuận lợi trên máy tính. Tiếp sau đó, quá trình Normalize sẽ chuẩn hóa dữ liệu gốc về miền giá trị phù hợp cho tính toán nằm trong [0,1] để tăng hiệu quả tính toán cho cả hai thuộc tính này.

Với mỗi quá trình Standardize và Normalize đều có nhiều phương pháp khác nhau cho thực hiện. Cụ thể một số cách chuẩn hóa Normalize: min-max normalize, z-score normalize, normalize by decimal scaling...

#### 2.4.2.3. Thu giảm dữ liệu (Data reduction)

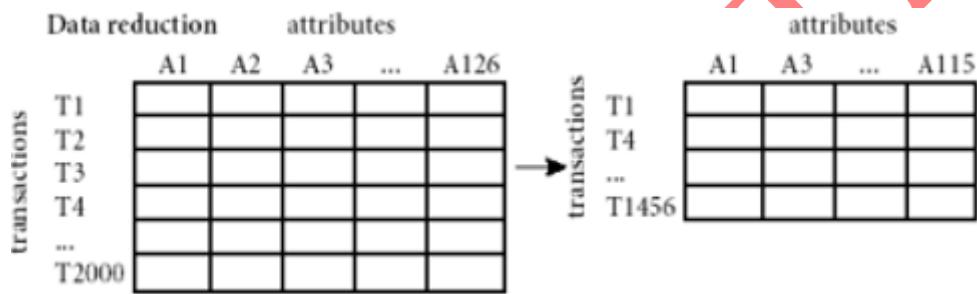
**Khái niệm:** Thu giảm dữ liệu là công việc giảm bớt các thuộc tính dữ liệu (Dimensionality reduction) hoặc kích thước số lượng bộ dữ liệu (Numerosity reduction) từ biểu diễn tập dữ liệu. Tập dữ liệu được thu giảm vẫn phải đảm bảo tính toàn vẹn nhưng nhỏ hơn nhiều so với tập dữ liệu ban đầu, đồng thời không ảnh hưởng tới kết quả khai phá dữ liệu sau này.

Theo đó, các kỹ thuật thu giảm dữ liệu được phân chia theo 2 hướng tiếp cận trên, với mỗi hướng đều có những phương pháp khác nhau để thực hiện công việc này.



Hình 2.25. Các hướng tiếp cận giảm chiều dữ liệu

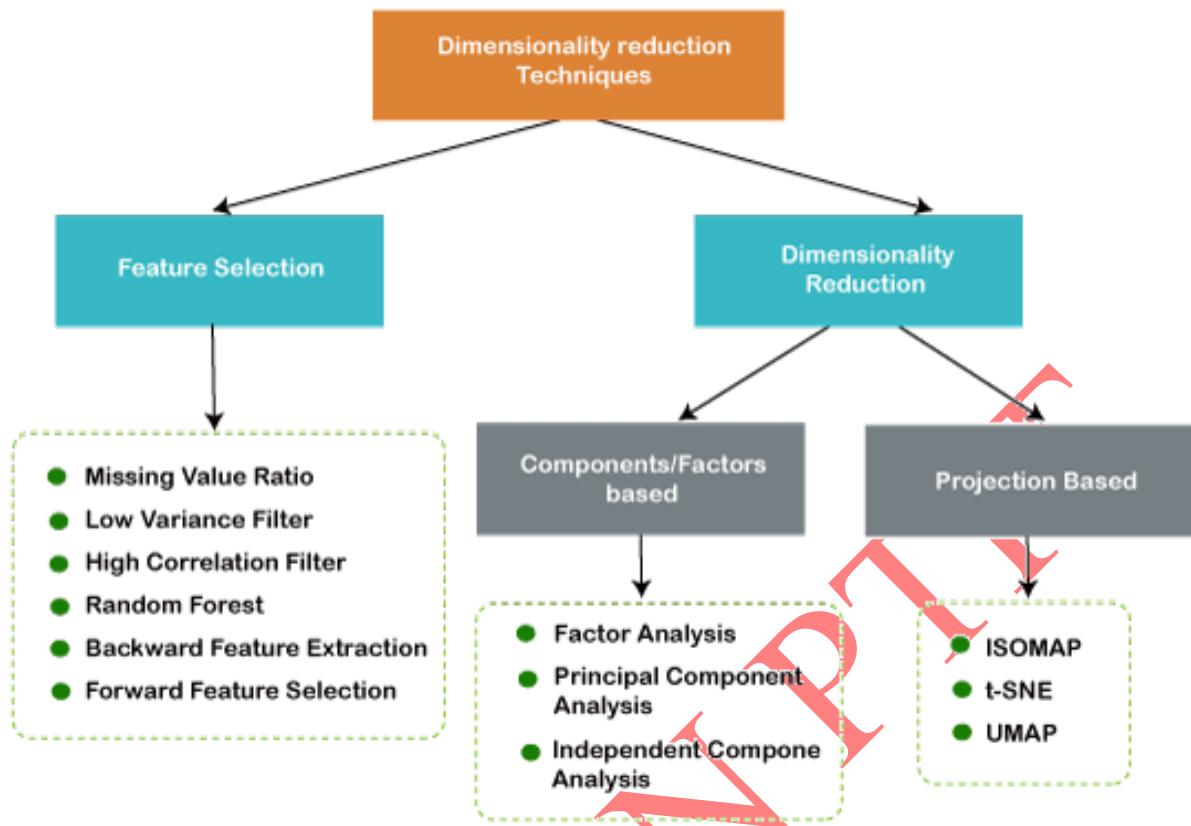
Với hướng tiếp cận thu giảm chiều dữ liệu thì các thuộc tính dư thừa sẽ bị loại bỏ nên việc lựa chọn thuộc tính giữ lại trong tập dữ liệu cho khai phá sẽ chọn những thuộc tính có độ lợi thông tin lớn.



Hình 2.26. Nguyên tắc hoạt động của hướng tiếp cận giảm bớt chiều dữ liệu

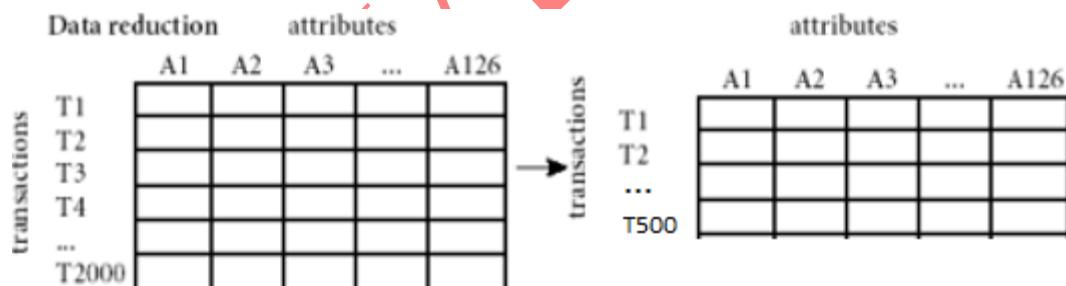
Đã có khá nhiều kỹ thuật được đưa ra cho mục đích này.

THƯỚC TÍNH



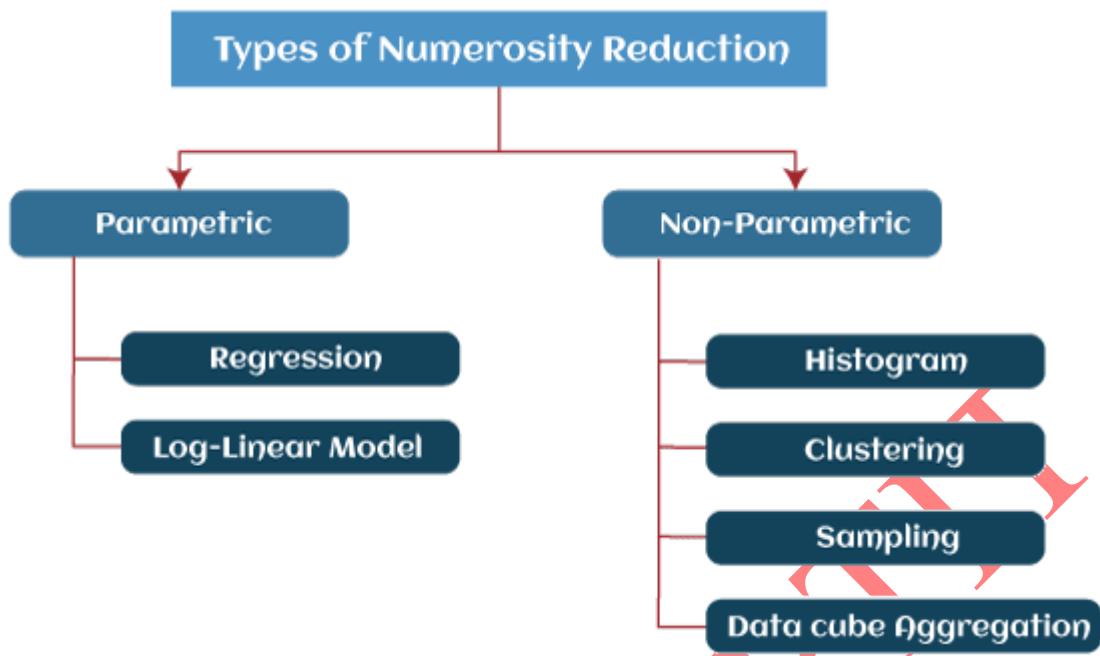
Hình 2.27. Các kỹ thuật giảm bớt chiều dữ liệu

Với hướng tiếp cận thu giảm kích thước số lượng bộ dữ liệu thì các bộ dữ liệu dư thừa/ trùng lặp sẽ bị loại bỏ.



Hình 2.28. Nguyên tắc hoạt động của hướng tiếp cận giảm bớt số lượng bộ dữ liệu

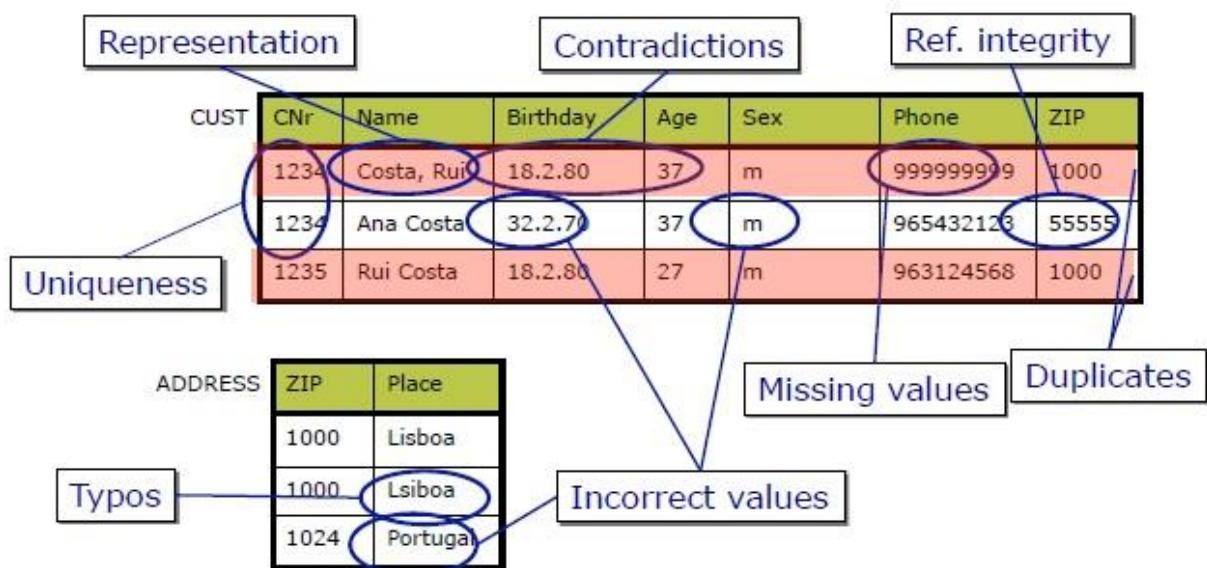
Đã có khá nhiều kỹ thuật được đưa ra cho mục đích thu giảm kích thước số lượng bộ dữ liệu được thể hiện trong hình dưới đây.



**Hình 2.29.** Các kỹ thuật giảm bớt số lượng bộ dữ liệu dữ liệu

#### 2.4.2.4. *Làm sạch dữ liệu (Data cleaning)*

**Khái niệm:** Làm sạch dữ liệu và quá trình gán các giá trị thuộc tính còn thiếu, sửa chữa các dữ liệu nhiễu/ lỗi, xác định hoặc loại bỏ các ngoại lai (outliers), giải quyết các mâu thuẫn dữ liệu.



Typo

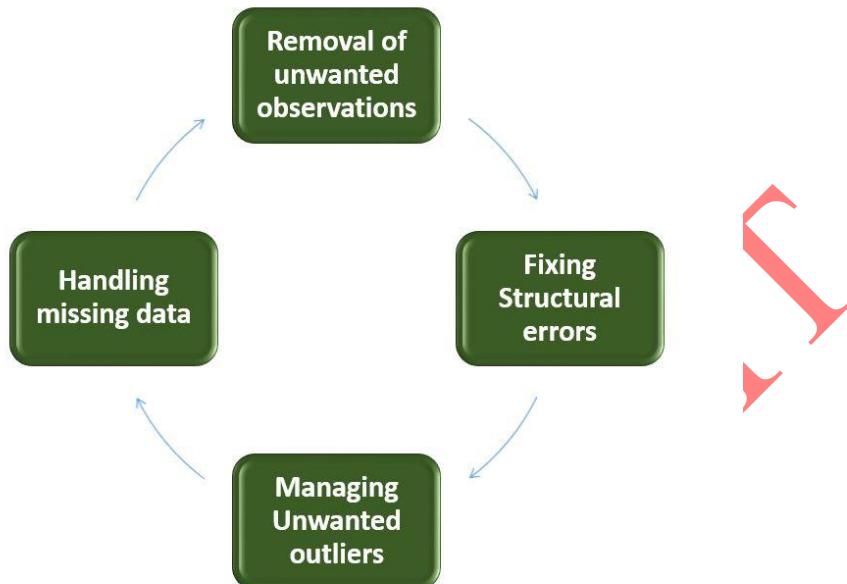
Missing values

Duplicates

Incorrect values

**Hình 2.30.** Minh họa dữ liệu cần được làm sạch

Đứng trước tập dữ liệu chưa được làm sạch, có một số kỹ thuật phổ biến được đưa ra như: Điều chỉnh giá trị còn thiếu cho dữ liệu, chỉnh sửa những giá trị dữ liệu sai lệch, loại bỏ những bộ dữ liệu dư thừa, loại bỏ các ngoại lai để tránh nhiễu cho tập dữ liệu...



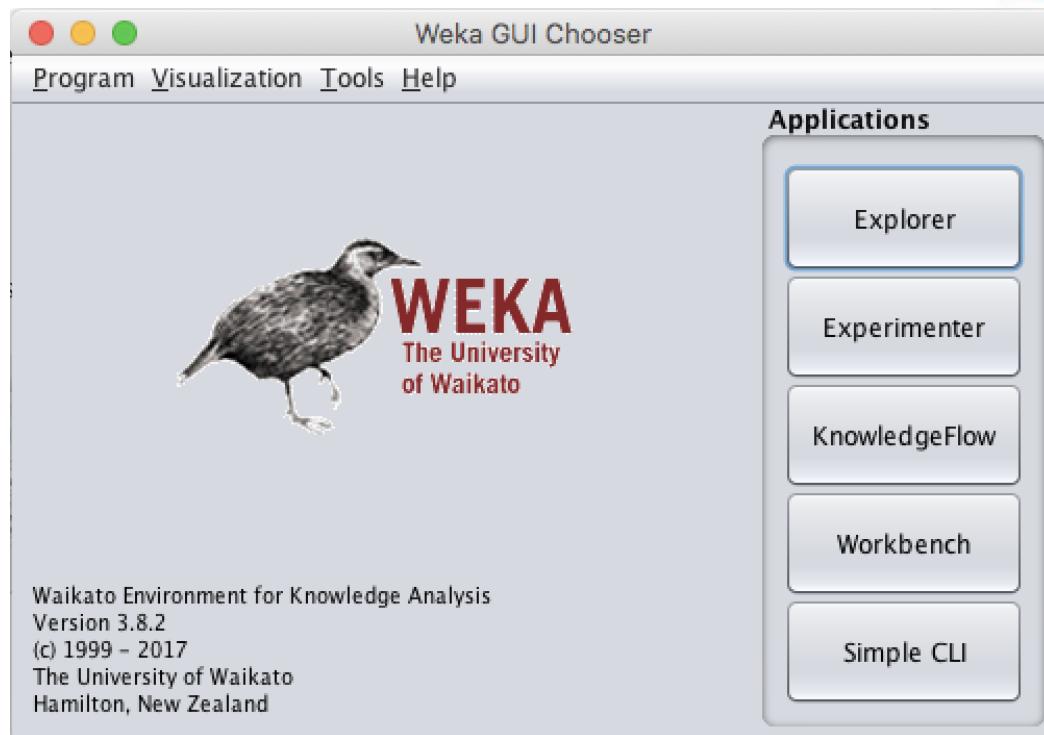
Hình 2.31. Các bước phổ biến thực hiện để làm sạch dữ liệu

## 2.5. Tiền xử lý dữ liệu đa phương tiện với Weka

### 2.5.1. Giới thiệu Weka

#### 2.5.1.1. Môi trường Weka

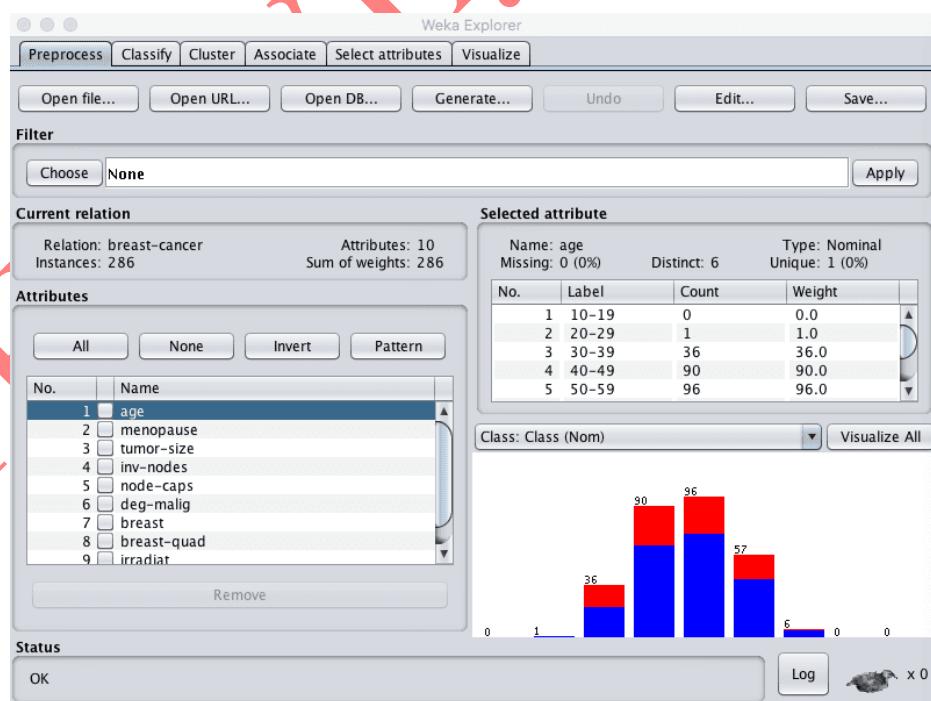
Weka là một công cụ phần mềm viết bằng Java, phục vụ lĩnh vực học máy và khai phá dữ liệu. Các tính năng chính có thể kể đến đó là: 1/ Cung cấp một tập hợp các công cụ tiền xử lý dữ liệu, các giải thuật học máy, khai phá dữ liệu và các phương pháp thí nghiệm đánh giá; 2/Cung cấp giao diện đồ họa (gồm cả tính năng hiển thị hóa dữ liệu); 3/ Cung cấp môi trường cho phép so sánh các giải thuật học máy và khai phá dữ liệu. Giao diện đồ họa cụ thể của Weka được minh họa dưới đây:



Hình 2.32. Giao diện phần mềm Weka

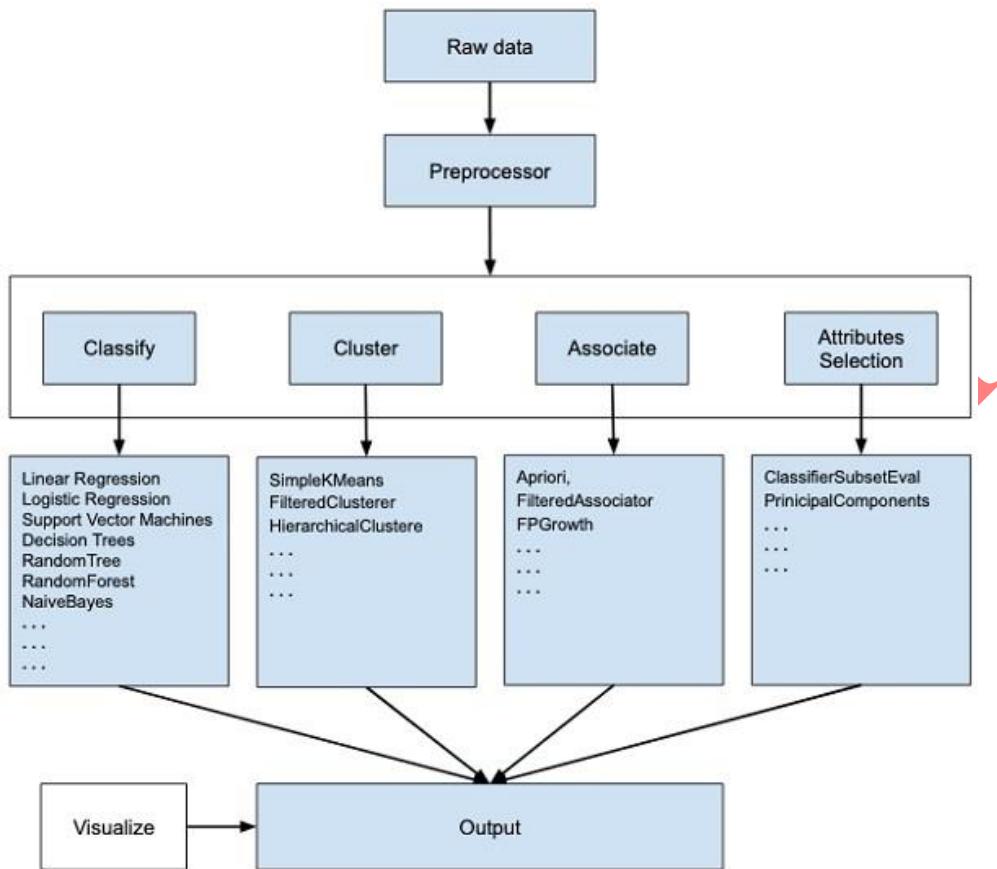
Trong đó:

- Explorer: Chức năng cho phép tiền xử lý dữ liệu, phân lớp, phân cụm, khai phá luận kết hợp, lựa chọn thuộc tính, trực quan hóa dữ liệu.



Hình 2.33. Giao diện chức năng Weka Explorer

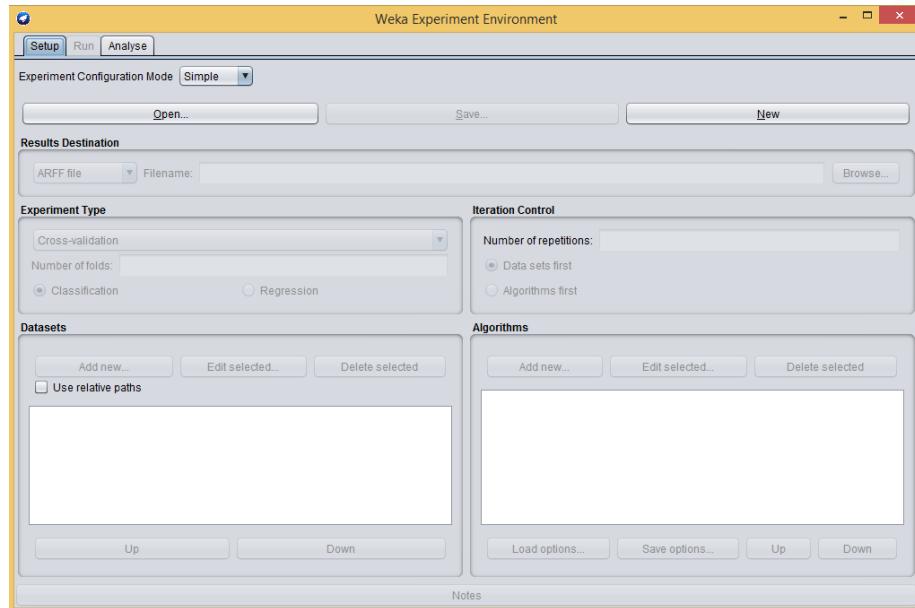
Sơ đồ dưới đây sẽ cho cái nhìn tổng quan về luồng xử lý dữ liệu trong Weka Explorer :



Hình 2.34. Luồng xử lý dữ liệu trong Weka Explorer

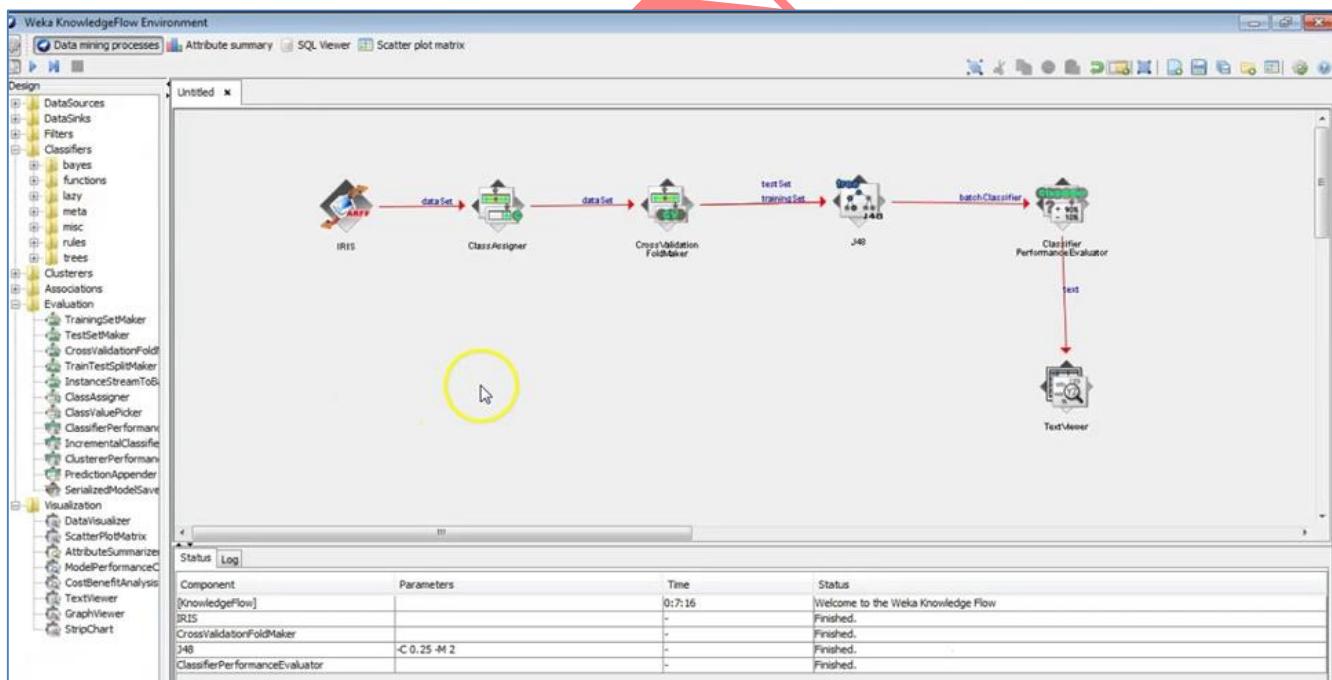
Như vậy quá trình xử lý dữ liệu trên trong Weka Explorer sẽ gồm có 3 giai đoạn: 1/ Tiên xử lý dữ liệu thô (Weka cung cấp một tập hợp các bộ lọc filter để trích chọn đặc trưng và dọn dẹp dữ liệu), dữ liệu đã được tiền xử lý được lưu đệm trong bộ nhớ để áp dụng các thuật toán học máy sau đó ; 2/ Lựa chọn mô hình học máy phù hợp cho mục tiêu bài toán (Phân loại, phân cụm, luật kết hợp). Trong giai đoạn này, chúng ta cũng có thể thu ~~giảm~~ dữ liệu bằng cách lựa chọn thuộc tính phù hợp. Lưu ý với mỗi loại mô hình, chúng ta có thể lựa chọn các thuật toán khác nhau để thực thi, với mỗi thuật toán sẽ cần thiết lập giá trị cho các tham số đầu vào; 3/ Thông kê kết quả xử lý của mô hình học máy và trực quan hóa dữ liệu được phân tích.

- Experimenter: Chức năng cho phép thiết kế các thí nghiệm, lựa chọn các thuật toán và tập dữ liệu, chạy thí nghiệm, phân tích kết quả (Cho phép so sánh các kết quả...)



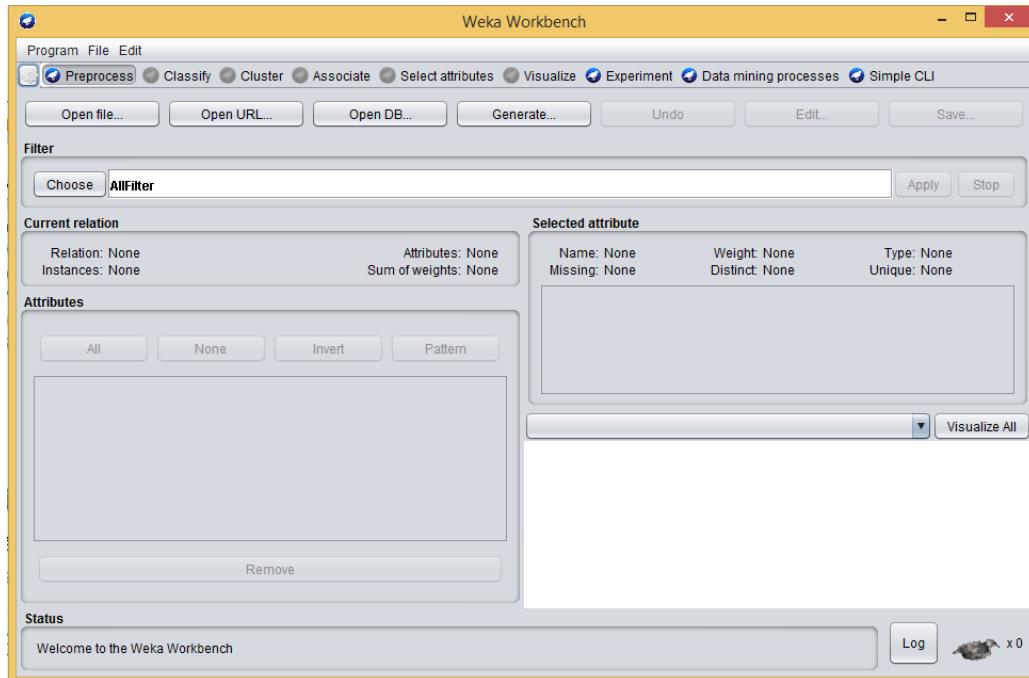
Hình 2.35. Giao diện chức năng Weka Experimenter

- Knowledge Flow: Chức năng cho phép thiết kế quá trình khai phá dữ liệu trực quan bằng hình ảnh



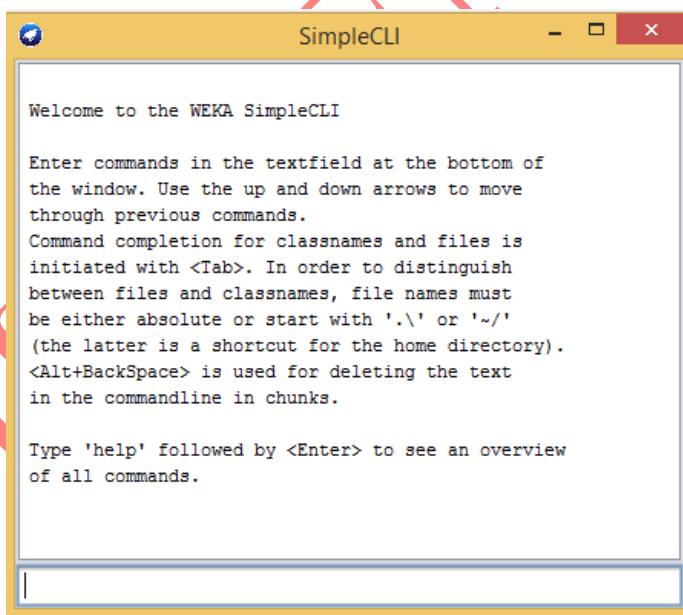
Hình 2.36. Giao diện chức năng Weka Knowledge Flow

- Workbench: Cung cấp cho người dùng công cụ mạnh mẽ để khai phá dữ liệu bằng việc tổng hợp các chức năng trên vào trong một ứng dụng.



Hình 2.37. Giao diện chức năng Workbench

- Simple CLI: Chức năng cho phép người dùng tương tác với Weka bằng cách gõ lệnh.

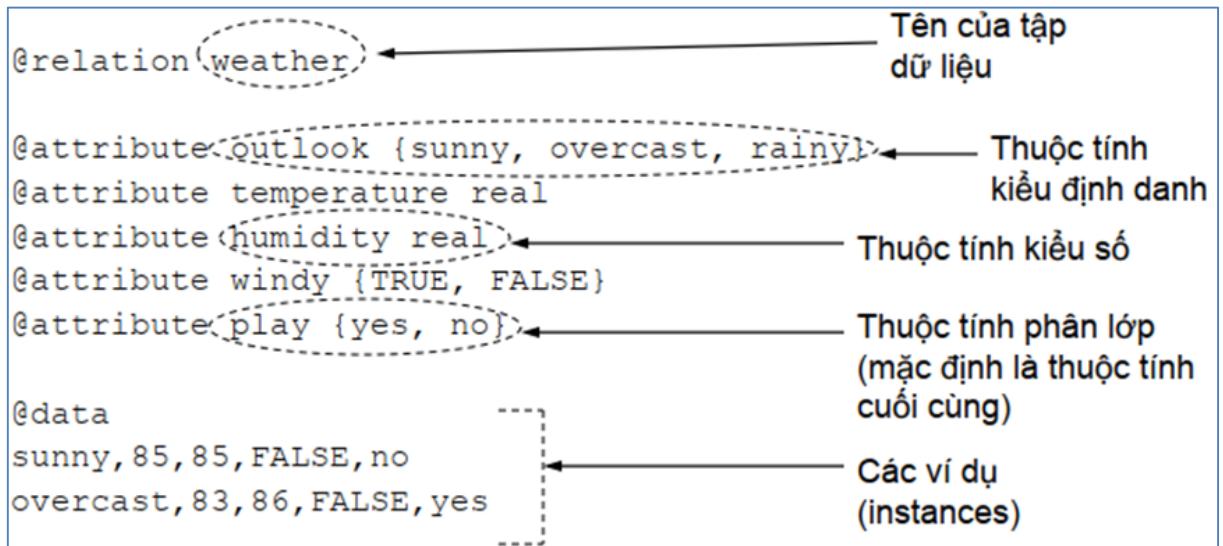


Hình 2.38. Giao diện chức năng Simple CLI

Việc cài đặt Weka và môi trường lập trình Java được thực hiện theo 3 bước sau: 1/ Cài đặt Java JDK; 2/Cài đặt weka bằng việc tải phần mềm từ địa chỉ: <https://www.cs.waikato.ac.nz/ml/weka/> ; Cài đặt Netbeans framework để lập trình với Java.

#### 2.5.1.2. Khuôn dạng của tập dữ liệu trong Weka

Khuôn dạng tệp tin văn bản chuẩn để làm việc trong môi trường Weka là ARFF (Attribute-Relation File Format)



Hình 2.39. Ví dụ và giải thích về tệp tin ARFF làm việc trong môi trường Weka

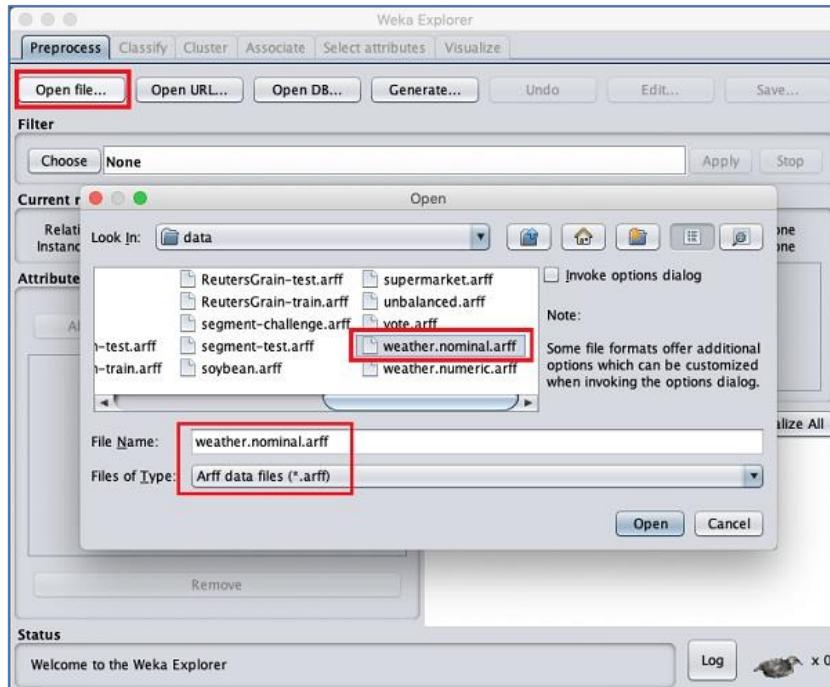
Trong đó:

- @relation : Thẻ định nghĩa tên của tập dữ liệu
- @attribute: thẻ định nghĩa thuộc tính. Với mỗi thuộc tính được định nghĩa đồng thời xác định luôn kiểu dữ liệu cho nó. Trường hợp thuộc tính có kiểu định danh thì cần xác định luôn các giá trị nhận vào cho thuộc tính này.
- @data: thẻ được đặt trước danh sách các ví dụ / đối tượng dữ liệu. Mỗi ví dụ sẽ gồm 1 bộ giá trị nhận được tương ứng với các thuộc tính đã được khai báo.

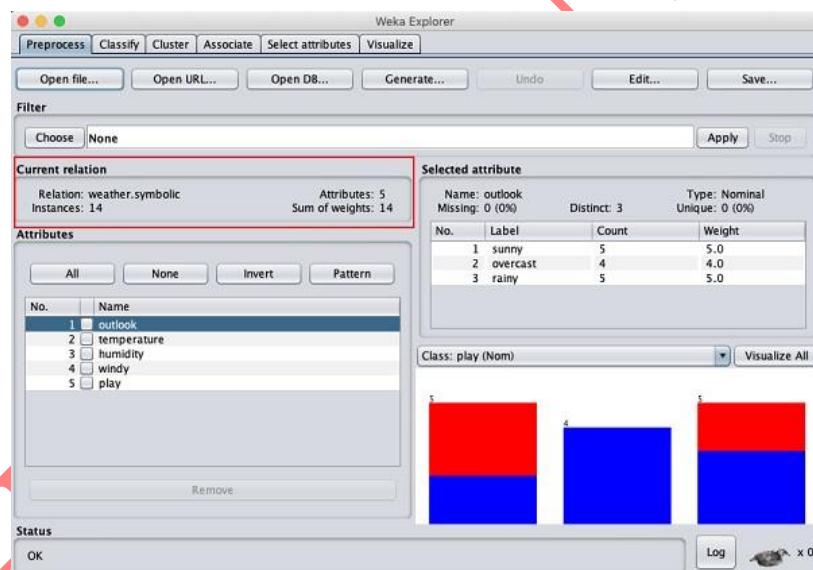
Ngoài ra, Weka còn hỗ trợ mở tập dữ liệu ở nhiều định dạng khác nhau: arff.gz, bsi, csv, dat, data, json, json.gz, libsvm, m, names, xrf, xrf.gz. Khi làm việc với file ở các định dạng này, chúng ta cần chuyển về định dạng file chuẩn .arff để có thể dùng sau đó trong môi trường Weka.

### 2.5.2. Mô tả dữ liệu trong Weka

Để hiểu về tập dữ liệu trong Weka, bài giảng tiếp cận tập dữ liệu về thời tiết (weather-nominal.arff) sau:



Khi file được mở ra sẽ cho chúng ta theo dõi các thông tin về tập dữ liệu này:



Tại giao diện này, các thông tin được cung cấp gồm có:

- Phân vùng “**Current relation**”: 1/ Tên tập dữ liệu (Relation); 2/ Số lượng đối tượng dữ liệu (Instances); 3/ Số lượng thuộc tính cho tập dữ liệu (Attributes).
- Phân vùng “**Attributes**”: Liệt kê danh sách các thuộc tính của tập dữ liệu. Tại đây chúng ta có thể thực hiện một số thao tác như thêm chọn hay xóa bỏ thuộc tính không cần thiết.
- Phân vùng “**Selected attribute**”: Cho phép hiển thị nội dung thông tin chi tiết cho mỗi thuộc tính được chọn trong phân vùng Attributes, đó là: 1/ Tên thuộc tính (Name); 2/ Kiểu dữ liệu của thuộc tính (Type); 3/ Tỷ lệ các đối tượng thiếu

dữ liệu (Missing value); 4/ Số lượng các giá trị phân biệt nhận được cho thuộc tính (Distinct); 5/ Thống kê các giá trị nhận được cho thuộc tính theo bảng bên dưới.

- Phân vùng Visualize cho phép trực quan hóa dữ liệu cùng tỷ lệ phân bổ dữ liệu theo 2 tùy chọn: 1/ Cho từng thuộc tính ; 2/ Tất cả thuộc tính.

### 2.5.3. Tiền xử lý dữ liệu với Weka

#### 2.5.3.1. Trích chọn đặc trưng dữ liệu với Weka

##### (1) Trích chọn đặc trưng văn bản

- Giả sử ta cần tiến hành trích chọn đặc trưng văn bản từ dataset là tập dữ liệu văn bản *sentences.arff*.

```
@relation sentences

@attribute Document string
@attribute class {yes,no}

@data

"The price of crude oil has increased significantly", yes
"Demand for crude oil outstrips supply", yes
"Some people do not like the flavor of olive oil", no
"The food was very oily", no
"Crude oil is in short supply", yes
"Use a bit of cooking oil in the frying pan", no
```

- Khởi động WEKA Explorer → Chọn dataset sẽ mở ra thông tin tập dữ liệu trước khi sử dụng bộ lọc để trích chọn đặc trưng:

**Weka Explorer**

Preprocess Classify Cluster Associate Select attributes Visualize

Open file... Open URL... Open DB... Generate... Undo Edit... Save...

Filter Choose None Apply Stop

Current relation Relation: sentences Attributes: 2 Instances: 6 Sum of weights: 6

Attributes All None Invert Pattern

No.	Name
1	Document
2	class

Remove

Selected attribute Name: class Type: Nominal  
Missing: 0 (0%) Distinct: 2 Unique: 0 (0%)

No.	Label	Count	Weight
1	yes	3	3
2	no	3	3

Class: class (Nom) Visualize All

3 3

Status OK Log x 0

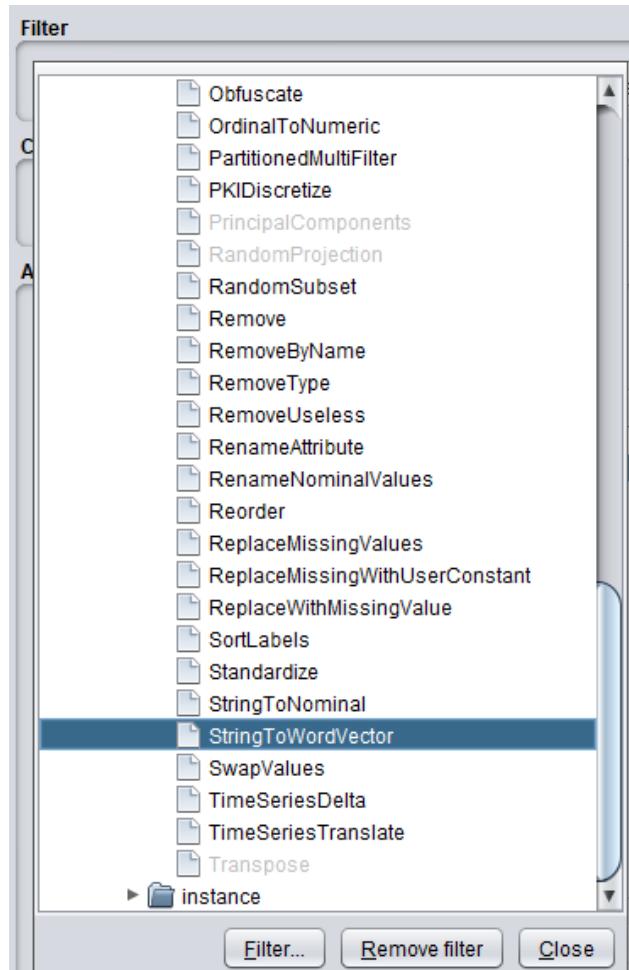
- Chọn **Edit** để xem tập dữ liệu trước khi trích xuất đặc trưng

**Viewer**

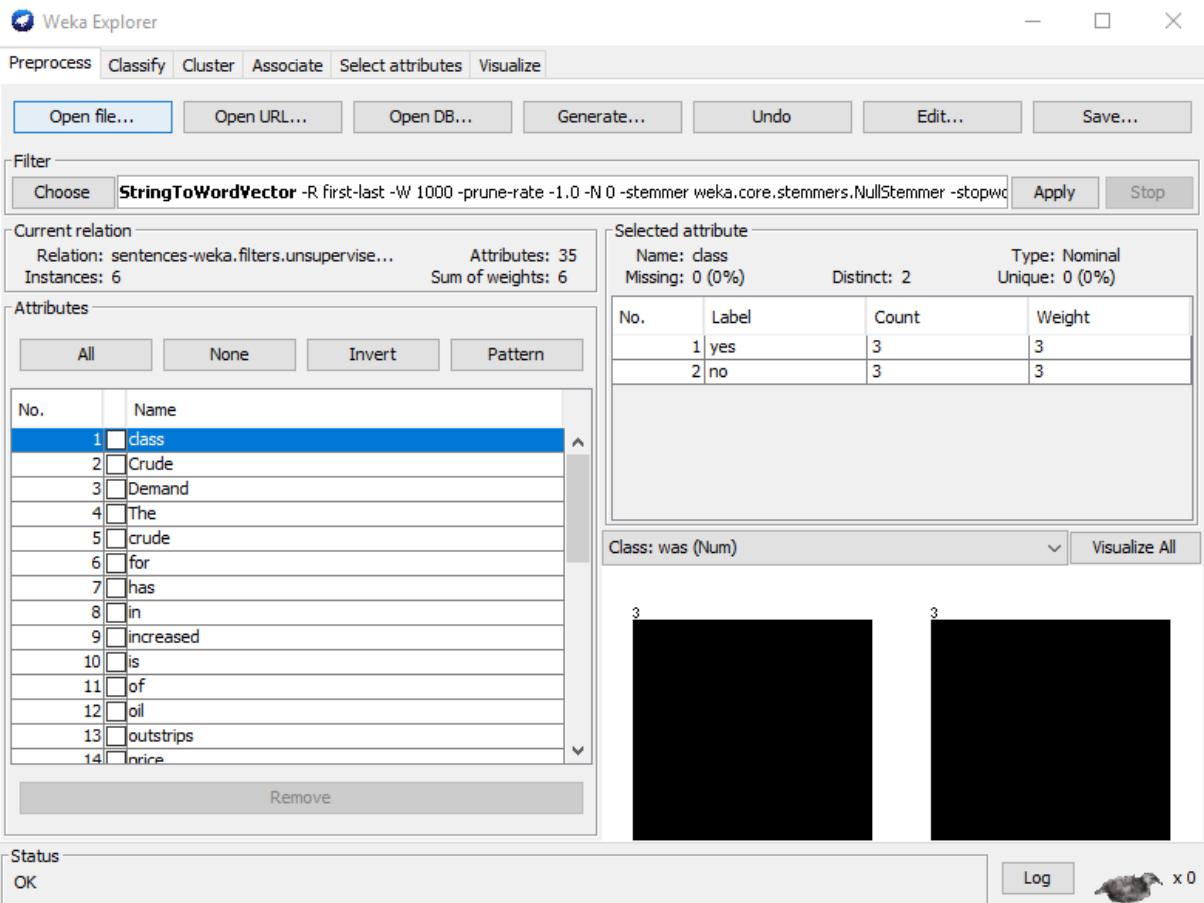
Relation: sentences

No.	1: Document String	2: class Nominal
1	The price of crude oil has increased significantly	yes
2	Demand for crude oil outstrips supply	yes
3	Some people do not like the flavor of olive oil	no
4	The food was very oily	no
5	Crude oil is in short supply	yes
6	Use a bit of cooking oil in the frying pan	no

- Minh họa thực hiện trích chọn văn bản theo phương pháp Bag of Words bằng cách chọn filter áp dụng : Filter → unsupervised → attribute → **StringToWordVector**



- Chọn **Apply** để sử dụng bộ lọc. Kết quả sẽ cho ra một tập các đặc trưng trích xuất được từ văn bản.



- Trên giao diện này chọn **Edit** để xem toàn bộ tập dữ liệu văn bản được biểu diễn dưới dạng bản ghi. Với mỗi bản ghi tương ứng với 1 vector đặc trưng của 1 văn bản.

Viewer

Relation: sentences-weka.filters.unsupervised.attribute.StringToWordVector -R first-last -W 1000 -prune-rate -1.0 -N 0 -stemmer weka.core.stemmers.NullStemmer -stopwords weka.core.stopwords.NullM1-t...

No.	1: class	2: Crude	3: Demand	4: The	5: crude	6: for	7: has	8: in	9: increased	10: is	11: of	12: oil	13: outstrips	14: price	15: short	16: significantly	17: supply
	Nominal	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric
1	yes	0.0	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0	1.0	1.0	0.0	1.0	0.0	1.0	0.0
2	yes	0.0	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	1.0
3	no	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0
4	no	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	yes	1.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	1.0	0.0	0.0	1.0	0.0	1.0
6	no	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0

## (2) Trích chọn đặc trưng hình ảnh

- Minh họa việc trích chọn đặc trưng văn bản từ dataset là tập dữ liệu về hoa sau:



- Tạo file *flower.arff* cho dataset trên để có thể sử dụng trong Weka như sau:

flower - Notepad

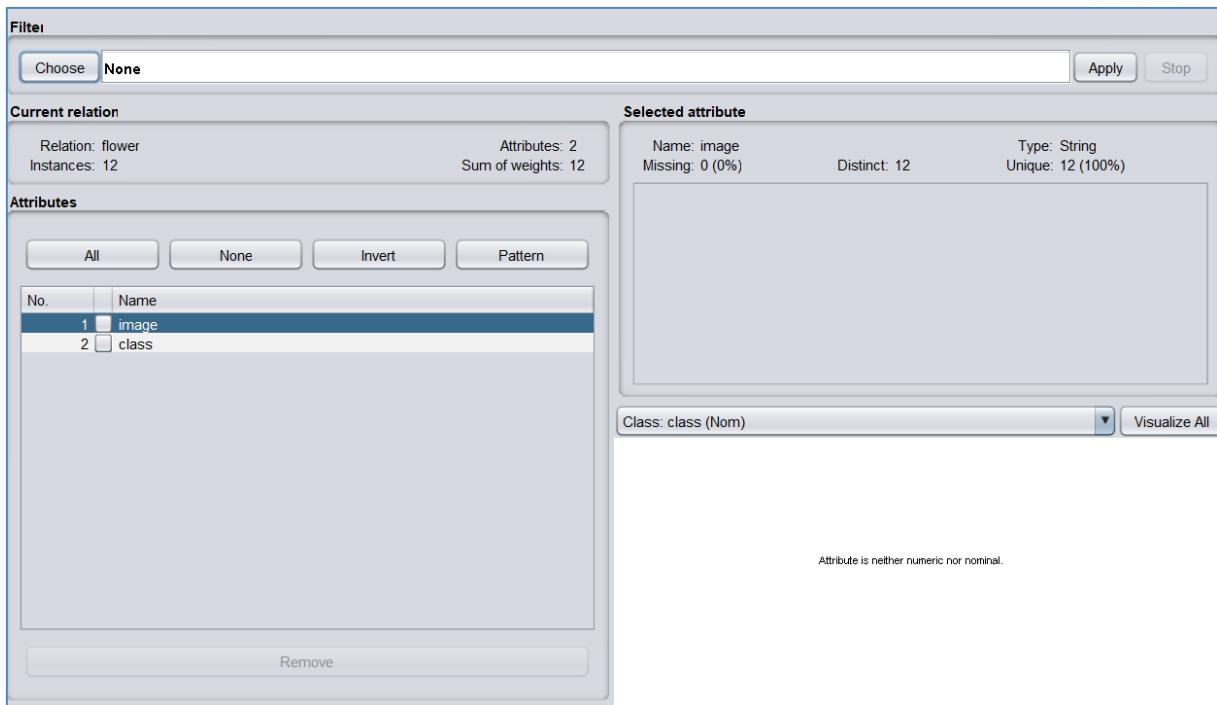
File Edit Format View Help

```
@relation flower

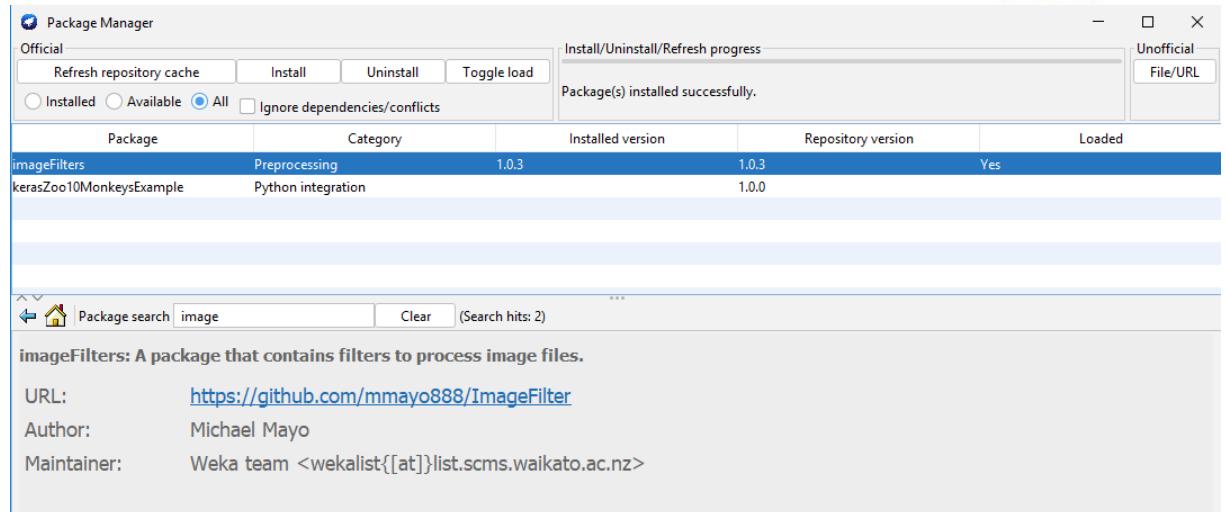
@attribute image string
@attribute class {Dao, Mai, Ban}

@data
hoadao_1.jpg, Dao
hoadao_2.jpg, Dao
hoadao_3.jpg, Dao
hoadao_4.jpg, Dao
hoadao_5.jpg, Dao
hoamai_1.jpg, Mai
hoamai_2.jpg, Mai
hoamai_3.jpg, Mai
hoamai_4.jpg, Mai
hoaaban_1.jpg, Ban
hoaaban_2.jpg, Ban
hoaaban_3.jpg, Ban
```

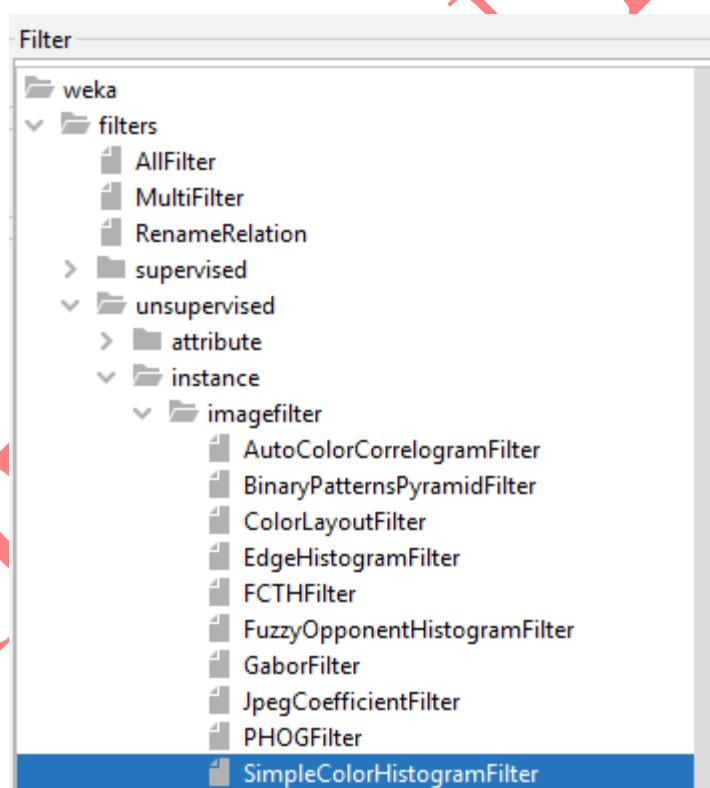
- Khởi động WEKA Explorer → Chọn dataset *flower.arff*



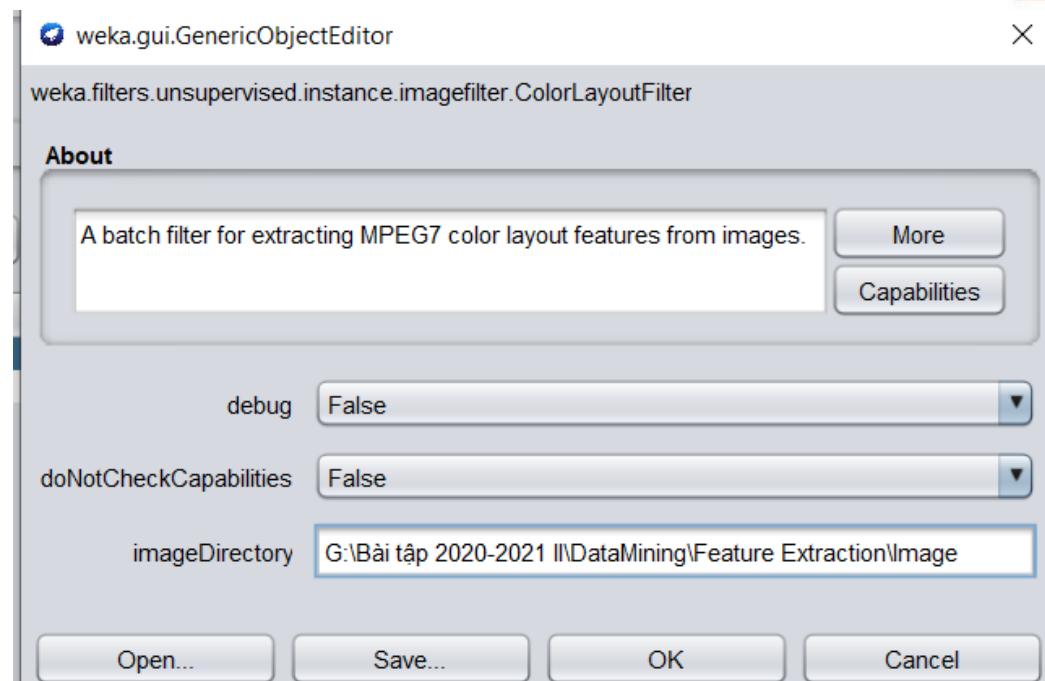
- Chọn bộ lọc để tiến hành trích chọn đặc trưng cho tập dữ liệu về hoa. Theo cơ sở lý thuyết đã đề cập trong mục 2.4.1, có ba loại đặc trưng chính được khai thác từ hình ảnh, đó là Color feature, Texture feature và Shape feature, và ứng với mỗi loại đó sẽ có các filter riêng chuyên biệt. Trong nội dung bài giảng này sẽ thực hiện đối với color feature và sử dụng filter *SimpleColorHistogramFilter*. Đây chính là phương pháp k-color histogram ( $k=64$ ) đã được đề cập tới trong mục trích chọn đặc trưng dữ liệu hình ảnh ở mục 2.4.1.
- Để sử dụng filter *ColorLayoutFilter* trong weka, ta cần cài đặt thêm package “imagefilter” cho Weka. Điều này được thực hiện như sau: Tại giao diện chính của Weka -> Tools -> Package manager -> Package search nhập vào *imagefilter* và chọn “install package” này vào cho Weka.



- Khi package imagefilter được install cho Weka, chúng ta thực hiện trích chọn đặc trưng hình ảnh theo phương pháp 64-Color histogram theo các bước thực hiện sau: Chọn Filter → unsupervised → instance → imageFilter → SimpleColorHistogramFilter



- Chọn vào tên filter để mở phần setting filter, tại đó thay thế *imageDirectory* bằng đường dẫn đến folder chứa các hình ảnh.



- Chọn **Apply** để sử dụng bộ lọc. Kết quả sẽ cho ra một tập các đặc trưng trích xuất được từ hình ảnh. Có 64 đặc trưng được tạo ra.

No.	Name
1	<input type="checkbox"/> RGB Color Histogram0
2	<input type="checkbox"/> RGB Color Histogram1
3	<input type="checkbox"/> RGB Color Histogram2
4	<input type="checkbox"/> RGB Color Histogram3
5	<input type="checkbox"/> RGB Color Histogram4
6	<input type="checkbox"/> RGB Color Histogram5
7	<input type="checkbox"/> RGB Color Histogram6
8	<input type="checkbox"/> RGB Color Histogram7
9	<input type="checkbox"/> RGB Color Histogram8
10	<input type="checkbox"/> RGB Color Histogram9
11	<input type="checkbox"/> RGB Color Histogram10
12	<input type="checkbox"/> RGB Color Histogram11
13	<input type="checkbox"/> RGB Color Histogram12
14	<input type="checkbox"/> RGB Color Histogram13
15	<input type="checkbox"/> RGB Color Histogram14
16	<input type="checkbox"/> RGB Color Histogram15
17	<input type="checkbox"/> RGB Color Histogram16
18	<input type="checkbox"/> RGB Color Histogram17

Remove

Status  
OK

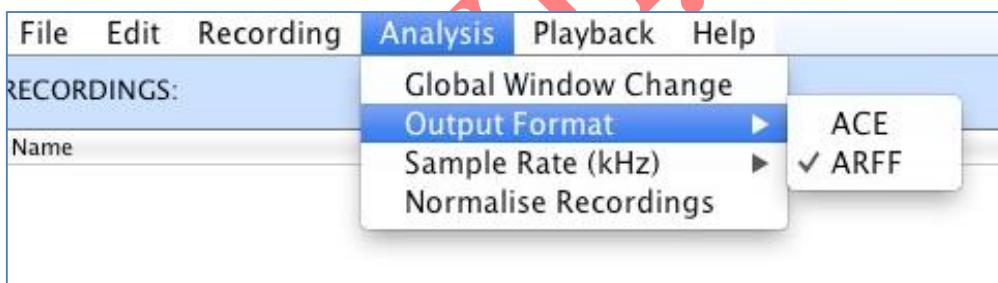
- Trên giao diện này chọn **Edit** để xem toàn bộ tập dữ liệu hình ảnh được biểu diễn dưới dạng bản ghi. Với mỗi bản ghi tương ứng với 1 vector đặc trưng của 1 hình ảnh.

No.	1: RGB Color Histogram0 Numeric	2: RGB Color Histogram1 Numeric	3: RGB Color Histogram2 Numeric	4: RGB Color Histogram3 Numeric	5: RGB Color Histogram4 Numeric	6: RGB Color Histogram5 Numeric	7: RGB Color Histogram6 Numeric
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	14.0	0.0	0.0	0.0	0.0	0.0	0.0
3	255.0	5.0	0.0	0.0	110.0	54.0	
4	230.0	0.0	0.0	0.0	10.0	0.0	
5	255.0	0.0	0.0	0.0	37.0	0.0	
6	38.0	0.0	0.0	0.0	6.0	0.0	
7	12.0	0.0	0.0	0.0	1.0	0.0	
8	4.0	0.0	0.0	0.0	6.0	0.0	
9	160.0	158.0	0.0	0.0	63.0	45.0	
10	255.0	0.0	0.0	0.0	91.0	6.0	
11	3.0	0.0	0.0	0.0	1.0	0.0	
12	10.0	2.0	0.0	0.0	0.0	0.0	

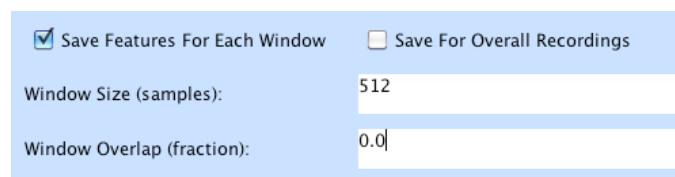
### (3) Trích chọn đặc trưng audio

Trong nội dung phần này khai thác thêm thư viện trích chọn đặc trưng cho audio đó là phần mềm jAudio, đây là thư viện dành riêng cho lựa chọn các phương pháp trích chọn đặc trưng cho audio khá phổ biến. Kết quả trích chọn được sẽ được dùng để xử lý khai phá dữ liệu trong Weka. Quá trình thực hiện miêu tả dưới đây:

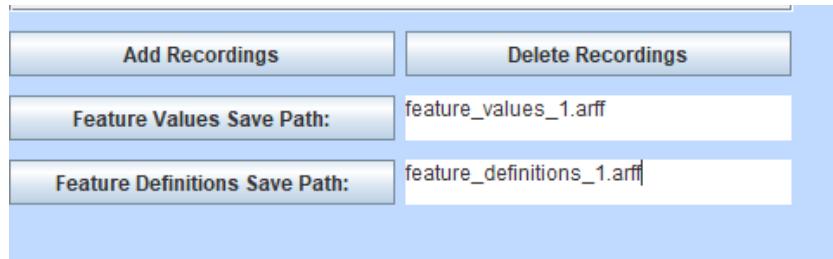
- Cài đặt thiết lập định dạng biểu diễn giữ liệu đầu ra sau trích chọn đặc trưng trong jAudio là arff để có thể sử dụng trong Weka.



- Thiết lập thông số: Thông thường các đặc trưng của audio được trích xuất trong một khoảng thời gian rất nhỏ (thường được gọi là khung) với sự chồng chéo. Đối với mỗi tệp âm thanh sẽ có nhiều cách lựa chọn các đặc trưng được sinh ra. Trong giao diện người dùng jAudio, chọn 'Save Features for Each Window', tùy chọn Window size , Window Overlap (mặc định giá trị cho 2 tham số này là 512, 0.



- Sau khi cấu hình, bắt đầu các tính năng extracting từ jAudio bằng cách nhấp vào nút ‘Add Recordings’ để chọn các tệp âm thanh. Sau đó lưu File đã “Extract Feature” dưới dạng .arff để import vào Weka.



- Kết quả sẽ lưu trong file feature\_values\_1.arff trong file cài đặt jAudio
- Lựa chọn trích chọn đặc trưng audio theo phương pháp MFCC, kết quả sẽ cho ra một tập các đặc trưng trích xuất và có thể mở trong Weka như sau:



trainAudio.arff - Notepad

Tệp Soạn thảo Định dạng Xem Trợ giúp

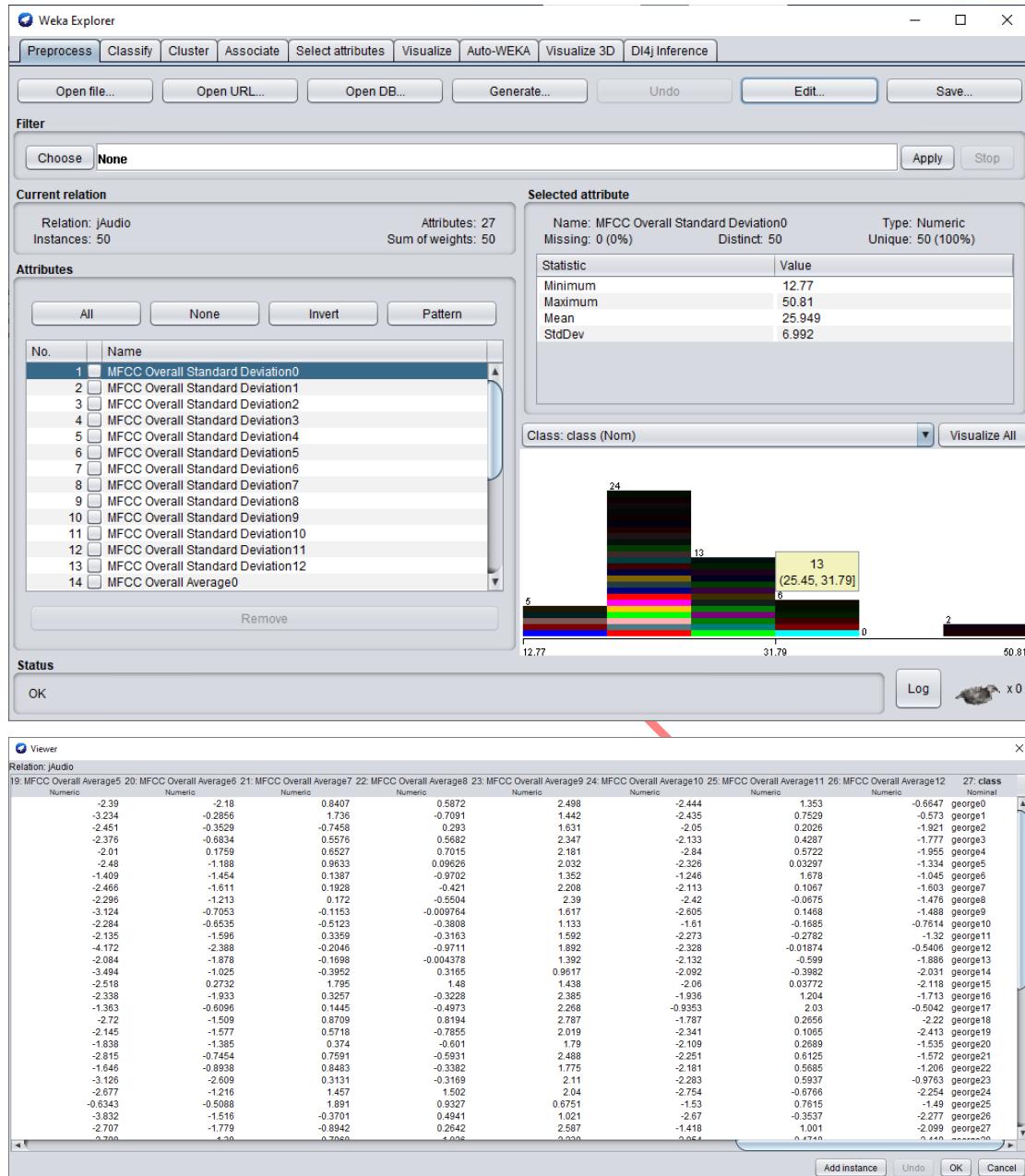
```

@ATTRIBUTE "MFCC Overall Average4" NUMERIC
@ATTRIBUTE "MFCC Overall Average5" NUMERIC
@ATTRIBUTE "MFCC Overall Average6" NUMERIC
@ATTRIBUTE "MFCC Overall Average7" NUMERIC
@ATTRIBUTE "MFCC Overall Average8" NUMERIC
@ATTRIBUTE "MFCC Overall Average9" NUMERIC
@ATTRIBUTE "MFCC Overall Average10" NUMERIC
@ATTRIBUTE "MFCC Overall Average11" NUMERIC
@ATTRIBUTE "MFCC Overall Average12" NUMERIC
@ATTRIBUTE class {george0,george1,george2,george3,george4,george5,george6,george7,
@DATA
1.277E1,6.504E0,7.479E0,3.954E0,3.635E0,2.362E0,2.101E0,3.914E0,1.855E0,2.197E0,1.
2.284E1,5.343E0,7.339E0,3.143E0,4.452E0,2.748E0,2.403E0,1.857E0,1.42E0,1.776E0,1.7
3.414E1,7.595E0,5.764E0,2.532E0,4.749E0,2.549E0,1.324E0,2.016E0,1.792E0,2.214E0,1.
2.478E1,5.728E0,6.159E0,2.913E0,4.087E0,3.161E0,2.115E0,2.595E0,1.558E0,2.433E0,1.
2.046E1,5.806E0,5.072E0,2.493E0,4.867E0,2.546E0,1.892E0,2.059E0,1.678E0,1.863E0,1.
2.287E1,6.927E0,5.656E0,2.207E0,5.029E0,2.292E0,2.731E0,2.53E0,1.673E0,1.808E0,1.9
2.197E1,8.037E0,5.925E0,3.473E0,3.952E0,2.837E0,2.38E0,2.475E0,1.486E0,1.452E0,1.6
2.381E1,6.583E0,5.176E0,3.734E0,4.082E0,2.767E0,2.387E0,3.196E0,1.754E0,1.609E0,1.
2.405E1,6.785E0,5.114E0,2.545E0,5.661E0,2.713E0,2.799E0,2.195E0,1.273E0,1.841E0,2.
2.734E1,5.54E0,5.366E0,2.751E0,4.967E0,3.111E0,2.015E0,2.438E0,1.653E0,1.549E0,1.9
2.523E1,6.346E0,5.953E0,3.474E0,5.68E0,2.926E0,2.49E0,2.653E0,1.528E0,1.509E0,1.76
1.896E1,6.898E0,6.075E0,3.405E0,5.032E0,2.489E0,2.148E0,2.816E0,1.184E0,1.275E0,1.
2.549E1,6.899E0,5.666E0,3.753E0,4.102E0,2.31E0,2.852E0,2.444E0,1.652E0,1.679E0,1.9
1.953E1,6.317E0,5.887E0,3.375E0,3.563E0,2.22E0,2.933E0,2.069E0,1.648E0,1.628E0,1.8
1.85E1,6.464E0,5.261E0,2.78E0,3.053E0,1.823E0,2.23E0,2.31E0,1.527E0,1.616E0,1.398E
2.586E1,6.937E0,5.721E0,3.115E0,3.706E0,2.278E0,2.025E0,2.175E0,1.933E0,1.757E0,2E
2.05E1,6.128E0,7.902E0,4.737E0,2.132E0,2.16E0,2.693E0,2.77E0,2.634E0,1.614E0,1.313
2.692E1,7.513E0,6.183E0,3.45E0,2.381E0,2.828E0,2.08E0,2.617E0,1.652E0,2.141E0,1.40

```

Dòng 1, Cột 1 100% Windows (CRLF) UTF-8

- Kết quả mở trong Weka để xem danh sách các đặc trưng được sinh ra và biểu diễn tập dữ liệu audio dưới dạng bản ghi qua các vector đặc trưng (chọn Edit).



#### (4) Trích chọn đặc trưng video

Nối tiếp kiến thức nêu ra trong nội dung trích chọn đặc trưng dữ liệu video (mục 2.4.1.4), nội dung phần này sẽ trình bày quá trình thực hiện trích chọn đặc trưng video với Weka.

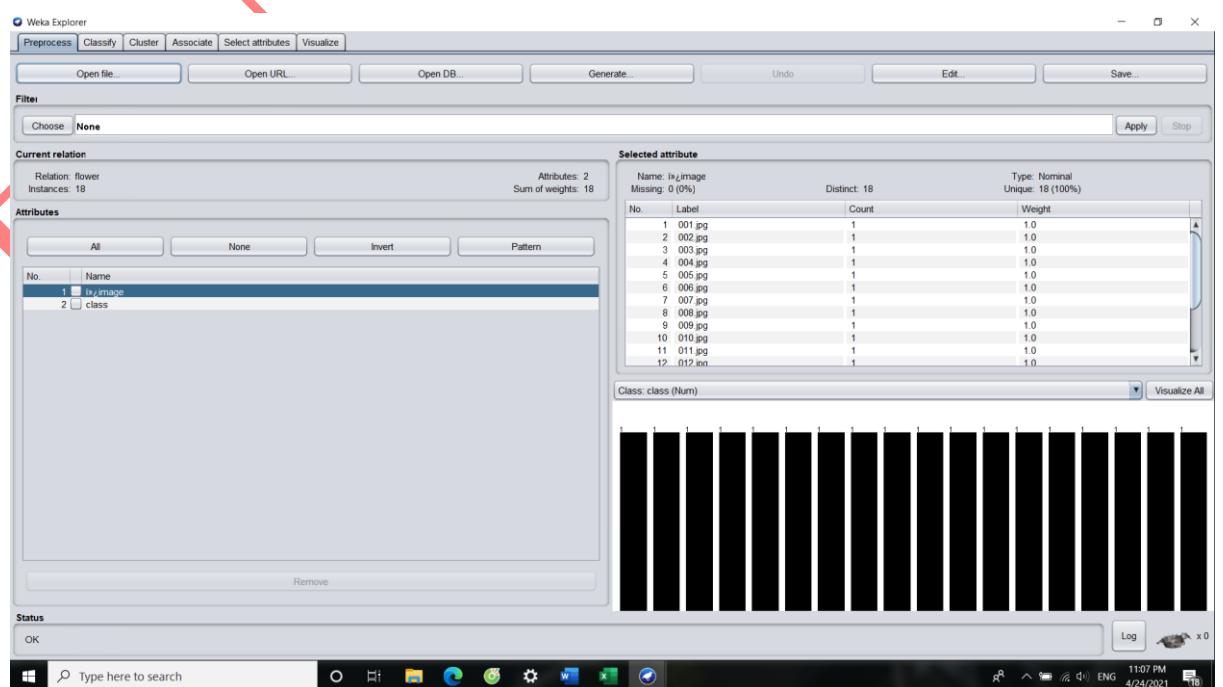
- Thực hiện cắt các keyframe từ video, có nhiều phần mềm có thể dùng cho mục đích này, kết quả giả sử việc cắt các keyframe từ video về hoa cho ra kết quả là 1 tập hợp các hình ảnh sau:



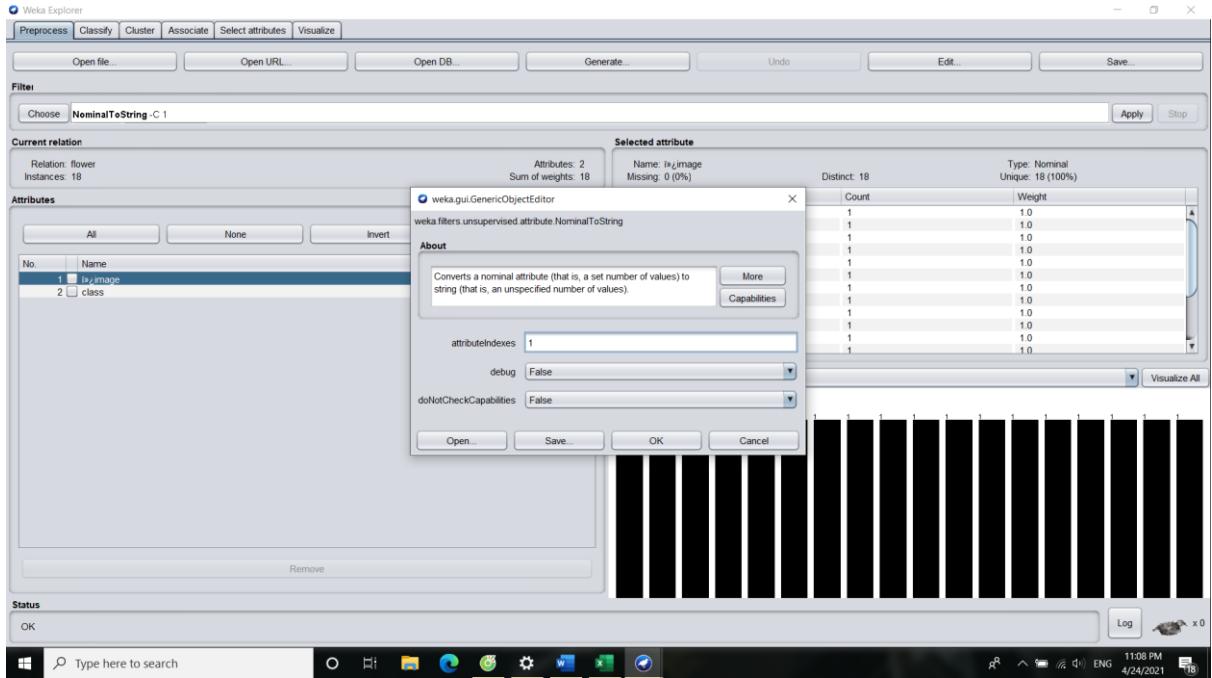
- Tạo 1 dataset gồm 2 cột image, class và lưu dưới dạng flower.csv

	A	B	C
1	image	class	
2	001.jpg	0	
3	002.jpg	1	
4	003.jpg	0	
5	004.jpg	1	
6	005.jpg	0	
7	006.jpg	0	
8	007.jpg	1	
9	008.jpg	1	
10	009.jpg	1	
11	010.jpg	1	
12	011.jpg	0	
13	012.jpg	0	
14	013.jpg	0	
15	014.jpg	1	
16	015.jpg	1	
17	016.jpg	0	
18	017.jpg	0	
19	018.jpg	1	

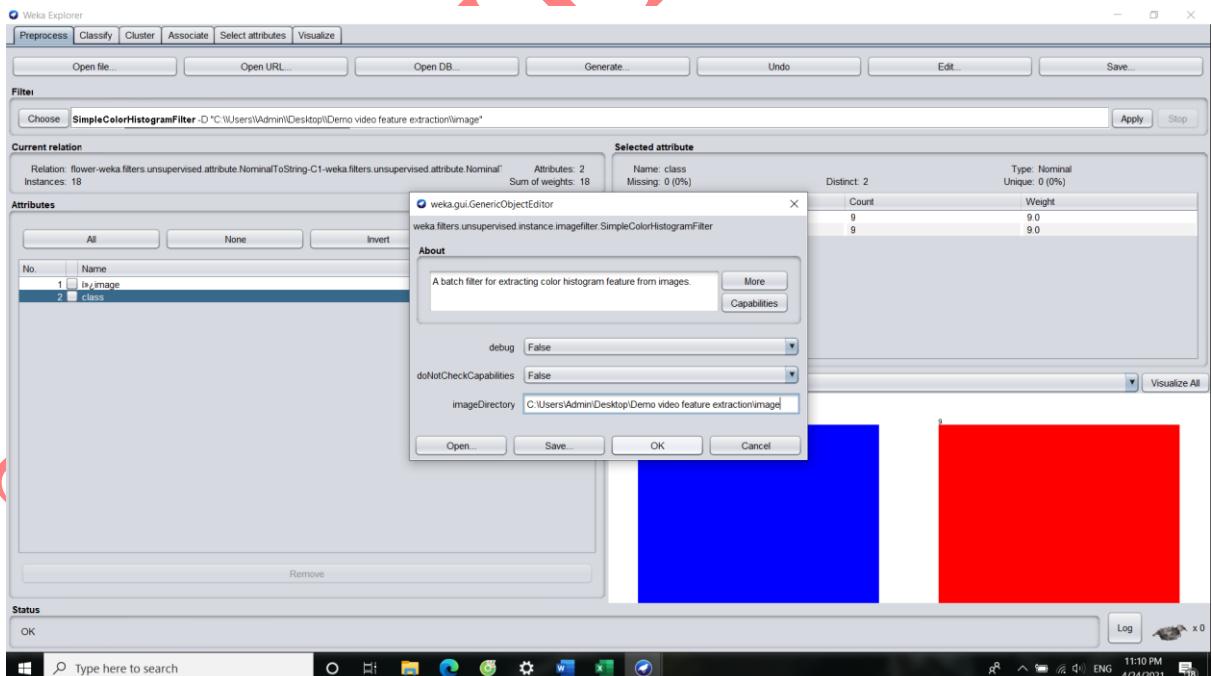
- Mở file dataset vừa tạo trong Weka:



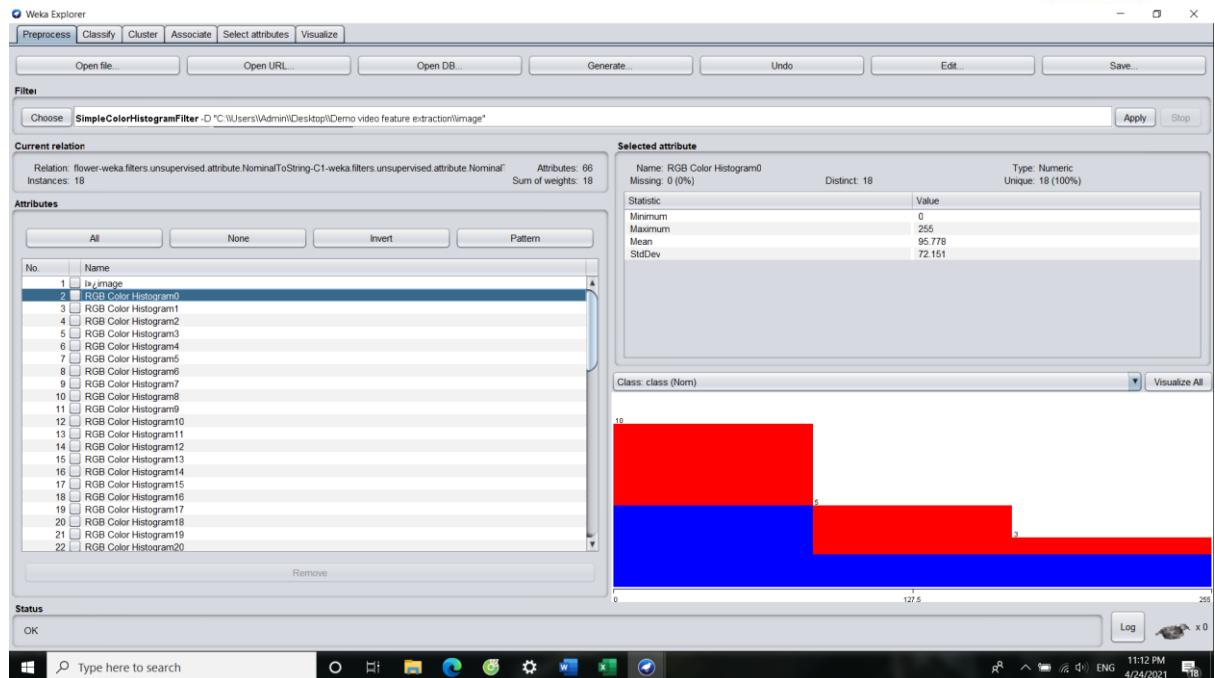
- Sử dụng filter **nominalToString** áp dụng chuyển kiểu dữ liệu của thuộc tính image



- Sử dụng filter **SimpleColorHistogramFilter** để trích chọn đặc trưng cho các frame của video theo phương pháp k-color histogram ( $k=64$ )



- Kết quả mở trong Weka để xem danh sách các đặc trưng được sinh ra và biểu diễn dữ liệu video dưới dạng bản ghi qua các vector đặc trưng.



### 2.5.3.2. *Dọn dẹp dữ liệu với Weka*

Dữ liệu đa phương tiện sau khi thực hiện trích chọn đặc trưng sẽ được đưa về biểu diễn dưới dạng bản ghi, trong đó mỗi bản ghi là một vector đặc trưng tương ứng với từng đối tượng dữ liệu. Bốn công việc chính cần thực hiện để dọn dẹp dữ liệu đã được đề cập trong mục 2.4.2, đó là : Tích hợp dữ liệu, chuyển đổi dữ liệu, thu giảm dữ liệu và làm sạch dữ liệu. Để thực hiện được việc này, Weka hỗ trợ những bộ lọc (filter) và lệnh tương ứng cho từng mục tiêu dọn dẹp.

#### (1) Tích hợp dữ liệu với Weka

Giả sử dữ liệu đầu vào là 2 bộ dữ liệu training.arff và test.arff được tách ra từ tập dữ liệu weather-numeric.arff. Việc tích hợp dữ liệu có thể gồm một số công việc: 1/ Nối 2 file dữ liệu; 2/ Hợp nhất 2 file dữ liệu

- Nối 2 file dữ liệu bằng lệnh trong Simple CLI của weka:

```
java weka.core.Instances append i:/training.arff i:/test.arff > i:/append.arff
```

Kết quả nhận được:

No.	1: outlook Nominal	2: temperature Numeric	3: humidity Numeric	4: windy Nominal	5: play Nominal
1	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes
4	rainy	70.0	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	70.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no
9	sunny	69.0	70.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes
12	overcast	72.0	90.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes
14	rainy	71.0	91.0	TRUE	no
15	sunny	85.0	85.0	FALSE	no
16	sunny	80.0	90.0	TRUE	no
17	overcast	83.0	86.0	FALSE	yes
18	rainy	70.0	96.0	FALSE	yes
19	rainy	68.0	80.0	FALSE	yes
20	rainy	65.0	70.0	TRUE	no
21	overcast	64.0	65.0	TRUE	yes
22	sunny	72.0	95.0	FALSE	no
23	sunny	69.0	70.0	FALSE	yes
24	rainy	75.0	80.0	FALSE	yes
25	sunny	75.0	70.0	TRUE	yes
26	overcast	72.0	90.0	TRUE	yes
27	overcast	81.0	75.0	FALSE	yes
28	rainy	71.0	91.0	TRUE	no

- Hợp nhất 2 file dữ liệu bằng lệnh trong Simple CLI của weka:

```
java weka.core Instances merge i:/training.arff i:/test.arff > i:/merge.arff \
```

Kết quả nhận được:

No.	1: outlook Nominal	2: temperature Numeric	3: humidity Numeric	4: windy Nominal	5: play Nominal	6: outlook2 Nominal	7: temperature2 Numeric	8: humidity2 Numeric	9: windy2 Nominal	10: play2 Nominal
1	sunny	85.0	85.0	FALSE	no	sunny	85.0	85.0	FALSE	no
2	sunny	80.0	90.0	TRUE	no	sunny	80.0	90.0	TRUE	no
3	overcast	83.0	86.0	FALSE	yes	overcast	83.0	86.0	FALSE	yes
4	rainy	70.0	96.0	FALSE	yes	rainy	70.0	96.0	FALSE	yes
5	rainy	68.0	80.0	FALSE	yes	rainy	68.0	80.0	FALSE	yes
6	rainy	65.0	70.0	TRUE	no	rainy	65.0	70.0	TRUE	no
7	overcast	64.0	65.0	TRUE	yes	overcast	64.0	65.0	TRUE	yes
8	sunny	72.0	95.0	FALSE	no	sunny	72.0	95.0	FALSE	no
9	sunny	69.0	70.0	FALSE	yes	sunny	69.0	70.0	FALSE	yes
10	rainy	75.0	80.0	FALSE	yes	rainy	75.0	80.0	FALSE	yes
11	sunny	75.0	70.0	TRUE	yes	sunny	75.0	70.0	TRUE	yes
12	overcast	72.0	90.0	TRUE	yes	overcast	72.0	90.0	TRUE	yes
13	overcast	81.0	75.0	FALSE	yes	overcast	81.0	75.0	FALSE	yes
14	rainy	71.0	91.0	TRUE	no	rainy	71.0	91.0	TRUE	no

Ngoài ra có một số bộ lọc khác có thể được khai thác cho mục đích hợp nhất giá trị từ các thuộc tính: MergeTwoValues, MergeManyValues

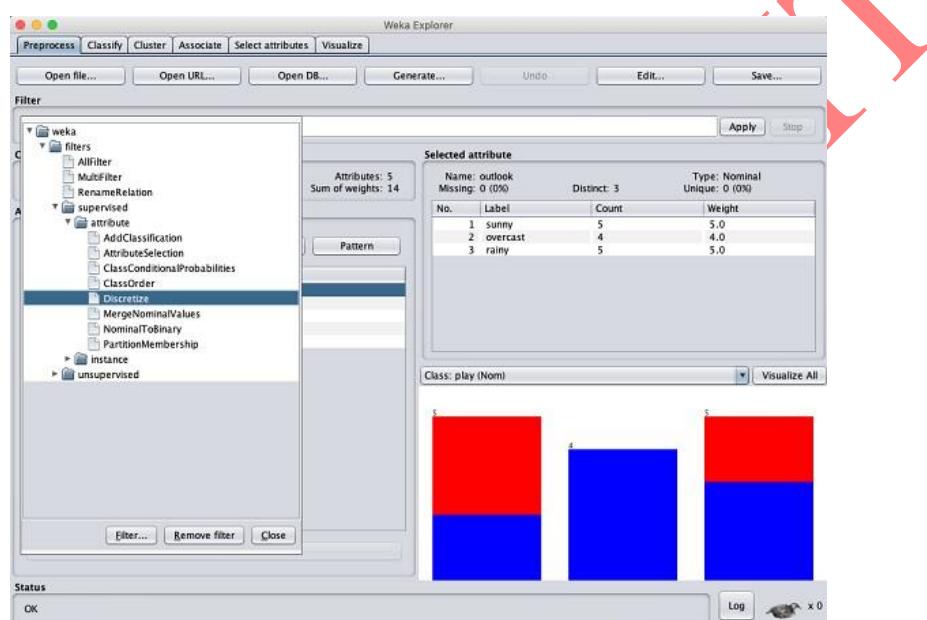
## (2) Chuyển đổi dữ liệu với Weka

Theo cơ sở lý thuyết về quá trình chuyển đổi dữ liệu về cơ bản gồm hai công việc chính là tiêu chuẩn hóa dữ liệu (Standardize) và chuẩn hóa dữ liệu (Normalization). Nội dung phần này sẽ minh họa việc sử dụng một số filter cơ bản cho 2 mục đích này.

- Tiêu chuẩn hóa dữ liệu (Standardize)

Một số bộ lọc thường được sử dụng cho công việc này là: Rời rạc hóa dữ liệu, chuyển đổi thuộc tính. Nội dung dưới đây sẽ minh họa các quá trình này trong Weka áp dụng cho tập dữ liệu weather-numeric.arff.

- Rời rạc hóa dữ liệu: Nhằm chuyển đổi giá trị dữ liệu từ miền liên tục (Ví dụ đối với thuộc tính có kiểu numeric) về miền rời rạc (Thuộc tính có kiểu nominal). Thủ nghiệm trên bộ dữ liệu weather-numeric, việc này được thực hiện bằng việc áp dụng bộ lọc  
**weka → filters → supervised → attribute → Discretize** nhằm chia khoảng cho giá trị của các thuộc tính temperature và humidity



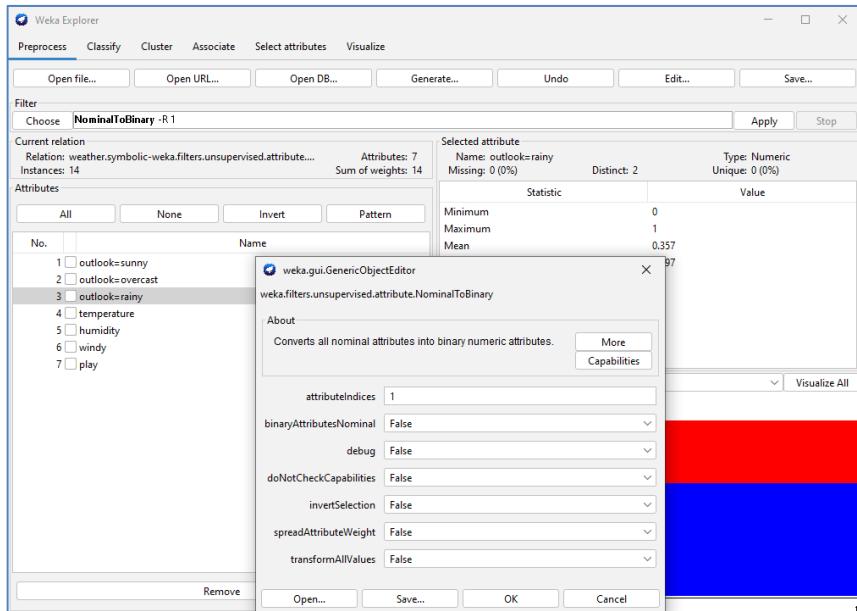
Chọn **Apply** để sử dụng bộ lọc. Kiểm tra kiểu giá trị cho các thuộc tính temperature và humidity đã chuyển từ numeric sang nominal.

Name: temperature		Distinct: 1	Type: Nominal
Missing: 0 (0%)		Unique: 0 (0%)	
No.	Label	Count	Weight
1	'All'	14	14.0

- Chuyển đổi thuộc tính từ kiểu dữ liệu nominal sang binary nhằm tạo ra các thuộc tính giả. Mỗi biến giả này nhận giá trị nhị phân cho từng giá trị của biến có kiểu dữ liệu nominal

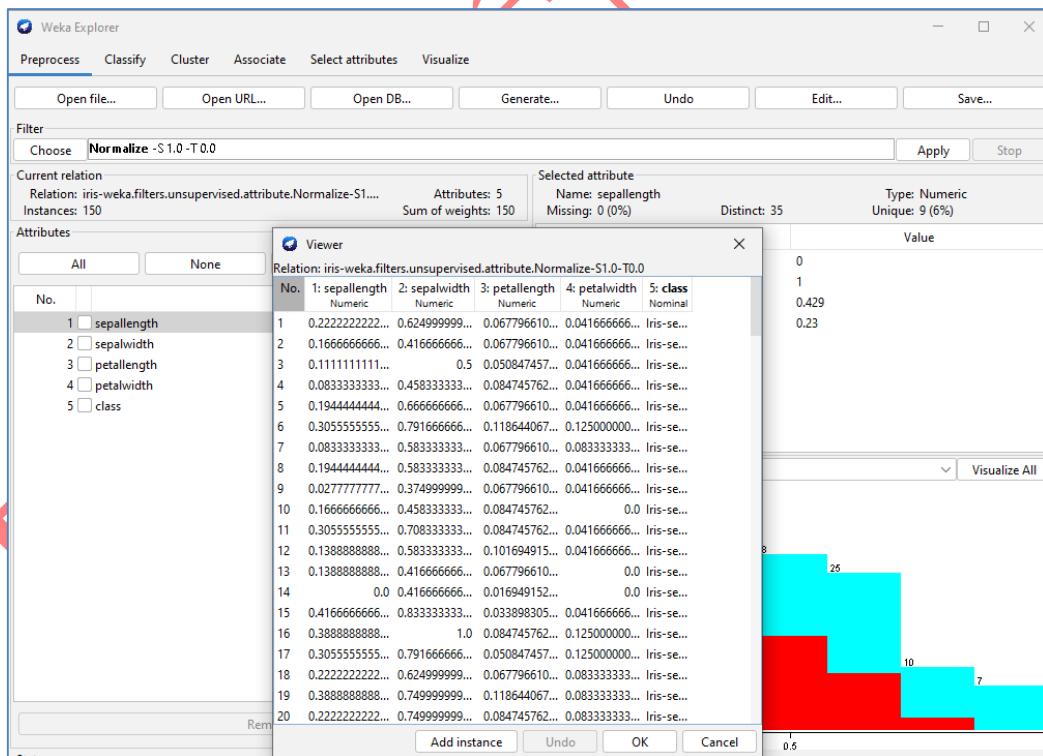
Minh họa áp dụng bộ lọc

**weka → filters → unsupervised → attribute → NominalToBinary** nhằm chuyển 1 thuộc tính outlook có kiểu dữ liệu nominal (nhận các giá trị: sunny, overcast, rainy) thành 3 thuộc tính giả có kiểu dữ liệu binary (outlook=sunny, outlook=overcast, outlook=rainy). Các thuộc tính giả tương ứng với từng giá trị của thuộc tính gốc sẽ nhận giá trị 1 và 0 trong trường hợp còn lại.



### - Chuẩn hóa dữ liệu (Normalization)

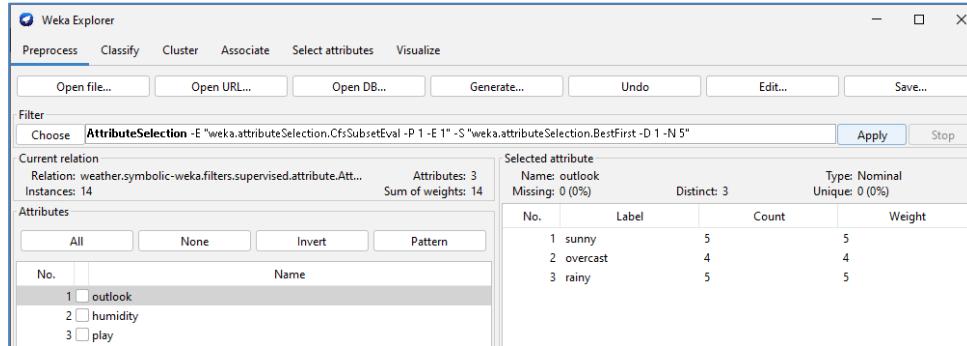
Áp dụng bộ lọc **weka→filters→unsupervised→Normalize** cho tập dữ liệu iris.arff nhằm chuyển đổi giá trị dữ liệu gốc của các thuộc tính về miền giá trị [0,1] để thuận tiện cho việc học dữ liệu sau này



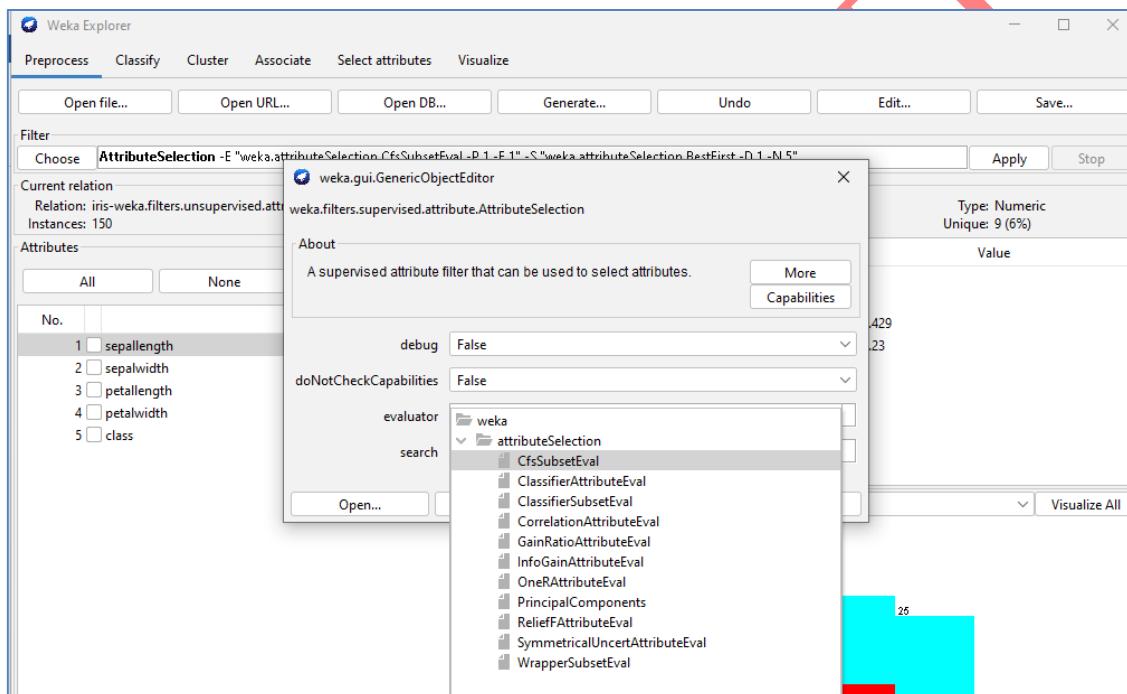
### (3) Thu giảm dữ liệu với Weka

Một số bộ lọc thường được sử dụng cho công việc này lựa chọn thuộc tính. Minh họa với tập dữ liệu weather-numeric.arff, việc này được thực hiện bằng việc áp dụng bộ lọc **weka→filters→supervised→attribute→AttributeSelection** nhằm chọn lọc ra các

thuộc tính quyết định tới việc chơi thể thao hay không. Như vậy sẽ có 3 thuộc tính được giữ lại là outlook, humidity, play.



Weka hỗ trợ nhiều thuật toán lựa chọn thuộc tính khác nhau:



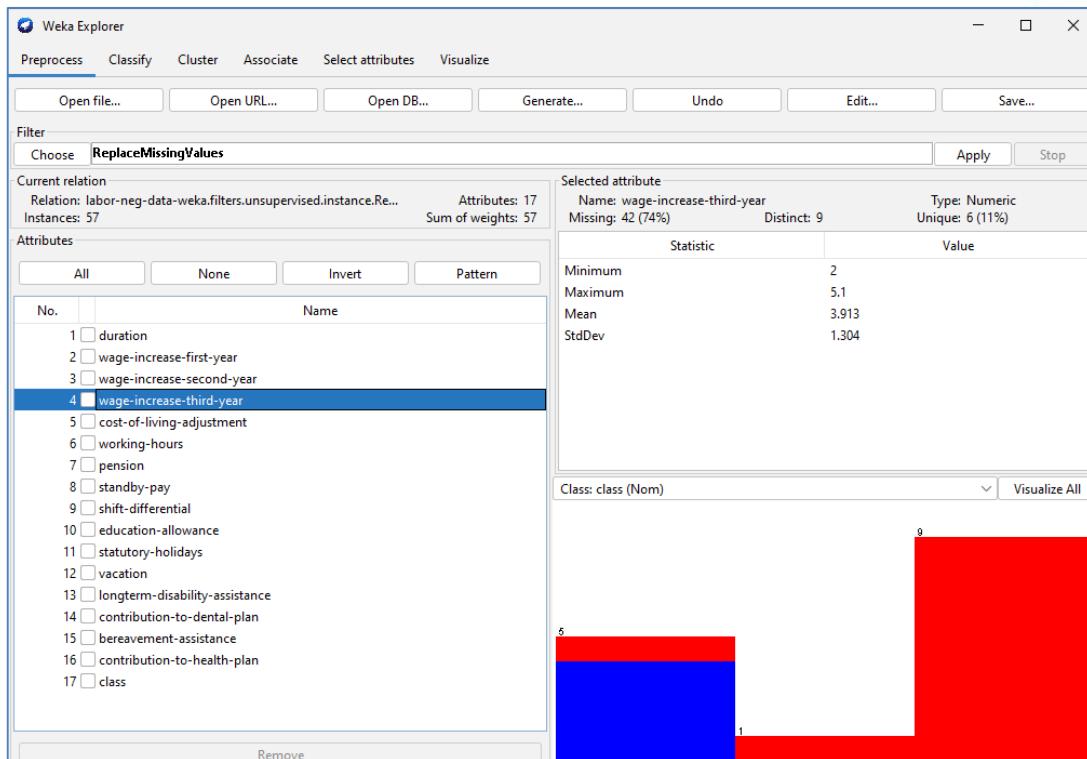
#### (4) Làm sạch dữ liệu với Weka

Một số bộ lọc thường được sử dụng cho công việc này là: Xóa bỏ dữ liệu trùng lặp, xử lý dữ liệu thiếu. Nội dung dưới đây sẽ minh họa các quá trình này trong Weka áp dụng cho tập dữ liệu labor.arff.

- Xóa bỏ những đối tượng dữ liệu trùng lặp nhằm tạo ra thể hiện duy nhất của mỗi đối tượng dữ liệu:

Sử dụng bộ lọc **weka→filters→unsupervised→instance→RemoveDuplicates**

- Xử lý dữ liệu thiếu: Có rất nhiều giá trị thiếu cho các thuộc tính trong tập dữ liệu labor.arff



Sử dụng bộ lọc weka → filters → unsupervised → attribute → ReplaceMissingValues.  
Quan sát kết quả trước và sau khi xử lý những giá trị còn thiếu của các thuộc tính.

Trước khi xử lý Missing values:

Viewer

Relation: labor-neg-data-weka.filters.unsupervised.instance.RemoveDuplicates

No.	1: duration	2: wage-increase-first-year	3: wage-increase-second-year	4: wage-increase-third-year	5: cost-of-living-adjustment	6: working-hours	7: pension	8: standby-p
1	1.0		5.0			40.0		
2	2.0		4.5	5.8		35.0	ret_allw	
3						38.0	empl_co...	
4	3.0		3.7	4.0	5.0 tc			
5	3.0		4.5	4.5	5.0		40.0	
6	2.0		2.0	2.5			35.0	
7	3.0		4.0	5.0	5.0 tc			empl_co...
8	3.0		6.9	4.8	2.3		40.0	
9	2.0		3.0	7.0			38.0	
10	1.0		5.7		none		40.0	empl_co...
11	3.0		3.5	4.0	4.6 none		36.0	
12	2.0		6.4	6.4			38.0	
13	2.0		3.5	4.0	none		40.0	
14	3.0		3.5	4.0	5.1 tcf		37.0	
15	1.0		3.0		none		36.0	
16	2.0		4.5	4.0	none		37.0	empl_co...
17	1.0		2.8				35.0	
18	1.0		2.1		tr		40.0	ret_allw

Sau khi xử lý Missing values:

No.	1: duration	2: wage-increase-first-year	3: wage-increase-second-year	4: wage-increase-third-year	5: cost-of-living-adjustment	6: working-hours	7: pension	8: standby-p
	Numeric	Numeric	Numeric	Numeric	Nominal	Numeric	Nominal	Numeric
1	1.0		5.0	3.971739130434783	3.9133333333333336 none		40.0 empl_c...	7.444444444
2	2.0		4.5		3.9133333333333336 none		35.0 ret_allw	7.444444444
3	2.1607142...	3.803571428571428		3.971739130434783	3.9133333333333336 none		38.0 empl_c...	7.444444444
4	3.0		3.7	4.0		5.0 tc	38.03921568627...	empl_c...
5	3.0		4.5	4.5		5.0 none	40.0 empl_c...	7.444444444
6	2.0		2.0	2.5	3.9133333333333336 none		35.0 empl_c...	7.444444444
7	3.0		4.0	5.0		5.0 tc	38.03921568627...	empl_c...
8	3.0		6.9	4.8		2.3 none	40.0 empl_c...	7.444444444
9	2.0		3.0	7.0	3.9133333333333336 none		38.0 empl_c...	
10	1.0		5.7	3.971739130434783	3.9133333333333336 none		40.0 empl_c...	7.444444444
11	3.0		3.5	4.0		4.6 none	36.0 empl_c...	7.444444444
12	2.0		6.4	6.4	3.9133333333333336 none		38.0 empl_c...	7.444444444
13	2.0		3.5	4.0	3.9133333333333336 none		40.0 empl_c...	7.444444444
14	3.0		3.5	4.0		5.1 tcf	37.0 empl_c...	7.444444444
15	1.0		3.0	3.971739130434783	3.9133333333333336 none		36.0 empl_c...	7.444444444
16	2.0		4.5	4.0	3.9133333333333336 none		37.0 empl_c...	7.444444444
17	1.0		2.8	3.971739130434783	3.9133333333333336 none		35.0 empl_c...	7.444444444
18	1.0		2.1	3.971739130434783	3.9133333333333336 tc		40.0 ret_allw	
19	1.0		2.0	3.971739130434783	3.9133333333333336 none		38.0 none	7.444444444
20	2.0		4.0	5.0	3.9133333333333336 tcf		35.0 empl_c...	
21	2.0		4.3	4.4	3.9133333333333336 none		38.0 empl_c...	7.444444444
22	2.0		2.5	3.0	3.9133333333333336 none		40.0 none	7.444444444
23	3.0		3.5	4.0		4.6 tcf	27.0 empl_c...	7.444444444

Như vậy, Trong nội dung này đã minh họa việc áp dụng một số filter cho từng mục tiêu tiền xử lý dữ liệu cơ bản. Có khá nhiều những bộ lọc khác được hỗ trợ bởi Weka, việc mở rộng tìm hiểu các bộ lọc khác cho tiền xử lý dữ liệu được coi như một bài tập của phần này.

## TỔNG KẾT CHƯƠNG 2

Nội dung chương 2 đã trình bày cụ thể về pha đầu tiên trong quy trình khai phá dữ liệu đa phương tiện được đề cập trong chương 1, đó là tiền xử lý dữ liệu và một số công việc liên quan. Đó là:

- Hiểu về cách biểu diễn và mô tả dữ liệu trên máy tính cho mục đích khai phá dữ liệu
- Khái niệm về tiền xử lý dữ liệu đa phương tiện
- Tầm quan trọng của tiền xử lý dữ liệu đa phương tiện
- Các nhiệm vụ chính của tiền xử lý dữ liệu đa phương tiện: trích chọn đặc trưng và lọc dữ liệu (lọc sạch, tích hợp, biến đổi, thu giảm dữ liệu).
- Áp dụng những cơ sở lý thuyết về tiền xử lý vào thực hành tiền xử lý dữ liệu đa phương tiện với Weka.

Trong chương 3 tiếp theo, bài giảng đi sâu về việc áp dụng các mô hình khai phá dữ liệu lên các dữ liệu đa phương tiện đã được tiền xử lý cho từng mục tiêu ứng dụng.

## CÂU HỎI VÀ BÀI TẬP CHƯƠNG 2

1. Nêu các cách để biểu diễn dữ liệu đa phương tiện? Trong số các cách biểu diễn đó thì cách nào là phổ biến để biểu diễn tập dữ liệu đa phương tiện nhằm thuận tiện cho máy tính xử lý khai phá dữ liệu? Cho ví dụ minh họa.
2. Trình bày hiểu biết của em về phương pháp trích chọn đặc trưng văn bản Bag of words? Thực hiện trích chọn đặc trưng cho tập dữ liệu văn bản sử dụng kỹ thuật Bag of words trong Weka.
3. Trình bày hiểu biết của em về phương pháp trích chọn đặc trưng văn bản TF-IDF? Thực hiện trích chọn đặc trưng cho tập dữ liệu sử dụng kỹ thuật TF-IDF trong Weka.
4. Trình bày hiểu biết của em về phương pháp trích chọn đặc trưng hình ảnh k-color histogram? Thực hiện trích chọn đặc trưng cho tập dữ liệu hình ảnh sử dụng kỹ thuật k-color histogram trong Weka.
5. Trình bày hiểu biết của em về phương pháp trích chọn đặc trưng audio MFCC? Thực hiện trích chọn đặc trưng cho tập dữ liệu audio sử dụng kỹ thuật MFCC
6. Trình bày hiểu biết của em về phương pháp trích chọn đặc trưng video? Thực hiện trích chọn đặc trưng cho tập dữ liệu video trong Weka.
7. Thực hiện dọn dẹp dữ liệu sau trích chọn đặc trưng cho tập dữ liệu văn bản trong Weka
8. Thực hiện dọn dẹp dữ liệu sau trích chọn đặc trưng cho tập dữ liệu audio trong Weka
9. Thực hiện dọn dẹp dữ liệu sau trích chọn đặc trưng cho tập dữ liệu video trong Weka

THƯ VIỆN

## CHƯƠNG 3

### MÔ HÌNH KHAI PHÁ DỮ LIỆU ĐA PHƯƠNG TIỆN

Nội dung chương này tập trung trình bày về một số mô hình khai phá dữ liệu đa phương tiện phổ biến, đó là:

- Mô hình phân lớp dữ liệu
- Mô hình phân cụm dữ liệu
- Mô hình khai phá luật kết hợp

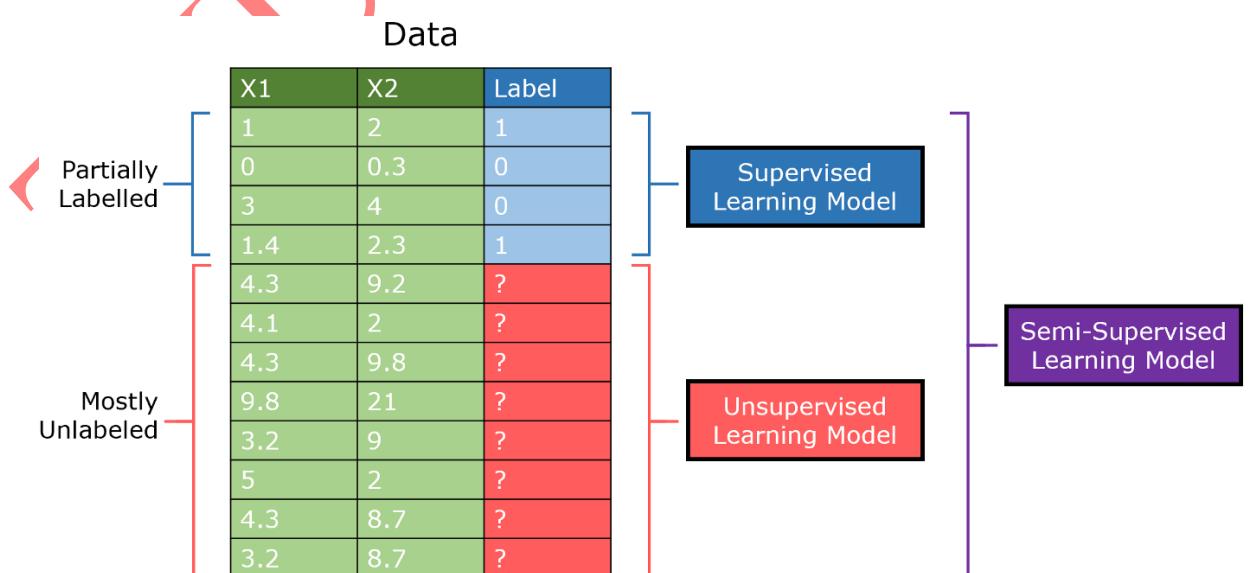
Ứng với mỗi mô hình, bài giảng tiếp cận từ phát biểu bài toán, áp dụng một số thuật toán khai phá dữ liệu điển hình cho bài toán tiếp cận và đánh giá độ chính xác của các mô hình áp dụng. Bên cạnh đó để có thể tiếp cận thực tế cho sinh viên, bài giảng sẽ đưa nội dung thực hành quá trình xây dựng và đánh giá mô hình khai phá dữ liệu tương ứng trong môi trường Weka tương ứng với từng nội dung liên quan.

#### 3.1. Phân lớp dữ liệu

##### 3.1.1. Bài toán phân lớp dữ liệu

Phân lớp dữ liệu (Classification) là một nhiệm vụ yêu cầu sử dụng các thuật toán học máy để dự đoán nhãn lớp cho đối tượng dữ liệu mới trên cơ sở khai thác thông tin từ một tập hợp các đối tượng dữ liệu đã có trong hệ thống.

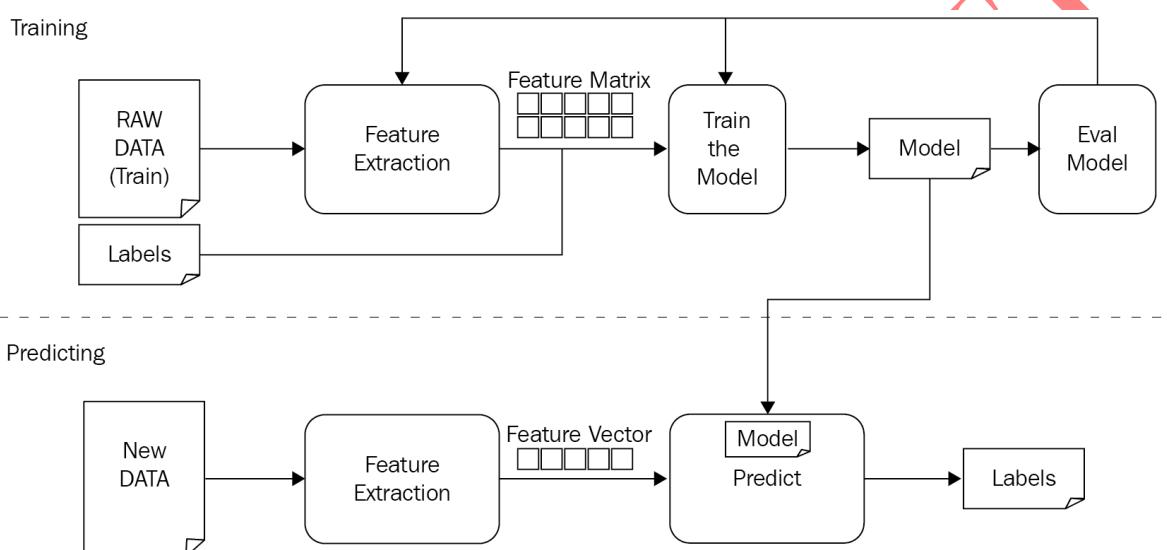
Các nghiên cứu về học máy dựa trên phương thức học dữ liệu chỉ ra rằng có ba hướng tiếp cận học máy cơ bản, đó là: 1) Học có giám sát (Supervised learning); 2) Học không giám sát (Unsupervised learning); 3) Học bán giám sát (Semi-supervised learning). Điểm khác nhau của 3 loại mô hình học máy này là dựa vào tính chất của tập dữ liệu đầu vào đã được gán nhãn biết trước hay không.



Hình 3.1. Minh họa các hướng tiếp cận học máy cơ bản dựa trên phương thức học dữ liệu

Trong các hướng tiếp cận này thì hướng tiếp cận học có giám sát và học bán giám sát có thể coi là phù hợp để giải quyết bài toán phân lớp dữ liệu. Trong đó học máy có giám sát được xây dựng dựa trên một tập hợp các đối tượng dữ liệu đều đã được gán nhãn lớp biết trước, trong khi mô hình học máy bán giám sát được xây dựng với tập dữ liệu có một số đối tượng dữ liệu có nhãn lớp và một số đối tượng dữ liệu chưa có nhãn lớp cho trước. Nội dung dưới đây đưa ra quá trình phân lớp dữ liệu theo 2 hướng tiếp cận học máy này.

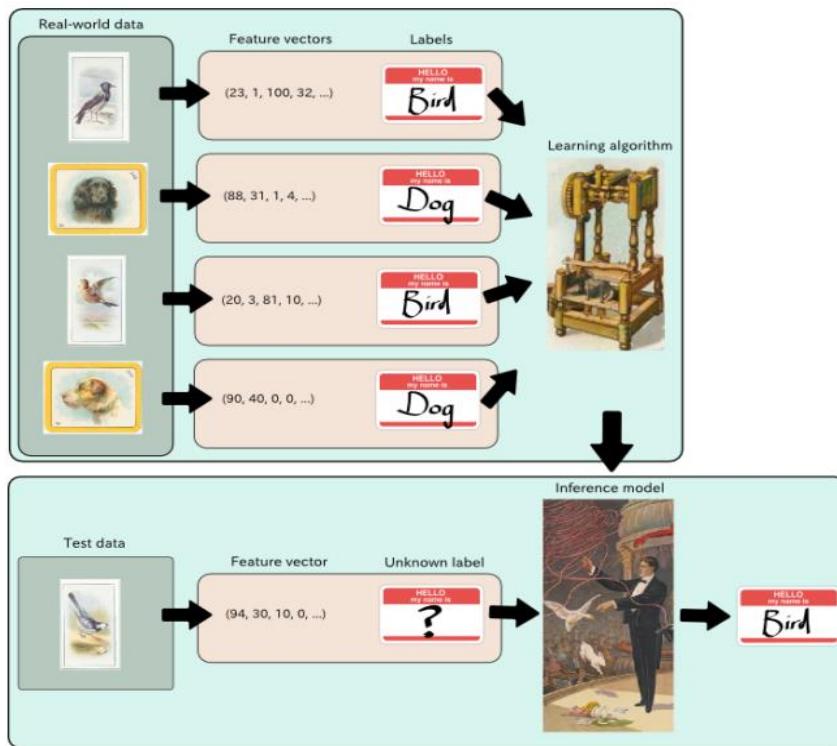
- Quá trình phân lớp dữ liệu theo hướng tiếp cận học máy có giám sát :



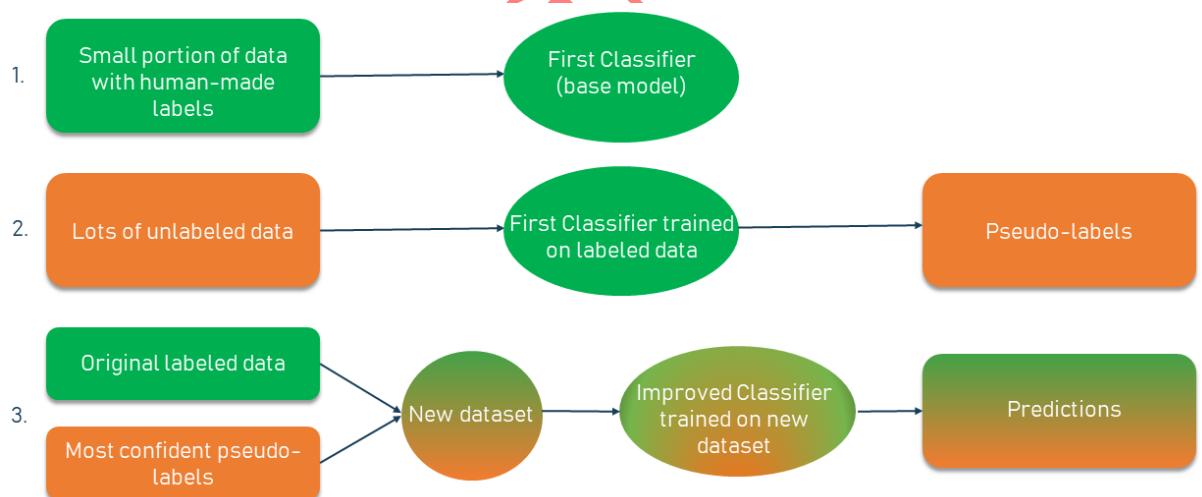
Hình 3.2. Quá trình phân lớp dữ liệu theo hướng tiếp cận học máy có giám sát

Áp dụng với việc phân loại ảnh:

THU

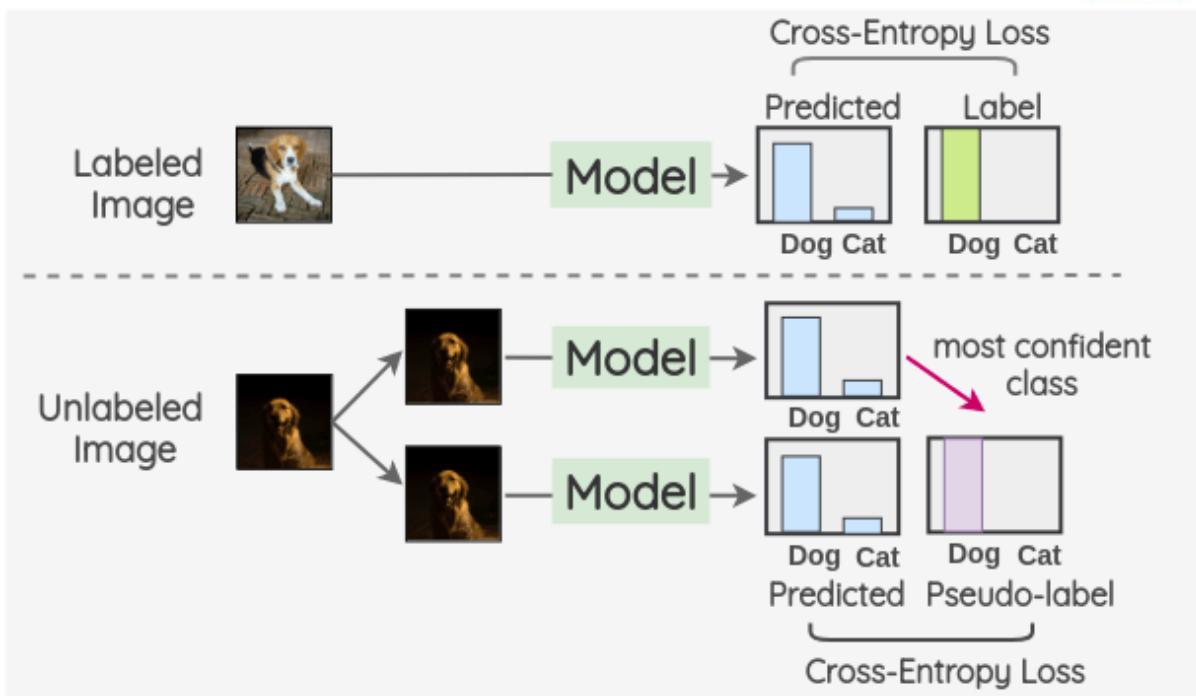


- Quá trình phân lớp dữ liệu theo hướng tiếp cận học máy bán giám sát :

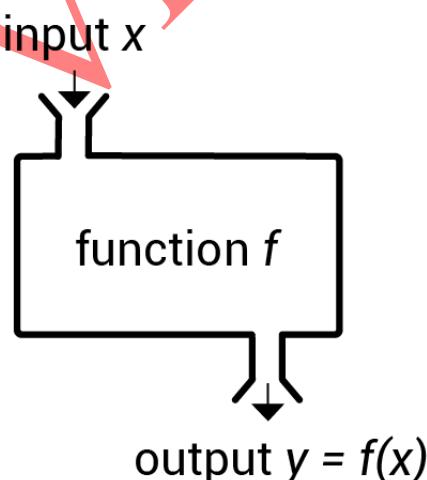


Hình 3.3. Quá trình phân lớp dữ liệu theo hướng tiếp cận học máy bán giám sát

Ví dụ ứng dụng học bán giám sát trong phân loại ảnh động vật:



Như vậy tiếp cận giải quyết bài toán phân lớp dữ liệu theo một trong hai hướng học máy có giám sát hay học máy bán giám sát đều nhằm mục tiêu tìm ra một mô hình dữ liệu giúp dự đoán nhãn lớp cho đối tượng dữ liệu mới. Biểu diễn dưới dạng toán học, mô hình dữ liệu học được có thể biểu diễn thông qua một hàm số  $f(x)$  giúp ánh xạ từ đối tượng dữ liệu đầu vào  $x$  thành nhãn lớp  $y$  ở đầu ra. Quá trình này được mô phỏng đơn giản như sau:



Theo quá trình phân lớp được chỉ ra trong Hình 3.1 và Hình 3.2 thì quá trình này sẽ gồm có 3 công việc chính kết hợp với nhau : 1) Chuẩn bị dữ liệu cho quá trình phân lớp; 2) Xây dựng mô hình phân lớp; 3) Đánh giá mô hình. Nội dung dưới đây sẽ trình bày cụ thể về mỗi pha.

### 3.1.2. Chuẩn bị dữ liệu cho quá trình phân lớp

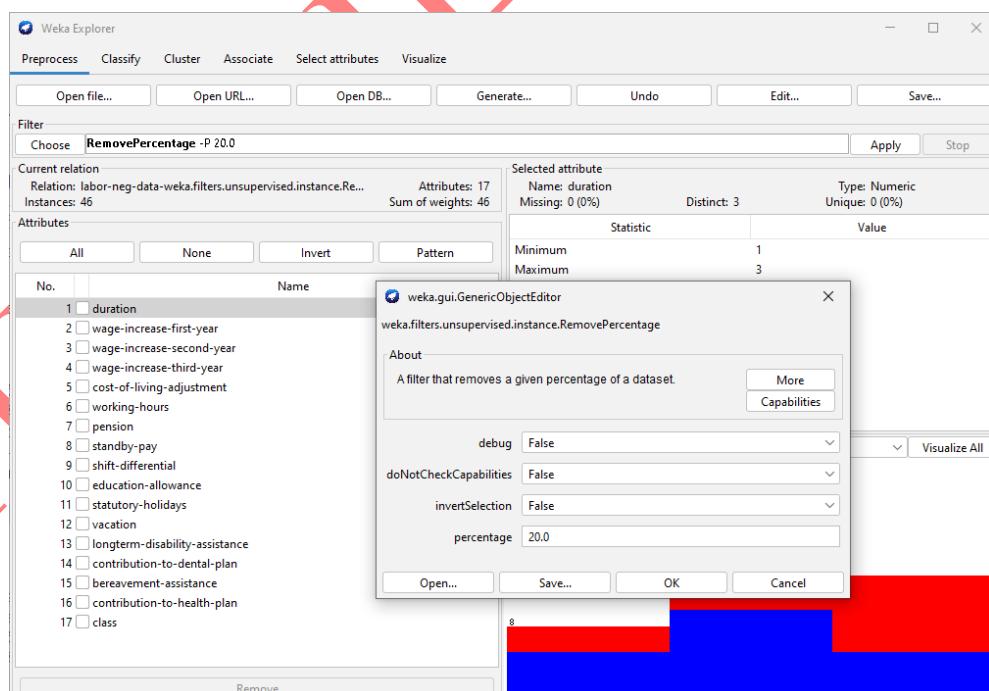
Để thực hiện việc xây dựng, thực nghiệm và ứng dụng mô hình phân lớp, chúng ta tiến hành chia tập dữ liệu ban đầu thành 3 tập dữ liệu: 1/ Tập dữ liệu huấn luyện (Training dataset) phục vụ cho quá trình học mô hình phân lớp; 2/ Tập dữ liệu kiểm nghiệm (Testing dataset) nhằm đánh giá hiệu quả dự đoán nhãn lớp dựa vào mô hình phân lớp đã học được trước đó; 3/Tập dữ liệu không có nhãn (Unlabel dataset) nhằm áp dụng mô hình phân lớp để đưa ra dự đoán nhãn lớp cho các đối tượng dữ liệu mới chưa biết bởi hệ thống.

Nội dung dưới đây sẽ minh họa quá trình thực hiện với Weka nhằm chuẩn bị dữ liệu cho phân lớp áp dụng lên tập dữ liệu labor.arff theo 2 cách: 1/ Áp dụng các bộ lọc trong Weka Explorer GUI để chia dữ liệu; 2/Sử dụng thư viện Weka để lập trình chia dữ liệu trong Java.

#### **- Áp dụng các bộ lọc trong Weka Explorer GUI để chia dữ liệu:**

Một trong hai bộ lọc thường được sử dụng cho mục đích này là RemovePercentage và Resample

- Sử dụng bộ lọc RemovePercentage: **Filter → unsupervised → instance → RemovePercentage**, thiết lập tham số percentage sẽ bị loại bỏ nhằm thu được tập training dataset (Ví dụ : percentage = 20 có nghĩa là 20% tập dữ liệu ban đầu sẽ bị loại bỏ để còn lại 80% dữ liệu vào tập training).

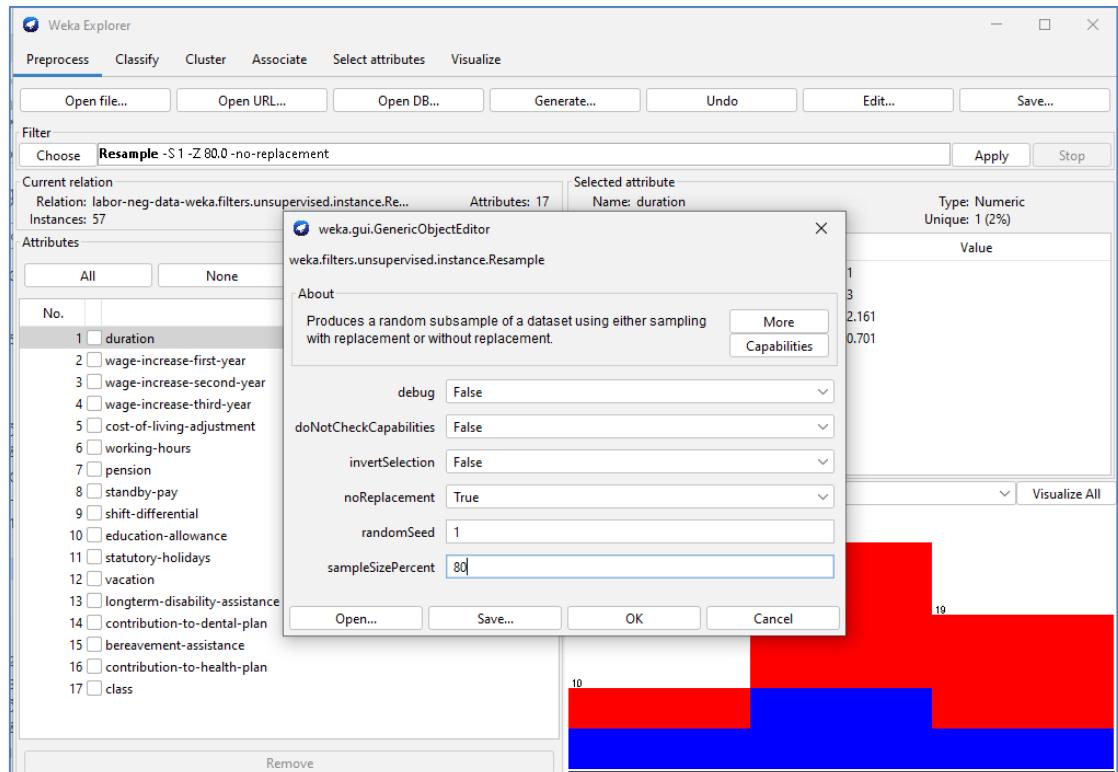


Chọn Apply và Save tập dữ liệu thu được (labor\_train.arff).

Chọn Undo để lấy lại tập dữ liệu gốc ban đầu, vẫn là với bộ lọc RemovePercentage, nhưng thiết lập tham số invertSelection = True nhằm

giữ lại 20% dữ liệu đó vào tập dữ liệu test. Chọn Apply và Save tập dữ liệu thu được (labor\_train.arff).

- Sử dụng bộ lọc Resample: **Filter → unsupervised → instance → Resample**, thiết lập tham số sampleSizePercent = 80 và noReplacement = True nhằm giữ lại 80% dữ liệu vào tập training.



Chọn **Apply** và **Save** tập dữ liệu thu được (labor\_train.arff).

Chọn **Undo** để lấy lại tập dữ liệu gốc ban đầu, vẫn là với bộ lọc Resample, nhưng thiết lập tham số invertSelection = True nhằm giữ lại 20% dữ liệu đó vào tập dữ liệu test. Chọn **Apply** và **Save** tập dữ liệu thu được (labor\_test.arff).

#### **- Sử dụng thư viện Weka để lập trình chia dữ liệu trong Java**

Lớp MyKnowledgeModel.java sử dụng một trong hai bộ lọc RemovePercentage và Resample thông qua Weka API như sau:

```

public class MyKnowledgeModel {
    DataSource source; //Luu nguon du lieu
    Instances dataset; //Luu giu du lieu dau vao

    ///////////
    Instances trainSet;
    Instances testSet;

    String[] model_options;//cac tham so cho mo hinh
    String[] data_options;//cac tham so xu ly du lieu

    public MyKnowledgeModel() {
    }

    public MyKnowledgeModel(String fileName) throws Exception {
        //1. Doc du lieu vao bo nho
        this.source = new DataSource(fileName);
        this.dataset = source.getDataSet();
    }

    //Tao train set, test set
    public Instances divideTrainTest(Instances originalSet, double percent, boolean isTest) throws Exception{
        RemovePercentage rp = new RemovePercentage();
        rp.setPercentage(percent);
        rp.setInvertSelection(isTest);
        rp.setInputFormat(originalSet);
        return rp.useFilter(originalSet, rp);
    }
}

```

Lớp WekaPro.java : Nạp dữ liệu đầu vào (Ví dụ với tập dữ liệu iris.arff và tiến hành tạo đối tượng của lớp MyKnowledgeModel và triệu gọi phương thức divideTrainTest() cho chia tập dữ liệu ban đầu thành 2 tập dữ liệu train và test.

```

public static void main(String[] args) throws Exception {
    // TODO code application logic here

    MyKnowledgeModel model = new MyKnowledgeModel("C:\\\\Program Files\\\\Weka-3-8-5\\\\data\\\\iris.arff");
    //System.out.println(model);

    /*System.out.println("-----");
    model.saveData("E:\\\\DATA LIEN STUDY TEACH\\\\DaoTao\\\\DaoTaoDaiHan\\\\Bai giang cac mon hoc\\\\KhaiPhaDu");
    model.saveData2CSV("E:\\\\DATA LIEN STUDY TEACH\\\\DaoTao\\\\DaoTaoDaiHan\\\\Bai giang cac mon hoc\\\\KhaiPhaDu");
    */

    System.out.println("before divide\\n");
    System.out.println(model.dataset.toSummaryString());

    System.out.println("\n Divide train & test set-----");
    model.trainSet = model.divideTrainTest(model.dataset, 20, false);
    model.testSet = model.divideTrainTest(model.dataset, 20, true);

    System.out.println("\n train set:");
    System.out.println(model.trainSet.toSummaryString());

    System.out.println("\n test set:");
    System.out.println(model.testSet.toSummaryString());
}

```

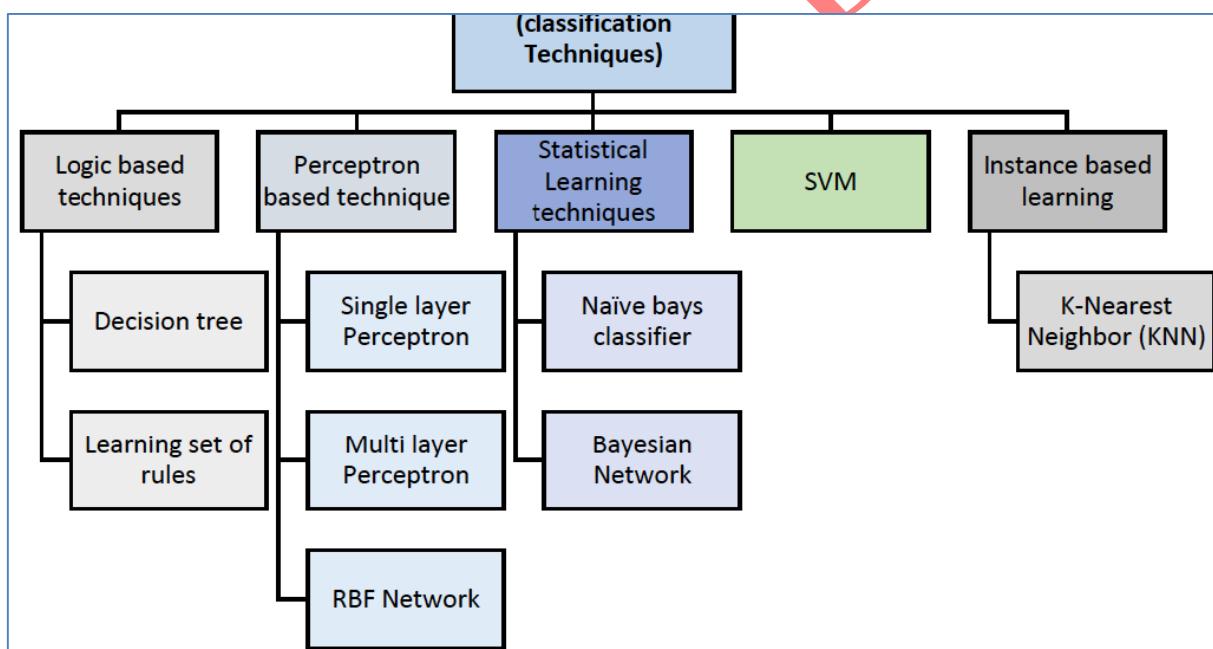
- Như đã đề cập ban đầu trong việc chuẩn bị dữ liệu sẽ gồm 3 tập : training dataset, testing dataset và unlabeled dataset. Quá trình thực hiện minh họa ở trên giúp tạo lập được 2 tập training dataset và testing dataset, để tạo lập tập unlabeled dataset ta tiến hành tách tiếp tập testing dataset ban đầu thành 2 phần: 1 phần làm testing dataset để phục vụ quá trình thực nghiệm mô hình và 1 phần unlabeled dataset. Việc này có thể thực hiện đơn giản bằng cách lấy đi một số mẫu dữ liệu từ testing

dataset sang unlabel dataset, tập unlabel dataset tạo được cần bõ nhãn (Đặt là dấu ?) để phục vụ cho quá trình dự đoán sau này. Hướng dẫn dưới đây giúp cho việc tiếp cận dễ dàng hơn, cụ thể như sau:

- Mở tập test.arff trong wordpad, mở thêm 1 wordpad trống. Copy dữ liệu từ tập test sang wordpad trống, lưu tên file là unlabel.arff.
- Xóa dữ liệu trong unlabel.arff để giữ lại 10 đối tượng dữ liệu.
- Xóa đi nhãn lớp trong tập unlabel.arff và đặt là dấu ?
- Trong tập test ban đầu : xóa đi 10 đối tượng dữ liệu mà đã đưa sang unlabel.arff.

### **3.1.3. Phương pháp phân lớp dữ liệu**

Theo 2 hướng tiếp cận học máy có giám sát và học máy bán giám sát, có rất nhiều các thuật toán đã được đưa ra để giải quyết bài toán phân lớp được phân loại như sau:



Hình 3.4. Một số phương pháp phân lớp dữ liệu

Trong nội dung này bài giảng sẽ trình bày về phương pháp phân lớp dữ liệu cơ bản là kNN. Trên cơ sở việc tiếp cận phương pháp cơ bản, sinh viên có thể mở rộng nghiên cứu và áp dụng đa dạng các phương pháp phân lớp dữ liệu khác.

#### **- Giới thiệu về phương pháp kNN (k-Nearest Neighbors)**

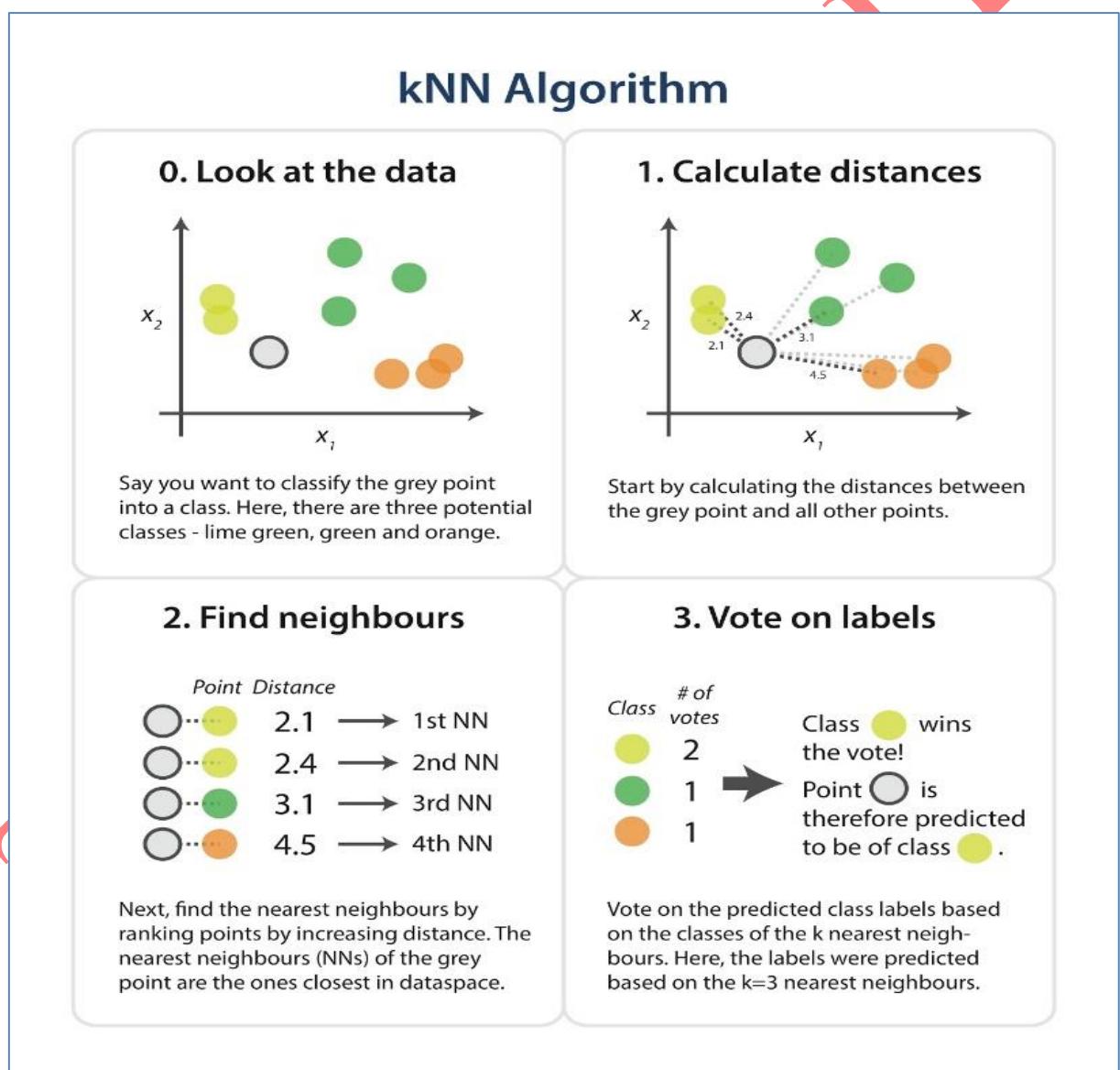
Phương pháp kNN được đề xuất từ những năm 1950. Phương pháp này dựa chủ yếu vào các phần tử lân cận trong tập dữ liệu huấn luyện. Nguyên tắc thực hiện như sau: Xét một bộ dữ liệu chưa được gán nhãn (mỗi đối tượng dữ liệu được xem như là 1 điểm trong không gian n chiều). Khi ấy, bộ phân lớp kNN sẽ tìm kiếm trong không gian những

điểm dữ liệu huấn luyện nào gần nhất với điểm dữ liệu hiện xét dựa trên phép đo khoảng cách Euclid truyền thống :

$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^n (x_{1i} - x_{2i})^2}$$

Nhãn lớp phổ biến trong số  $k$  điểm lân cận sẽ là nhãn lớp của điểm dữ liệu  $X$  cần dự đoán.

- **Minh họa thuật toán kNN:**



Hình 3.5. Minh họa thuật toán phân lớp kNN

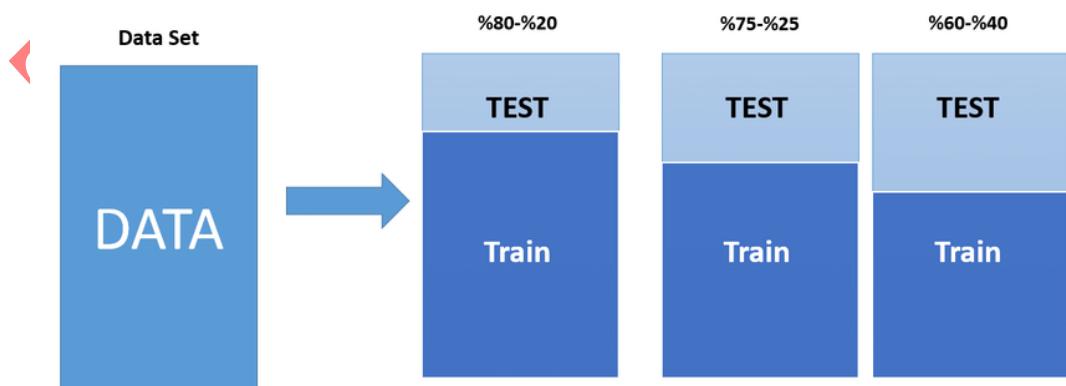
- **Hướng dẫn thực hành xây dựng mô hình phân lớp với thuật toán kNN bằng Weka Explorer GUI:**

Minh họa việc xây dựng mô hình phân lớp dữ liệu với thuật toán kNN cho tập dữ liệu labor.arff. Các bước thực hiện như sau:

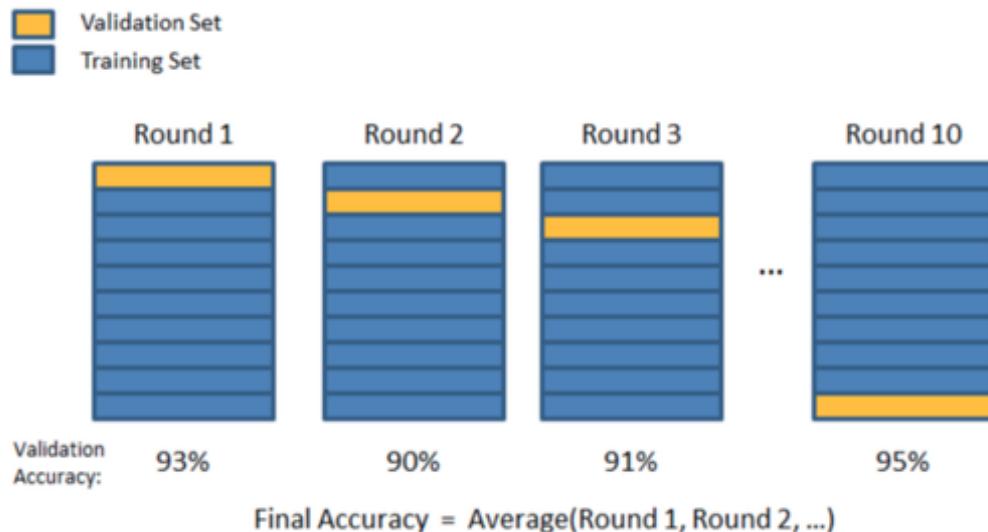
Bước 1. Tiến hành thực hiện chuẩn bị dữ liệu để tạo ra 3 tập dữ liệu labor\_train.arff, labor\_test.arff, labor\_unlabel.arff.

Bước 2. Huấn luyện tập dữ liệu labor\_train.arff bằng thuật toán KNN bằng cách trong Weka Explorer, tại tab Preprocess mở file labor\_train.arff để nạp tập dữ liệu cần huấn luyện. Tiếp đó chuyển tab Classify, chọn Classifier -> lazy -> iBk chính là lựa chọn thuật toán kNN để huấn luyện mô hình phân lớp từ tập labor\_train.arff.

Tại bước 2 này, quá trình xây dựng mô hình phân lớp sẽ cần chia tập dữ liệu training thành 2 phần là tập dữ liệu để xây dựng mô hình (training subset) và tập dữ liệu để đánh giá nhằm tối ưu hóa các tham số cho mô hình (validation set). Một số phương pháp chia dữ liệu được biết tới đó là : Phương pháp Holdout, Random sub-sampling, Cross-validation, Bootstrap. Trong đó: 1/ Phương pháp Holdout cho phép chia tập dữ liệu đầu vào thành hai tập phân biệt với tỉ lệ cho trước; 2/ Phương pháp Random subsampling là một biến thể của phương pháp Holdout với cơ chế hoạt động đó là lặp lại phương pháp Holdout k lần, độ chính xác dự đoán được tính là số trung bình của độ chính xác của mô hình được xây dựng cho mỗi lần lặp; 3/ Phương pháp Cross-validation có cơ chế hoạt động tương tự như Random subsampling nhưng chia tập dữ liệu thành k phần bằng nhau, trong đó 1 phần dùng cho validation set và (k-1) phần còn lại dùng cho training subset. Độ chính xác dự đoán được tính là trung bình của k lần lặp; 4/ Phương pháp lấy mẫu có hoàn lại Bootstrap có cơ chế hoạt động tương tự như Cross-validation chia dữ liệu thành k phần, nhưng tại mỗi lần lặp (Bootstrap) các mẫu dữ liệu này có thể được lặp lại nhiều lần để thực hiện huấn luyện xây dựng mô hình. Phương pháp này đặc biệt phù hợp với các tập dữ liệu có cỡ nhỏ.



Hình 3.6. Chia dữ liệu theo phương pháp Holdout

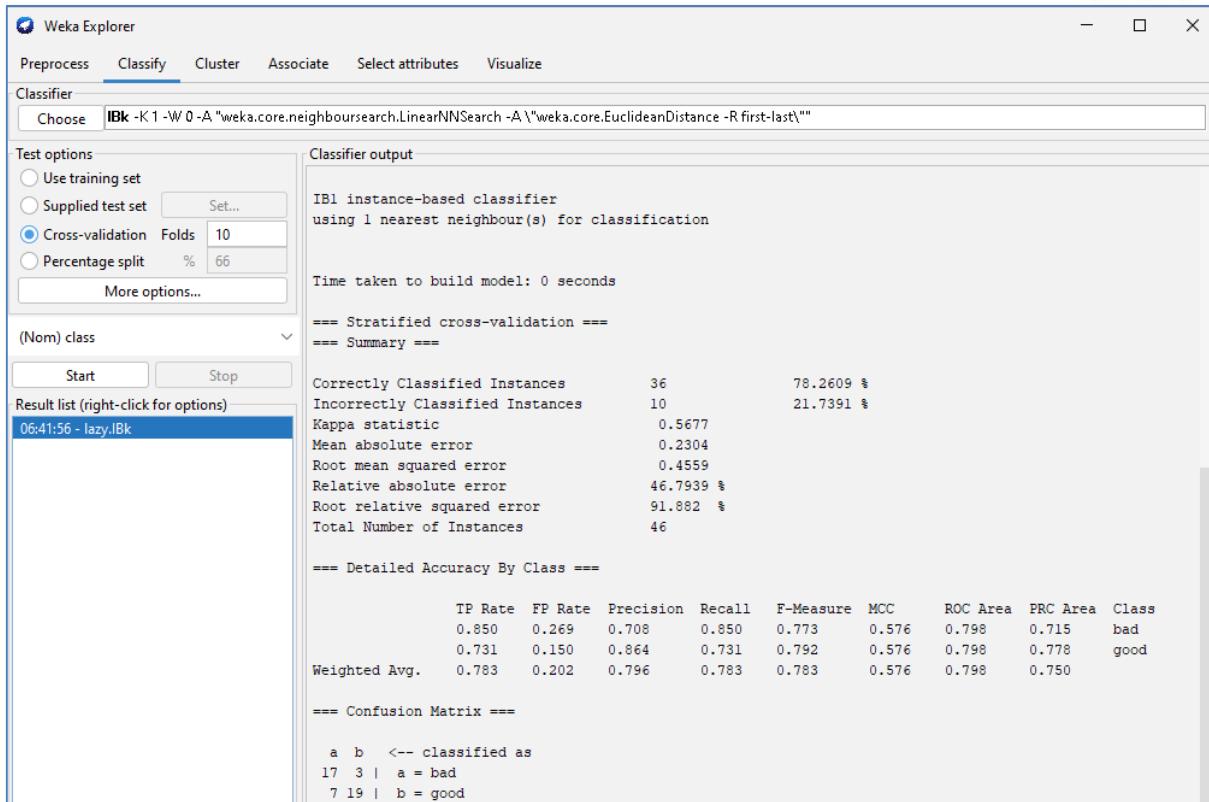


Hình 3.7. Chia dữ liệu theo phương pháp Cross-validation



Hình 3.8. Chia dữ liệu theo phương pháp Bootstrap

Thực hành quá trình chia tập dữ liệu thành 2 phần này trong Weka được thực hiện tại bước xây dựng mô hình. Weka hỗ trợ 2 phương pháp chính là Holdout và Cross-validation, thực hiện việc này bằng việc thiết lập giá trị tương ứng cho Percentage Split hoặc Cross-Validation.



Sau khi lựa chọn thuật toán phân loại và thiết lập tùy chọn chia dữ liệu, chọn Start để thực hiện quá trình huấn luyện xây dựng mô hình phân lớp dữ liệu. Kết quả phân tích cho mô hình phân lớp học được sẽ được hiển thị bên phần vùng Classifier output. Việc phân tích ngữ nghĩa của kết quả thực nghiệm này sẽ được bàn luận cụ thể trong mục tiếp theo của bài giảng (Mục 3.1.4).

Qua kết quả thực nghiệm này chúng ta có thể thử chọn những thuật toán phân lớp khác để so sánh hoặc thiết lập lại giá trị cho các tham số để đánh giá lựa chọn bộ tham số tốt nhất mang lại độ chính xác cao cho mô hình phân lớp. Việc tối ưu hóa tham số cho mô hình phân lớp cũng là một vấn đề được cộng đồng nghiên cứu rất quan tâm và đưa ra các phương pháp lựa chọn tham số khác nhau, một trong số đó có thể kể tới là CVParameterSelection, GridSearch, MultiSearch... Về mặt thực hành Weka cho phép cài đặt Plug-in là phần mềm Auto-weka để thực hiện các thuật toán tối ưu nhằm lựa chọn bộ giá trị phù hợp cho tham số của mô hình phân lớp (tunning model).

- **Hướng dẫn lập trình phân lớp dữ liệu với thuật toán kNN bằng Weka API trong Java**

Quá trình lập trình này cũng gồm các bước tương ứng với thao tác trên Weka Explorer GUI, đó là : 1/ Chia tập dữ liệu ban đầu thành 3 tập dữ liệu (train, test, unlabeled); 2/ Xây dựng mô hình phân lớp với thuật toán kNN, lưu lại mô hình; 3/ Sử dụng mô hình đã xây dựng được để dự đoán nhãn lớp cho các đối tượng (Các đối tượng dữ liệu trong tập unlabeled), lưu kết quả dự đoán ra file.

Nội dung dưới đây minh họa quá trình thực hiện với Java áp dụng cho tập dữ liệu về hoa diên vĩ iris.arff.

Lớp kNNModel:

```

public class kNNModel extends MyKnowledgeModel{
    IBk knn;
    Evaluation eval;

    public kNNModel(String fileName, String m_opts, String d_opts) throws Exception {
        super(fileName, m_opts, d_opts);
    }

    //build knn model
    public void buildKNN(String fileName) throws Exception{
        //Doc train set vao bo nho
        setTrainSet(fileName);

        //thiet lap va chi ro truong dong vai tro nhan lop
        this.trainSet.setClassIndex(this.trainSet.numAttributes()-1);

        //Huan luyen mo hinh kNN
        this.knn = new IBk(); //Khởi tạo mô hình
        knn.setOptions(model_options); //Khởi tạo mô hình, đưa option của mô hình vào
        knn.buildClassifier(this.trainSet); //Tien hanh xay dung mo hinh
    }

    public void evaluateKNN(String fileName) throws Exception{
        setTestSet(fileName); //Doc test set vao bo nho
        this.testSet.setClassIndex(this.testSet.numAttributes()-1);

        //Danh gia mo hinh bang 10-fold cross-validation
        Random rd = new Random();
        int folds = 10;
        eval = new Evaluation(this.trainSet);
        eval.crossValidateModel(knn, this.testSet, folds, rd);
        System.out.println(eval.toSummaryString("\n Ket qua danh gia mo hinh kNN----\n", false));
    }
}

```

Phương thức buildKNN nhằm xây dựng mô hình phân lớp sử dụng thuật toán kNN, sẽ gồm một số công việc chính: 1) Đọc dữ liệu từ file vào bộ nhớ; 2) Thiết lập trường đóng vai trò nhán lớp; 3) Khởi tạo mô hình IBk; 4) Thiết lập giá trị tham số bằng cách cung cấp option cho mô hình; 5) Tiến hành xây dựng mô hình.

Phương thức evaluateKNN nhằm đánh giá mô hình phân lớp xây dựng được trên tập dữ liệu test. Trong trường hợp này phương pháp Cross-validation được sử dụng để phân chia tập dữ liệu training subset và validation set với fold = 10.

Cũng trong lớp kNNModel, tiến hành cài đặt phương thức predictClassLabel cho mục đích dự đoán nhán lớp với dữ liệu đầu vào fileIn là tập dữ liệu unlabeled.arff và kết quả dữ liệu dự đoán được đầu ra sẽ được lưu trong fileOut.

```

public void predictClassLabel(String fileIn, String fileOut) throws Exception{
    //Doc du lieu can du doan vao bo nho : file unLabel
    DataSource source = new DataSource(fileIn);
    Instances unLabel = source.getDataSet();
    unLabel.setClassIndex(unLabel.numAttributes()-1);

    //Du doan classLabel cho tung instances
    for (int i = 0; i < unLabel.numInstances()-1; i++) {
        double predict = knn.classifyInstance(unLabel.instance(i));
        unLabel.instance(i).setClassValue(predict);
    }

    //Xuat ket qua ra fileOut
    BufferedWriter outBuff = new BufferedWriter(new FileWriter(fileOut));
    outBuff.write(unLabel.toString());
    outBuff.newLine();
    outBuff.flush();
    outBuff.close();
}

```

Lớp WekaPro.java, trong phương thức main sẽ khởi tạo tham số và gọi thực thi các phương thức buildKNN, evaluteKNN, predictClassLabel tương ứng với quá trình học mô hình phân lớp, đánh giá mô hình học được và sử dụng mô hình học được cho việc dự đoán nhãn lớp.

```

System.out.println("\n ----- \n KNN \n");
kNNModel m_knn = new kNNModel("", "-K 1 -W 0 -A \"weka.core.neighboursearch.LinearNNSearch -A \\\"weka.core.EuclideanDistance -R first-
m_knn.buildKNN("F:\\iris_train.arff");

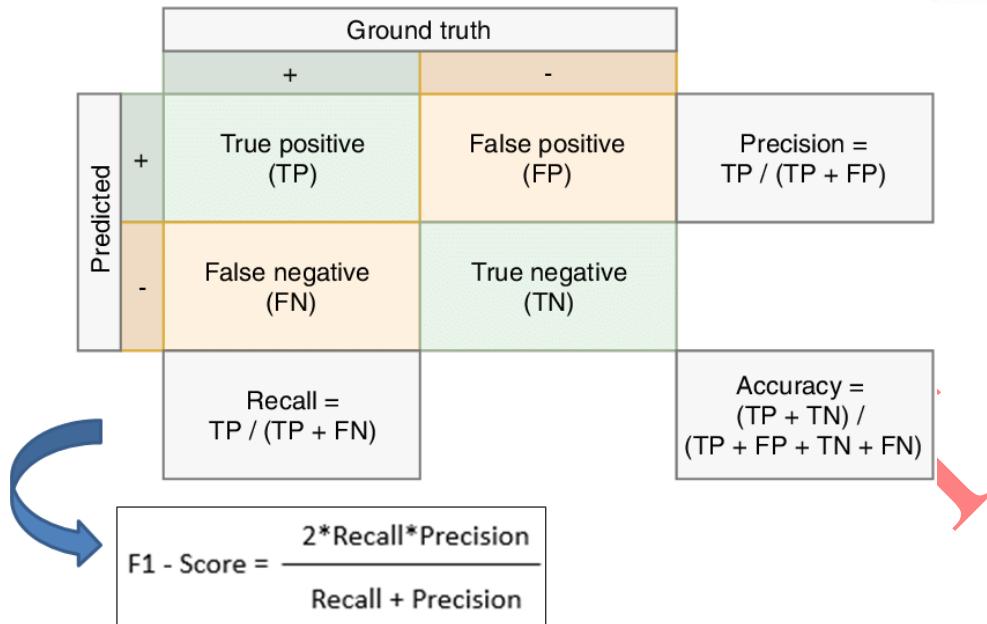
m_knn.evaluteKNN("F:\\iris_test.arff");

m_knn.predictClassLabel("F:\\iris_unlabel.arff",
                        "F:\\iris_predict_knn.arff");
System.out.println(m_knn);

```

### 3.1.4. Đánh giá mô hình phân lớp dữ liệu

Một số độ đo phổ biến thường được sử dụng để đánh giá độ chính xác của mô hình phân lớp là : 1/ Accuracy; 2/ Precision, Recall, F-measure; 3/ Đường cong ROC. Trong đó độ đo Accuracy, Precision, Recall, F-measure được tính toán dựa vào ma trận nhầm lẫn (Confusion matrix) sau:



Ví dụ với bài toán dự đoán bệnh nhân bị ung thư. Giả sử ta cần dự đoán kết quả xét nghiệm của 1005 bệnh nhân xem họ có bị ung thư hay không. Dưới đây là những gì mô hình của chúng ta dự đoán:

- 90 bệnh nhân bị ung thư và tất cả dự đoán này của chúng ta đều đúng.
- 915 bệnh nhân không bị ung thư nhưng thật ra có tới 910 người lại bị trong thực tế.

Từ số liệu ghi nhận trên, ta tiến hành xây dựng ma trận nhầm lẫn sau:

		Thực tế (có)	Thực tế (không)
Dự đoán (có)	90 (True Positive)	0 (False Positive)	
Dự đoán (không)	910 (False Negative)	5 (True Negative)	

Giải thích trong bảng trên, có 4 thuật ngữ ta cần để ý đến:

- **True Positive (TP):** những bệnh nhân ta đoán là có bệnh đúng là đang mang bệnh.
- **True Negative (TN):** những bệnh nhân ta đoán là không có bệnh đúng là đang khỏe mạnh.
- **False Positive (FP):** những bệnh nhân ta đoán là có bệnh thật ra đang khỏe mạnh.
- **False Negative (FN):** những bệnh nhân ta đoán là không có bệnh thật ra đang mang bệnh.

Tính toán các độ đo chính xác từ ma trận nhầm lẫn ta được:

Accuracy cho biết tỷ lệ dự đoán chính xác đối với cả bệnh nhân mắc bệnh và không mắc bệnh trên toàn bộ bệnh nhân:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} = \frac{90 + 5}{90 + 0 + 910 + 5} = 9.45\%$$

Precision cho biết tỷ lệ giữa người dự đoán thật sự có bệnh so với tất cả các ca được dự đoán là có bệnh:

$$Precision = \frac{TP}{TP + FP} = \frac{90}{90 + 0} = 100\%$$

Recall cho biết tỷ lệ giữa người dự đoán thật sự có bệnh so với tất cả các ca thực tế có bệnh:

$$Recall = \frac{TP}{TP + FN} = \frac{90}{90 + 910} = 9\%$$

Để cân bằng giữa 2 độ đo Precision và Recall, ta tiến hành tính F-measure:

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} = 0.165$$

Như vậy có một số nhận xét đưa ra như sau:

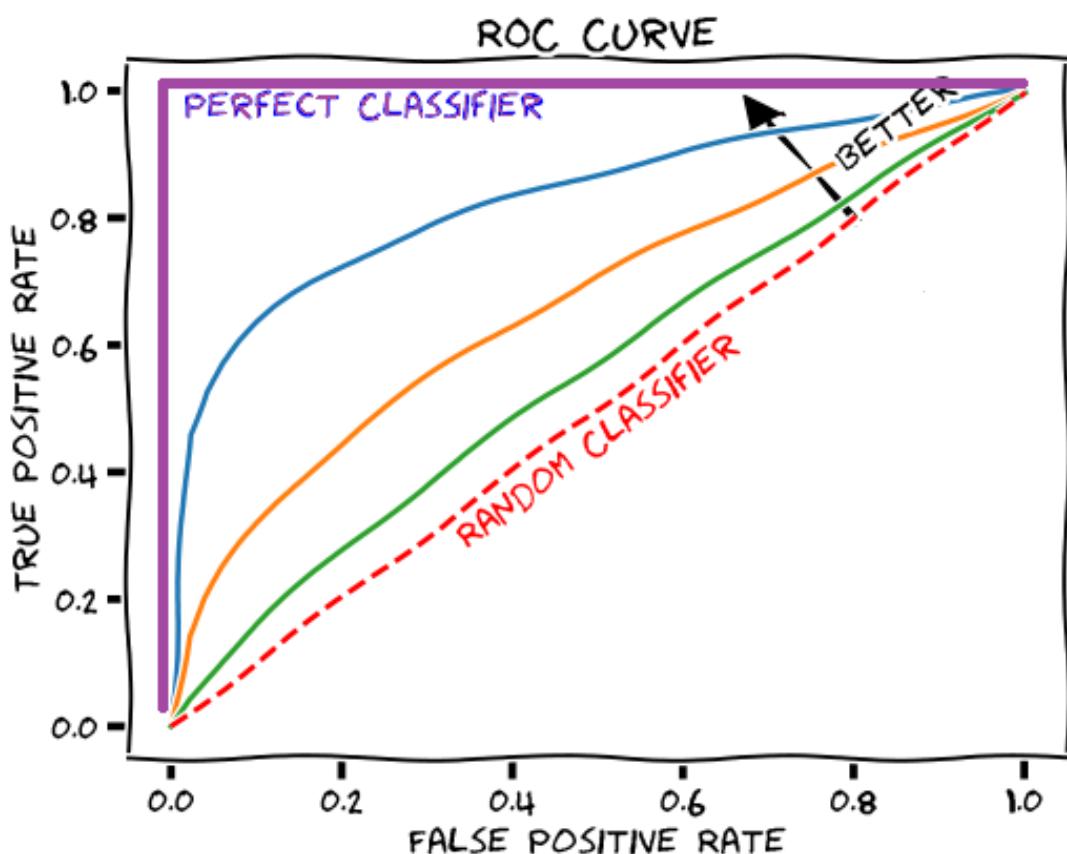
- Giá trị Precision ở ví dụ này cho thấy việc dự đoán 90 người có bệnh và trong thực tế những người này đúng là đang bị bệnh thật. Vậy ra, 100% số người ta dự đoán có bệnh là chính xác.
- Giá trị Recall cho biết trong những người thực sự có bệnh thì mô hình phân lớp trong trường hợp này chỉ dự đoán được 90 người có bệnh, mà thực tế có tới 1000 người trong thực tế mắc bệnh. Vậy ra, mô hình phân lớp của chúng ta chỉ có thể dự đoán được 9% số lượng người có bệnh trong thực tế.
- Rõ ràng Precision cao, Recall thấp trong trường hợp này sẽ dẫn tới rất nhiều bệnh nhân bị bệnh mà không được điều trị. Mô hình phân lớp nhận được có lẽ không phù hợp cho bài toán này.
- Giả sử chúng ta chọn được mô hình phân lớp cho bài toán dự đoán bệnh nhân mắc bệnh ung thư với ma trận nhầm lẫn sau:

	Thực tế (có)	Thực tế (không)	
Dự đoán (có)	90	910	Precision = 9%
Dự đoán (không)	10	5	
Recall = 90%			

Rõ ràng trong trường hợp này Precision rất nhỏ nếu đem so với Recall (9% so với 90%), điều này cho ta biết mô hình đã dự đoán sai quá nhiều người lành thành người bệnh. Tuy nhiên có vẻ như dự đoán sai này ít tác hại hơn là trường hợp trước đó. 90% trường hợp này có thể bị hóa trị nhầm nhưng ít ra là có thể họ vẫn sống, trong khi ở trường hợp Precision cao, Recall thấp, số lượng người không được điều trị quá cao và cầm chắc cái chết sớm.

Như vậy tùy từng bài toán khác nhau mà chúng ta cần phân tích để lựa chọn mô hình phân lớp với giá trị Precision, Recall ở ngưỡng chấp nhận được hoặc loại bỏ.

Ngoài việc sử dụng các độ đo trên để phân tích đánh giá mô hình phân lớp thì đường cong ROC (Receiver Operating Characteristic) là một công cụ khác để so sánh hiệu năng giữa hai hay nhiều mô hình khác nhau một cách trực quan thông qua biểu đồ. Đường ROC của mô hình nào nằm trên cùng sẽ biểu thị hiệu năng của mô hình đó cao hơn các mô hình còn lại.



Hình 3.9. Minh họa so sánh độ chính xác của các mô hình phân lớp thông qua đường cong ROC

Nội dung dưới đây sẽ tiến hành thực thi đánh giá mô hình phân lớp theo các độ đo đề cập ở trên trong Weka:

- Vẽ đường cong ROC của mô hình phân lớp bằng Weka Explorer: Với mô hình phân lớp học được, click chuột phải vào mô hình đã xây dựng trước đó trong Result list và chọn **Visualize threshold curve**, chọn ROC trên biến nhãn lớp.
- Vẽ đường cong Precision, Recall của mô hình phân lớp: Cách thao tác giống với vẽ đường cong ROC, nhưng chọn trực hoành x là recall, trực tung y là precision.
- So sánh mô hình phân lớp với chức năng Experimenter trong Weka: Lần lượt thao tác qua các tab, cụ thể tại tab Setup chọn mở tập dữ liệu training dataset, thêm một số mô hình phân lớp (Ví dụ: Ibk, Native Bayes, ANN...) trong mục Algorithms để thiết lập các thuật toán được sử dụng để xây dựng các mô hình phân lớp khác nhau. Chuyển tiếp sang tab Run để thực hiện huấn luyện đồng thời các mô hình phân lớp. Ké tiếp là tab Analyse, tại đây click chọn Experiment nhằm thiết lập một số tham số (Chú ý test base chọn lấy mô hình là cơ sở để cho các mô hình khác so sánh), click Perform test để thực hiện kiểm nghiệm các mô hình phân lớp học được, kết quả sẽ hiển thị bộ phân lớp gắn chữ v là chiến thắng, \* là thua và không gắn gì là mang nghĩa trung gian.

### **3.1.5. Một số ứng dụng của bài toán phân lớp:**

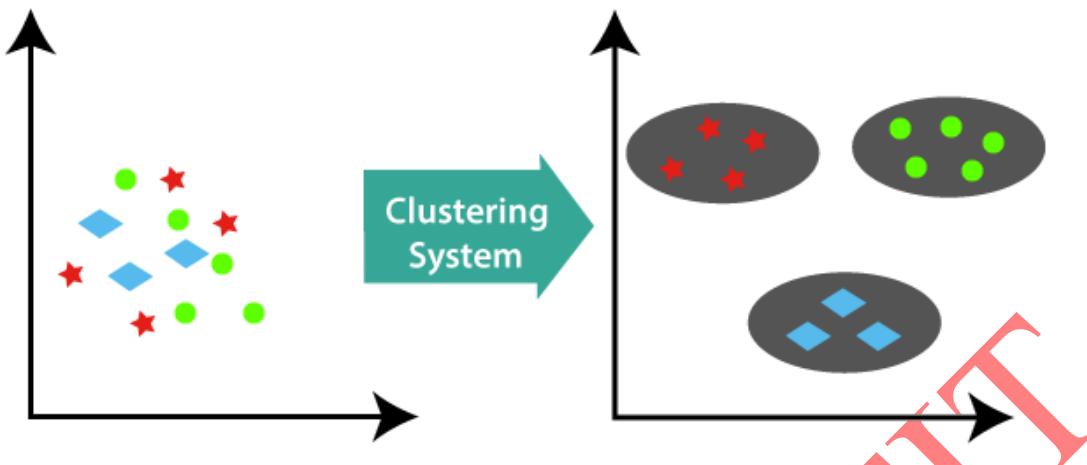
- Phân lớp/ nhận dạng chữ viết tay, ảnh, audio, video.
- Dự đoán đánh giá cho bài toán tư vấn sản phẩm đa phương tiện (vd: phim ảnh, bài hát, sách báo...)
- Phân loại hồ sơ tín dụng là an toàn hay rủi ro
- Lựa chọn phương thức điều trị A, B hay C cho bệnh nhân
- Phân loại một e-mail là spam hay không
- Dự đoán thời tiết
- ...

## **3.2. Phân cụm dữ liệu**

### **3.2.1. Bài toán phân cụm dữ liệu**

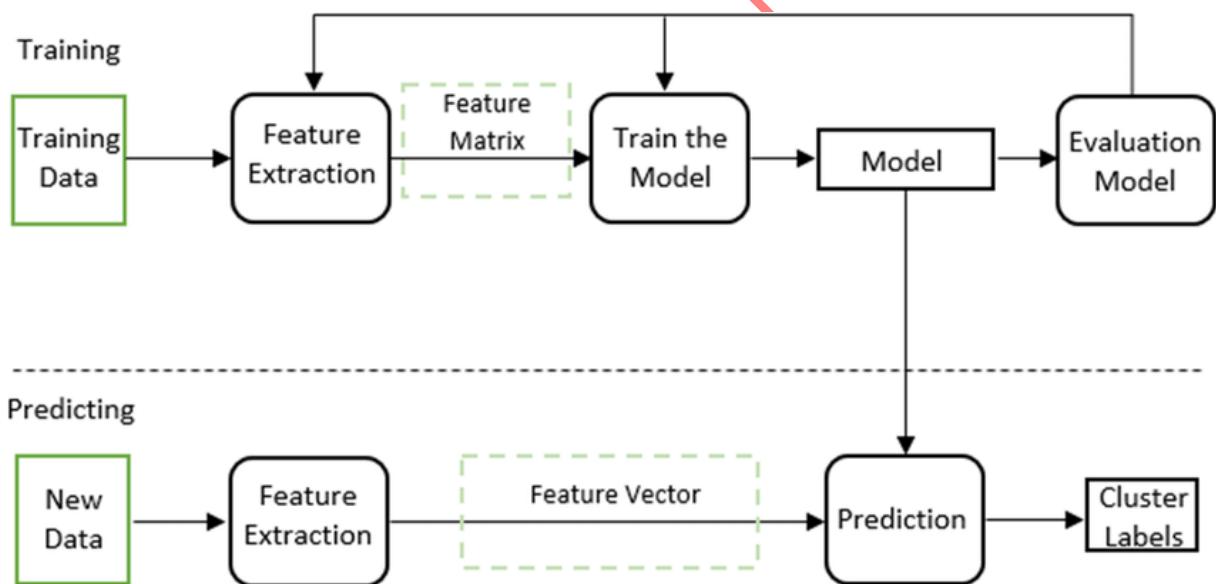
Phân cụm dữ liệu là bài toán gom nhóm các đối tượng dữ liệu vào thành từng cụm (cluster) sao cho các đối tượng trong cùng một cụm có sự tương đồng theo một tiêu chí nào đó. Các đối tượng dữ liệu đầu vào cho bài toán phân cụm thường không được gán nhãn trước đó.

Quá trình phân cụm dữ liệu được minh họa theo hình dưới đây.



Hình 3.10. Minh họa bài toán phân cụm dữ liệu

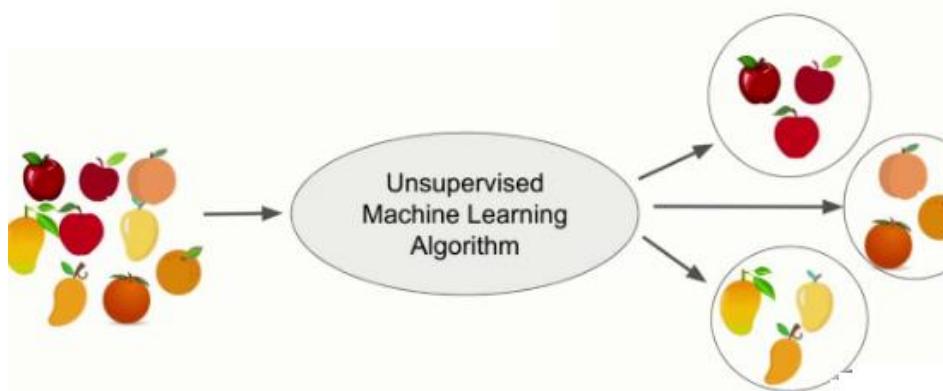
Trong các cách tiếp cận học máy thì cách tiếp cận học không giám sát là phù hợp để giải quyết bài toán phân cụm, với đầu vào là một tập hợp các đối tượng dữ liệu không có nhãn lớp cho trước. Quá trình học không giám sát được miêu tả theo luồng sau:



Hình 3.11. Quá trình phân cụm dữ liệu theo hướng tiếp cận học máy không giám sát

Biểu diễn dưới dạng toán học, quá trình huấn luyện mô hình phân cụm sẽ học cách xác định hàm  $y = f(x)$  từ tập dữ liệu huấn luyện gồm  $\{x_1, x_2, \dots, x_N\}$ . Các dữ liệu trong tập dữ liệu dùng để huấn luyện không có nhãn. Các thuật toán phân cụm dựa trên tập dữ liệu chính là cách xác định cấu trúc ẩn trong tập dữ liệu đó.

Ví dụ phân cụm ảnh hoa quả:



Trọng tâm chính của thuật toán phân cụm là tính độ đo tương đồng giữa các đối tượng dữ liệu. Thông thường để đo độ tương đồng giữa các đối tượng dữ liệu thì mỗi đối tượng dữ liệu sẽ được biểu diễn bởi 1 vector đặc trưng, khi đó độ tương đồng giữa các đối tượng dữ liệu căn cứ trên độ tương tự/ tương quan/ độ đo khoảng cách giữa các điểm trong không gian có n đặc trưng đó. Một số độ đo như khoảng cách Euclid, Manhattan, Cosine, Jaccard... thường được sử dụng cho mục đích này.

- Khoảng cách Euclidean:

$$d(i, j) = \sqrt{(x_{i1} - x_{j1})^2 + \dots + (x_{in} - x_{jn})^2} \quad (3.1)$$

- Khoảng cách Manhattan

$$d(i, j) = |x_{i1} - x_{j1}| + \dots + |x_{in} - x_{jn}| \quad (3.2)$$

- Khoảng cách Minkowski

$$d(i, j) = \sqrt[p]{|x_{i1} - x_{jp}|^p + \dots + |x_{in} - x_{jn}|^p} \quad (3.3)$$

- Hệ số Jaccard (được sử dụng khi giá trị các thuộc tính được thể hiện ở dạng nhị phân)

$$J(x, y) = \frac{\sum_i \min(x_i, y_i)}{\sum_i \max(x_i, y_i)} \quad (3.4)$$

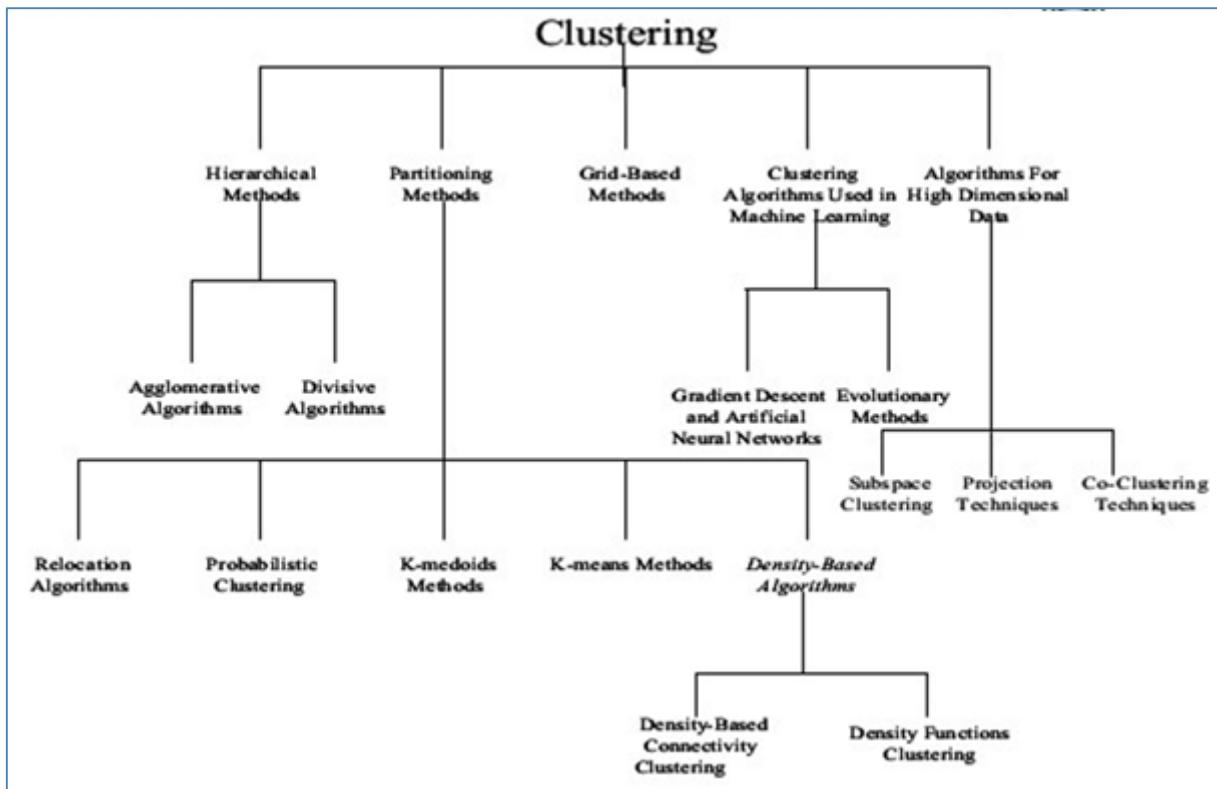
- Khoảng cách đối với các giá trị thuộc dạng danh mục

$$d(i, j) = \frac{p - m}{p}, \quad (3.5)$$

với  $p$  là tổng số các biến và  $m$  là tổng số trùng

### 3.2.2. Phương pháp phân cụm dữ liệu

Có rất nhiều các thuật toán đã được đưa ra để giải quyết bài toán phân cụm được phân loại như sau:



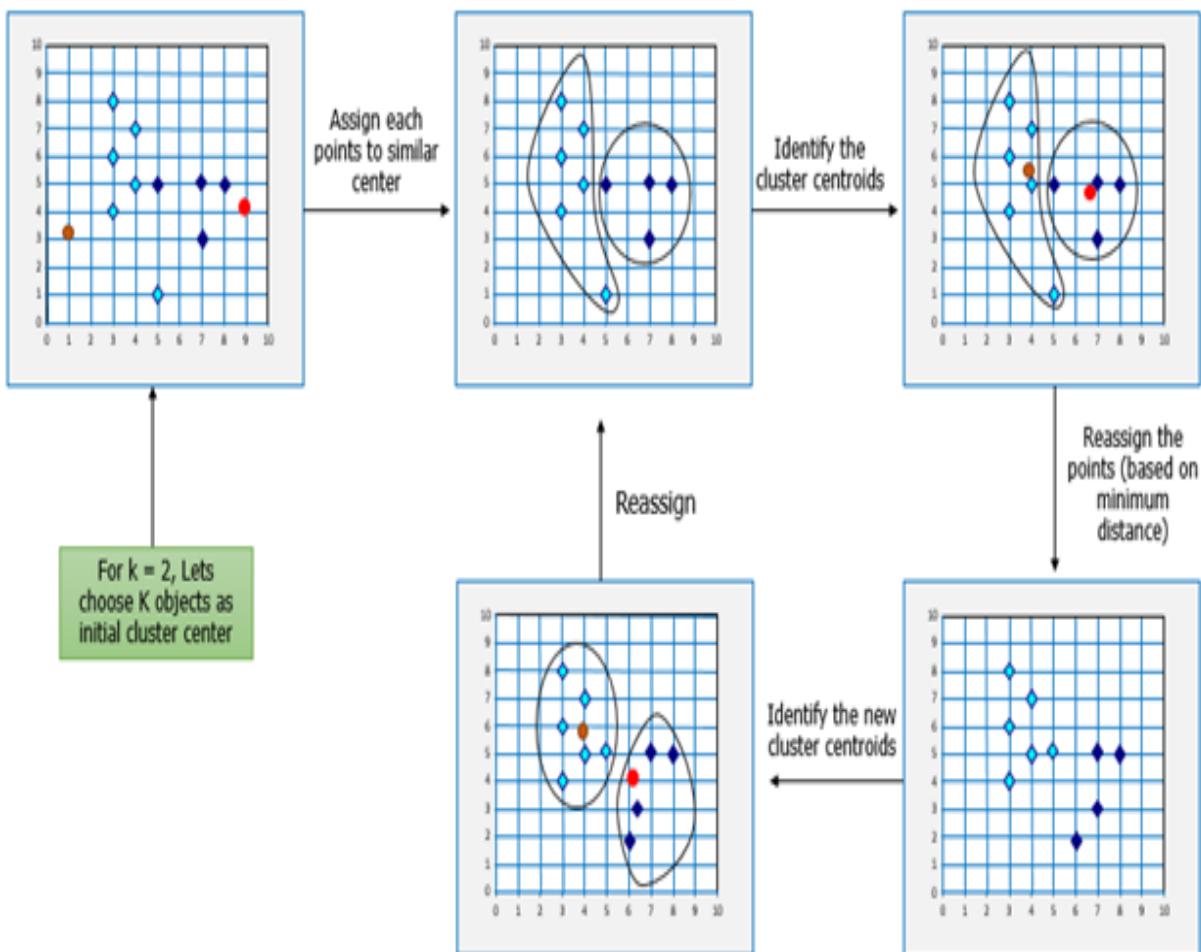
Hình 3.12. Một số phương pháp phân cụm dữ liệu

Trong nội dung này bài giảng sẽ trình bày về phương pháp phân cụm dữ liệu cơ bản là k-means. Trên cơ sở việc tiếp cận phương pháp cơ bản, sinh viên có thể mở rộng nghiên cứu và áp dụng đa dạng các phương pháp phân cụm dữ liệu khác.

### - *Giới thiệu về phương pháp k-means*

Thuật toán phân cụm K-means được giới thiệu năm 1957 bởi Lloyd K-means và là phương pháp phổ biến nhất cho việc phân cụm, dựa trên việc phân vùng dữ liệu. Biểu diễn tập dữ liệu:  $D = \{x_1, x_2, \dots, x_r\}$ , với  $x_i$  là đối tượng dữ liệu thứ i, được biểu diễn bởi vector đặc trưng có n chiều trong không gian Euclidean. K-means phân cụm D thành K cụm dữ liệu (K là một hằng số cho trước). Trong đó mỗi cụm dữ liệu có một điểm trung tâm gọi là centroid.

### - *Minh họa thuật toán k-means:*



Hình 3.13. Minh họa thuật toán phân cụm k-means

Thuật toán k-means như sau:

**Đầu vào:** Cho tập dữ liệu  $D$ , với  $K$  là số cụm, phép đo khoảng cách giữa 2 điểm dữ liệu là  $d(x,y)$

**Khởi tạo:** Khởi tạo  $K$  điểm dữ liệu trong  $D$  làm các điểm trung tâm (centroid)

**Lặp lại các bước sau đến khi hội tụ:**

- Bước 1: Với mỗi điểm dữ liệu, gán điểm dữ liệu đó vào cluster có khoảng cách đến điểm trung tâm của cluster là nhỏ nhất.
- Bước 2: Với mỗi cluster, xác định lại điểm trung tâm của tất cả các điểm dữ liệu được gán vào cluster đó.

Điều kiện hội tụ (điều kiện dừng thuật toán) được xác định theo một số cách như sau:

- Tại 1 vòng lặp có ít các điểm dữ liệu được gán sang cluster khác

- Điểm trung tâm (centroid) không thay đổi nhiều. Điểm trung tâm  $m_i$  của cluster  $C_i$  được xác định như sau:

$$\mathbf{m}_i = \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{x} \quad (3.6)$$

- Giá trị hàm mất mát không thay đổi nhiều. Giá trị hàm mất mát được xác định như sau:

$$Error = \sum_{i=1}^k \sum_{\mathbf{x} \in C_i} d(\mathbf{x}, \mathbf{m}_i)^2 \quad (3.7)$$

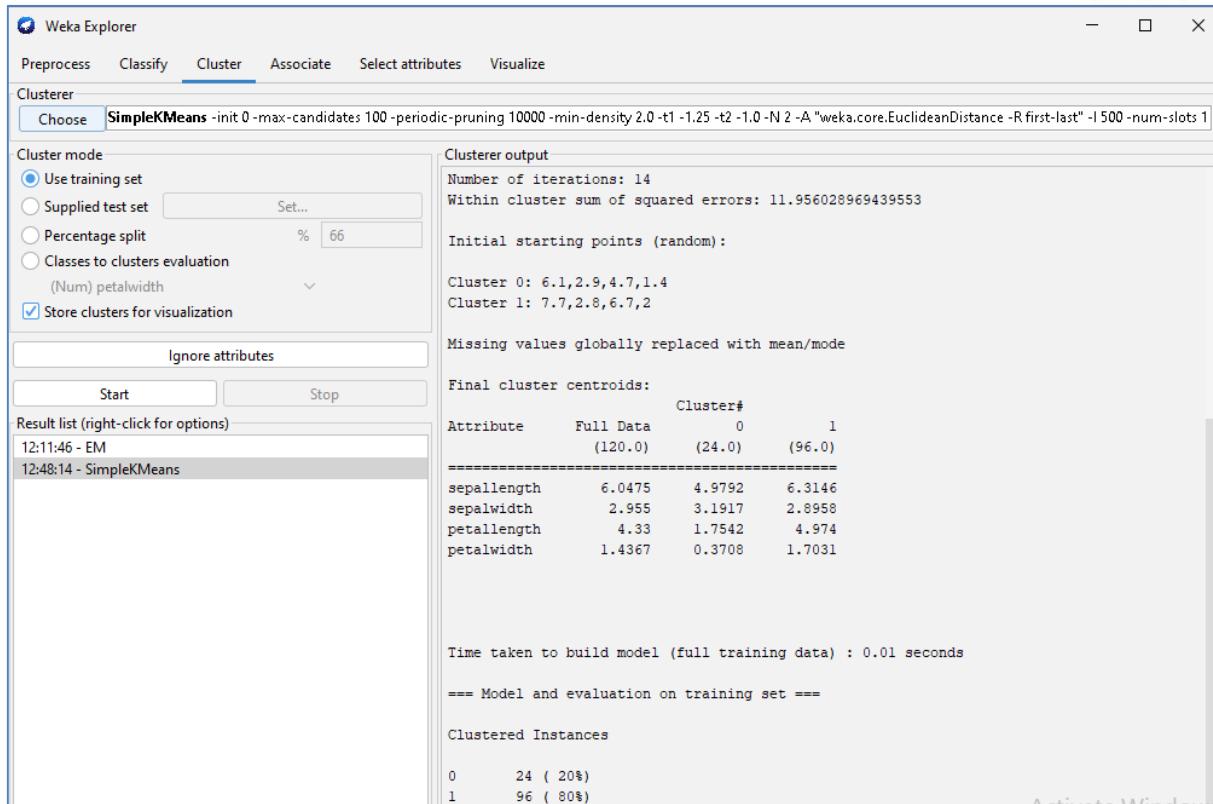
- **Hướng dẫn thực hành xây dựng mô hình phân cụm với thuật toán k-means bằng Weka Explorer GUI:**

Minh họa việc xây dựng mô hình phân cụm dữ liệu với thuật toán k-means cho tập dữ liệu iris.arff trong Weka Explorer GUI. Các bước thực hiện như sau:

Bước 1: Chọn tab Preprocess, ở tập dữ liệu iris.arff, Remove thuộc tính class nhằm đảm bảo tập dữ liệu đầu vào chưa được gán nhãn.

Bước 2: Tiến hành thực hiện chuẩn bị dữ liệu để tạo ra 2 tập dữ liệu training dataset, testing dataset (Tỷ lệ 80%:20%). Trong đó tập dữ liệu training dataset để huấn luyện mô hình phân cụm và tập dữ liệu testing dataset sẽ dùng để dự đoán cụm cho các đối tượng dữ liệu mới.

Bước 3: Nạp tập dữ liệu training dataset, chuyển tab Cluster chọn Clusterer -> SimpleKMeans, thiết lập numClusters = 2, distanceFunction = EuclideanDistance. Cluster mode chọn *Use training set* và Start để tiến hành huấn luyện mô hình phân cụm. Kết quả được thể hiện trong Clusterer output như sau:



Bước 4: Thực hiện áp dụng mô hình phân cụm đã học nhằm dự đoán cụm cho đối tượng dữ liệu mới trong tập dữ liệu testing data: Chọn Supplied test set với tập dữ liệu đầu vào là testing dataset. Tiếp đó, click chuột phải vào SimpleKMeans trong phân vùng Result list, chọn **Re-evaluate model on current test set**. Kết quả sẽ được thể hiện trong phân vùng Clusterer output.

**- Hướng dẫn lập trình phân cụm dữ liệu với thuật toán k-means bằng Weka API trong Java**

Quá trình lập trình này cũng gồm các bước tương ứng với thao tác trên Weka Explorer GUI, đó là : 1/ Chia tập dữ liệu ban đầu thành 2 tập dữ liệu (train, unlabeled); 2/ Xây dựng mô hình phân cụm với thuật toán k-means, lưu lại mô hình; 3/ Sử dụng mô hình đã xây dựng để dự đoán phân cụm cho các đối tượng (Các đối tượng dữ liệu trong tập unlabeled), lưu kết quả dự đoán ra file.

Nội dung dưới đây minh họa quá trình thực hiện với Java áp dụng cho tập dữ liệu về hoa diên vĩ iris.arff.

Lớp kMeanModel.java

```

public class kMeanModel extends MyKnowledgeModel{
    SimpleKMeans kmeans;

    public kMeanModel(String filename) throws Exception{
        super(filename);
    }

    public void buildKMeansModel(String filename) throws Exception{ //train dataset
        setTrainSet(filename);

        //thiet lap mo hinh kmeans
        kmeans = new SimpleKMeans();
        kmeans.setNumClusters(2);
        kmeans.setDistanceFunction(new EuclideanDistance());
        kmeans.buildClusterer(trainSet);
        System.out.println(kmeans);
    }

    public void predictCluster(String filename) throws Exception{//Unlabel dataset
        if(!filename.isEmpty()){
            DataSource ds = new DataSource(filename);
            Instances unlabeled = ds.getDataSet();
            for(int i=0; i< unlabeled.numInstances();i++){
                double predict = kmeans.clusterInstance(unlabeled.instance(i));
                System.out.println("Instance"+i+" thuoc cluster :" +predict);
            }
        }
    }
}

```

Lớp WekaPro.java, trong phương thức main sẽ khởi tạo tham số và gọi thực thi các phương thức buildKMeansModel, predictCluster tương ứng với quá trình học mô hình phân cụm và sử dụng mô hình học được cho việc dự đoán cụm cho đối tượng dữ liệu mới.

```

kMeanModel model = new kMeanModel("");
model.buildKMeansModel("E:\\iris_train_2.arff");
model.predictCluster("E:\\iris_unlabel_2.arff");

```

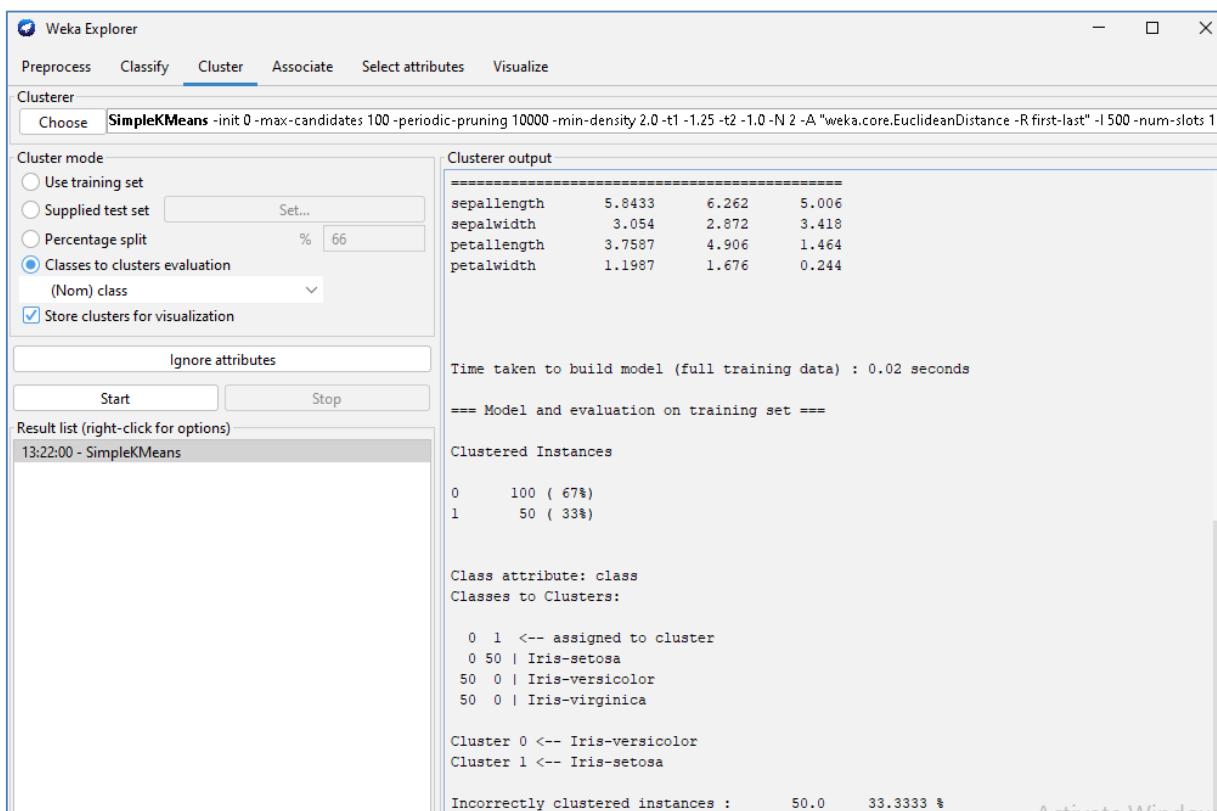
### 3.2.3. Đánh giá mô hình phân cụm dữ liệu

Để đánh giá chất lượng mô hình phân cụm ta có thể đánh giá thông qua một số phương pháp như sau:

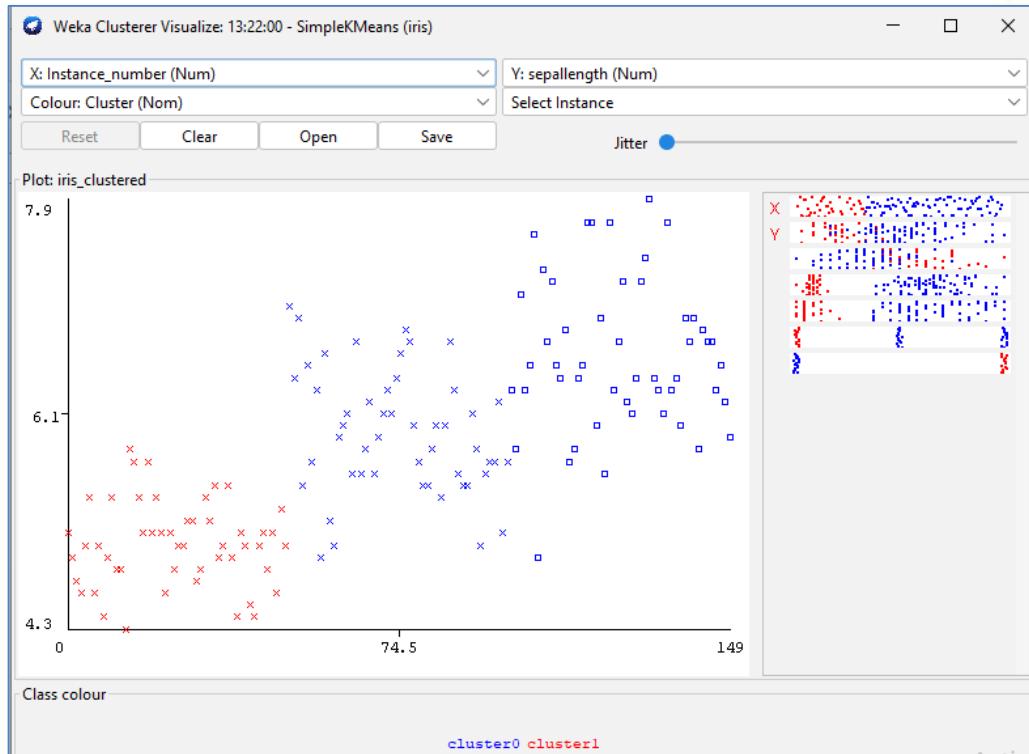
- Giữa các cụm phải được tách biệt nhau hoàn toàn và sự khác biệt / khoảng cách giữa 2 cụm phải đủ lớn để phân biệt 2 cụm với nhau.
- Chênh lệch giữa các điểm dữ liệu bên trong một cụm phải nhỏ. Chênh lệch ở đây thể hiện sự khác biệt với nhau về mặt tương đồng giữa 2 điểm dữ liệu theo tiêu chí phân cụm.

Tiến hành đánh giá mô hình phân cụm trong Weka áp dụng cho tập dữ liệu iris.arff như sau:

- Bước 1: Chọn tập dữ liệu đầu vào là iris.arff, giữ nguyên trường class
- Bước 2: Chuyển tab Cluster, chọn Cluster mode: Classes to clusters evaluation. Click Start để tiến hành xây dựng và đánh giá mô hình phân cụm. Giả sử cho num
- Quan sát kết quả trong Clusterer Output. Incorrectly clustered instances cho biết tỷ lệ sai sót của mô hình phân cụm.



Trực quan hóa các cluster thông qua biểu đồ bằng cách click chuột phải vào SimpleKMeans, chọn Visualize Cluster assignments



### **3.2.4. Một số ứng dụng của bài toán phân cụm**

- Nghiên cứu thị trường khám phá các nhóm khách hàng khác nhau dựa vào hành vi mua hàng của họ
- Nhận dạng mẫu
- Phân tích dữ liệu và xử lý ảnh
- Tìm kiếm trên Web
- ...

## **3.3. Khai phá luật kết hợp**

### **3.3.1. Bài toán khai phá luật kết hợp**

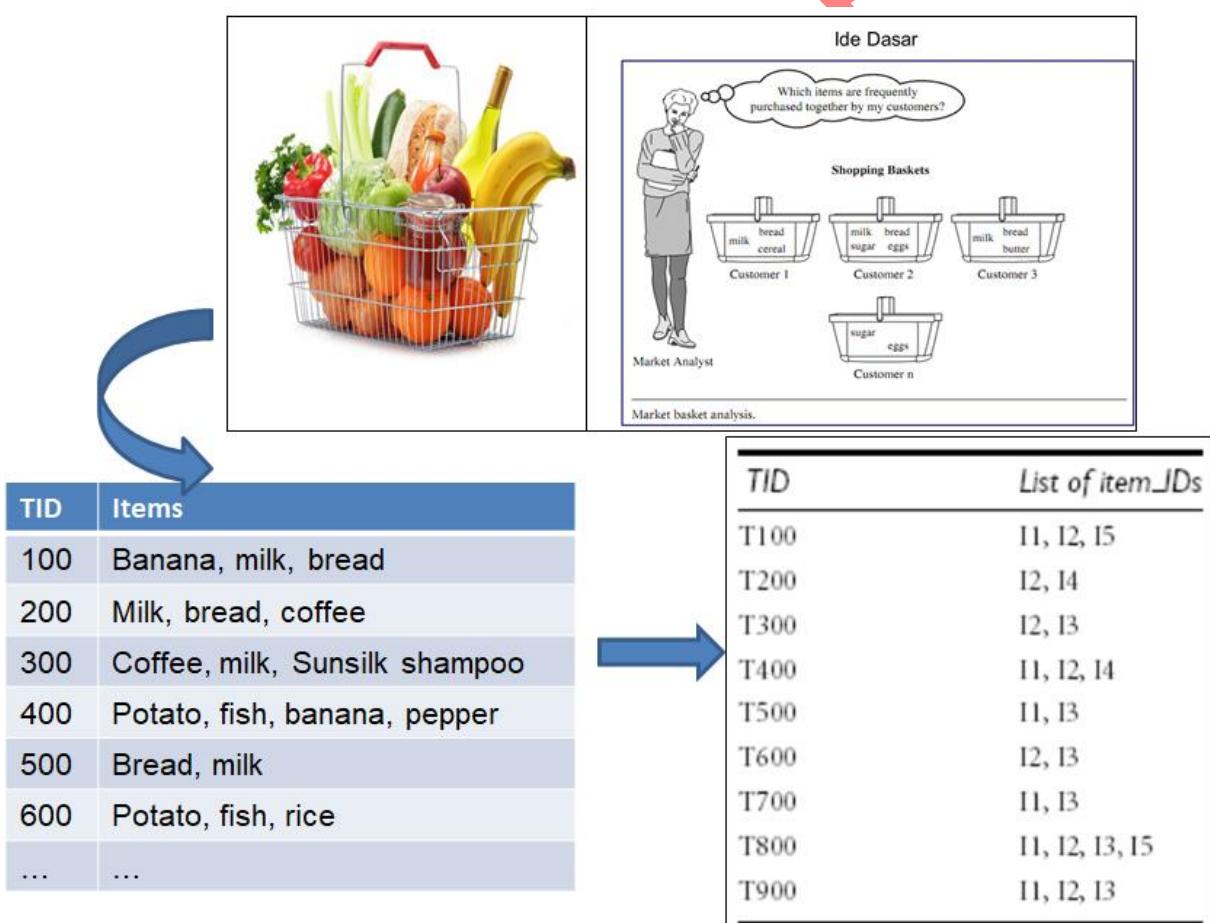
Khai phá luật kết hợp (Association rule in data mining) là một kỹ thuật quan trọng của khai phá dữ liệu với mục tiêu nhằm phát hiện mối quan hệ giữa các đối tượng dữ liệu.

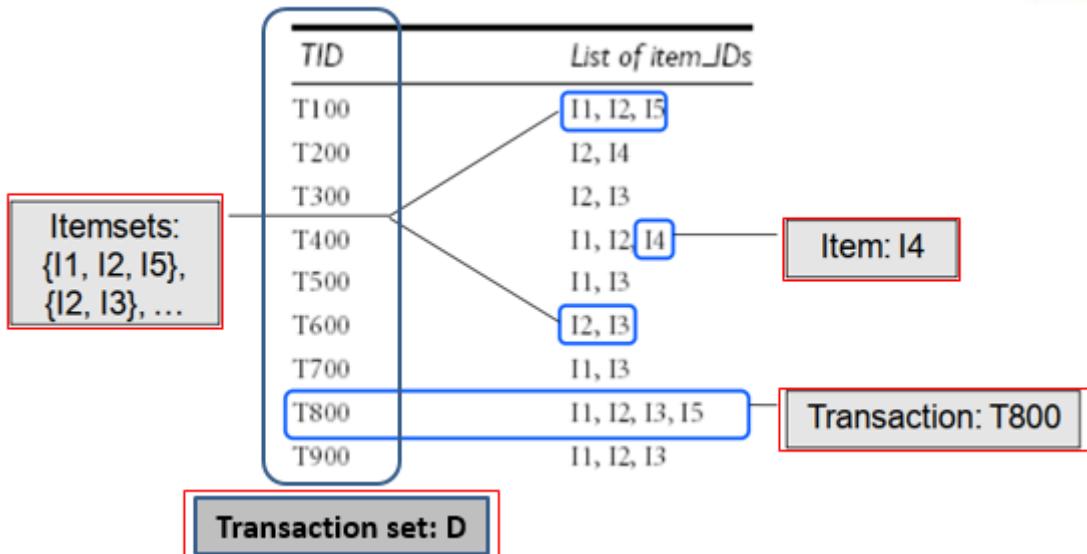
Tiếp cận bài toán phân tích giỏ hàng nhằm ứng dụng khai phá luật kết hợp để sắp đặt các sản phẩm ở các kệ hàng cạnh nhau trong siêu thị



Hình 3.14. Minh họa bài toán phân tích giỏ hàng

Chuyển đổi biểu diễn dữ liệu giỏ hàng như sau:



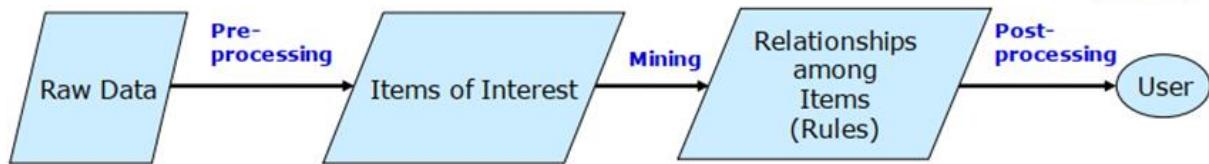


Trong đó các khái niệm cơ bản liên quan được sử dụng trong khai phá luật kết hợp:

- Item (Phần tử): là phần tử / mẫu/ đối tượng dữ liệu trong giỏ hàng.
- Itemset (Tập phần tử):  $I = \{I_1, I_2, \dots, I_m\}$  là tập tất cả các Item có trong mỗi giỏ hàng. Một ItemSet có k items còn được gọi là k-itemset.
- Transaction (Giao dịch): là lần thực hiện tương tác với hệ thống (Ví dụ giao dịch “Khách hàng mua hàng”). Một transaction T liên quan tới một tập hợp các phần tử trong một phiên giao dịch.
- Association (Kết hợp): Các phần tử cùng xuất hiện với nhau trong một hay nhiều giao dịch.
- Association rule (Luật kết hợp): Quy tắc kết hợp có điều kiện giữa các phần tử. Cho A và B là các tập phần tử, luật kết hợp được ký hiệu là  $A \rightarrow B$  ( $B$  xuất hiện trong điều kiện  $A$  xuất hiện).
- Luật  $A \rightarrow B$  trong tập giao dịch D có độ hỗ trợ (support s) : Với s là phần trăm số giao dịch trong D chứa  $A \cup B$ 

$$\text{support}(A \rightarrow B) = P(A \cup B)$$
- Luật  $A \rightarrow B$  trong tập giao dịch D có độ tin cậy (Confidence c): Với c là phần trăm giao dịch trong D có chứa A thì cũng chứa B
$$\text{confidence } (A \rightarrow B) = P(B/A)$$
- Frequent itemset (Tập phần tử phổ biến): A là gồm tập các phần tử phổ biến nếu có support thỏa mãn :  $\text{support}(A) \geq \text{min\_sup}$ .
- Strong association rule (Luật kết hợp mạnh): Luật kết hợp  $A \rightarrow B$  được coi là luật kết hợp mạnh nếu  $\text{support}(A \rightarrow B) \geq \text{min\_sup}$  và  $\text{confidence}(A \rightarrow B) \geq \text{min\_conf}$ .

Quá trình khai phá luật kết hợp như sau:

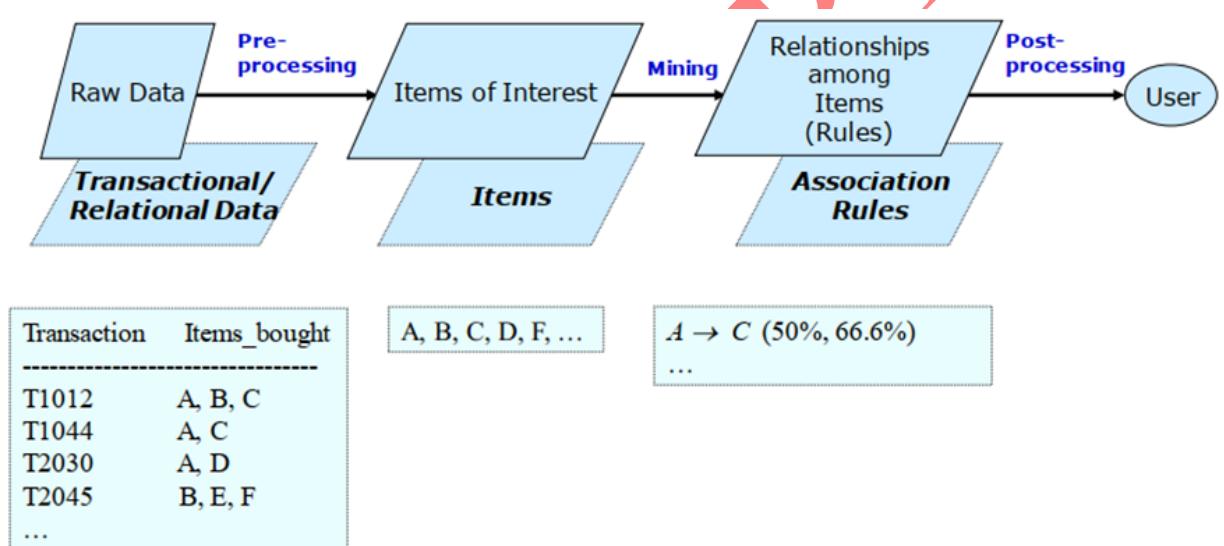


Hình 3.15. Quá trình khai phá luật kết hợp

Như vậy khai phá luật kết hợp là một quá trình 2 bước.

- Tìm tất cả các tập mục thường gặp (thường xuyên): Mỗi itemset được gọi là tập mục thường xuyên nếu độ hỗ trợ của nó lớn hơn hoặc bằng min\_sup.
- Tạo các luật kết hợp mạnh từ các tập mục thường xuyên: Những luật kết hợp mạnh phải có độ hỗ trợ và độ tin cậy lớn hơn min\_sup và min\_conf tương ứng.

Với bài toán phân tích giỏ hàng ở trên, quá trình khai phá luật kết hợp được thể hiện theo hình sau:



Hình 3.16. Quá trình khai phá luật kết hợp cho bài toán phân tích giỏ hàng

### 3.3.2. Phương pháp khai phá luật kết hợp

Một số thuật toán đã được đưa ra để giải quyết bài toán khai phá luật kết hợp được biết đến như : Apriori, FilteredAssociator, FP-Growth. Trong nội dung này bài giảng sẽ trình bày về thuật toán khai phá luật kết hợp cơ bản là Apriori. Trên cơ sở việc tiếp cận phương pháp cơ bản, sinh viên có thể mở rộng nghiên cứu và áp dụng đa dạng các phương pháp khai phá luật kết hợp khác.

#### - Giới thiệu về thuật toán Apriori

Thuật toán Apriori Do R. Agrawal và R. Srikant giới thiệu năm 1994 nhằm khai phá các tập mục thường xuyên cho các luật kết hợp dạng Boolean. Chiến lược lặp của

Apriori: các k-itemset được sử dụng để khảo sát các  $(k + 1)$ -itemset. Trên cơ sở các tập mục thường xuyên tìm được sẽ sử dụng để tạo các luật kết hợp mạnh.

- ***Giải thuật Apriori***

- o Xây dựng danh sách các ứng viên k-itemsets và sau đó trích chọn ra danh sách thường xuyên của k-itemsets dùng min-sup.
- o Sử dụng danh sách thường xuyên k-itemsets để xác định danh sách ứng viên và thường xuyên của  $(k+1)$ -itemsets
- o Loại bỏ các tập mục không thường xuyên
- o Lặp lại cho đến khi danh sách ứng viên và thường xuyên của k-itemsets rỗng
- o Trả lại danh sách của  $(k-1)$ -itemsets.

- ***Minh họa thuật toán Apriori:***

- o Cho min-sup =2
- o Lần lặp 1:

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5
500	1 3 5

UV1	Itemset	Support
	{1}	3
	{2}	3
	{3}	4
	{4}	1
	{5}	4

TX1	Itemset	Support
	{1}	3
	{2}	3
	{3}	4
	{5}	4

- o Lần lặp 2:

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5
500	1 3 5

UV2	Itemset	Support
	{1, 2}	1
	{1, 3}	3
	{1, 5}	2
	{2, 3}	2
	{2, 5}	3
	{3, 5}	3

TX2	Itemset	Support
	{1, 3}	3
	{1, 5}	2
	{2, 3}	2
	{2, 5}	3
	{3, 5}	3

TX1	Itemset	Support
	{1}	3
	{2}	3
	{3}	4
	{5}	4

- o Lần lặp 3:

UV3		UV3	
TID	Items	Itemset	Support
100	1 3 4	{1, 2, 3}	
200	2 3 5	{1, 2, 5}	
300	1 2 3 5	{1, 3, 5}	
400	2 5	{2, 3, 5}	
500	1 3 5		

TX2		TX3	
Itemset	Support	Itemset	Support
{1, 3}	3	{1, 3, 5}	2
{1, 5}	2	{2, 3}	2
{2, 3}	2	{2, 5}	3
{2, 5}	3	{3, 5}	3
{3, 5}	3		

- Lần lặp 4:

UV4		TX4	
TID	Items	Itemset	Support
100	1 3 4	{1, 2, 3, 5}	1
200	2 3 5		
300	1 2 3 5		
400	2 5		
500	1 3 5		

TX3		UV4	
Itemset	Support	Itemset	In TX3?
{1, 3, 5}	2	{1, 2, 3, 5}	
{2, 3, 5}	2	{1, 2, 3}; {1, 2, 5}; {1, 3, 5}; {2, 3, 5}	No

TX3

TX4

Tại lần lặp thứ 4, danh sách thường xuyên của k-itemsets rỗng cho nên thuật toán dừng lại và trả về danh sách của (k-1)-itemsets chính là tập mục thường xuyên sinh ra ở lần lặp thứ 3. Tập mục thường xuyên này sẽ là đầu vào sử dụng để khám phá các luật kết hợp mạnh. Cụ thể như sau:

- **Khám phá các luật kết hợp**

Cho danh sách các tập mục thường xuyên tìm được thông qua thuật toán Apriori

TX3	
Itemset	Support
{1, 3, 5}	2
{2, 3, 5}	2

Sử dụng để sinh ra tất cả các tập mục con khác rỗng của mỗi tập mục thường xuyên:

- Với  $I = \{1, 3, 5\} \rightarrow \{1, 3\}; \{1, 5\}; \{3, 5\}; \{1\}; \{3\}; \{5\}$
- Với  $I = \{2, 3, 5\} \rightarrow \{2, 3\}; \{2, 5\}; \{3, 5\}; \{2\}; \{3\}; \{5\}$

Với mỗi tập con khác rỗng  $s$  của  $I$ , sinh ra luật :  $s \rightarrow (I-s)$  nếu  $\frac{\text{support\_count}(I)}{\text{support\_count}(s)} \geq \text{min\_conf}$ .

Giả sử cho  $\text{min\_conf} = 60\%$ . Ta lần lượt xét qua các luật tìm được xem có thỏa mãn điều kiện trên không:

- R1:  $1 \& 3 \rightarrow 5$ 
  - $\text{Conf} = \text{sup}\{1, 3, 5\} / \text{sup}\{1, 3\} = 2/3 = 66.66\%$
  - R1 được **lựa chọn**
- R2:  $1 \& 5 \rightarrow 3$ 
  - $\text{Conf} = \text{sup}\{1, 3, 5\} / \text{sup}\{1, 5\} = 2/2 = 100\%$
  - R2 được **lựa chọn**
- R3:  $3 \& 5 \rightarrow 1$ 
  - $\text{Conf} = \text{sup}\{1, 3, 5\} / \text{sup}\{3, 5\} = 2/3 = 66.66\%$
  - R3 được **lựa chọn**
- R4:  $1 \rightarrow 3 \& 5$ 
  - $\text{Conf} = \text{sup}\{1, 3, 5\} / \text{sup}\{1\} = 2/3 = 66.66\%$
  - R4 được **lựa chọn**
- R5:  $3 \rightarrow 1 \& 5$ 
  - $\text{Conf} = \text{sup}\{1, 3, 5\} / \text{sup}\{3\} = 2/4 = 50\%$
  - R5 bị **loại bỏ**
- R6:  $5 \rightarrow 1 \& 3$ 
  - $\text{Conf} = \text{sup}\{1, 3, 5\} / \text{sup}\{5\} = 2/4 = 50\%$
  - R6 bị **loại bỏ**

$\text{min\_conf} = 60\%$

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5
500	1 3 5

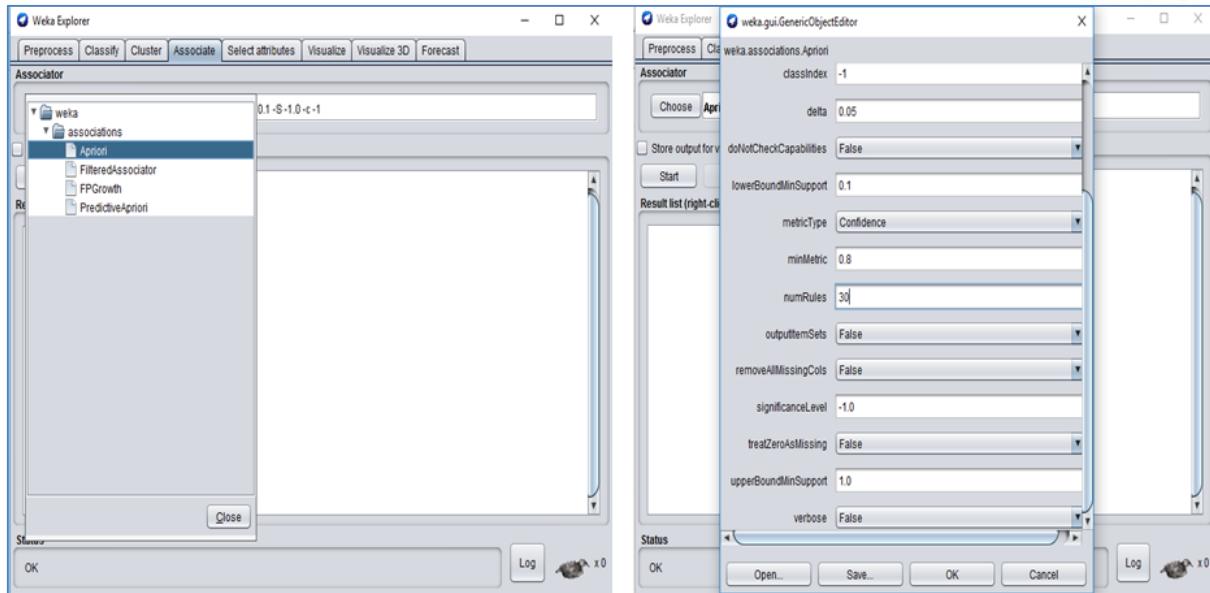
Như vậy có 4 luật kết hợp được khám phá là : R1, R2, R3, R4.

- **Hướng dẫn thực hành xây dựng mô hình khai phá luật kết hợp với thuật toán Apriori bằng Weka Explorer GUI:**

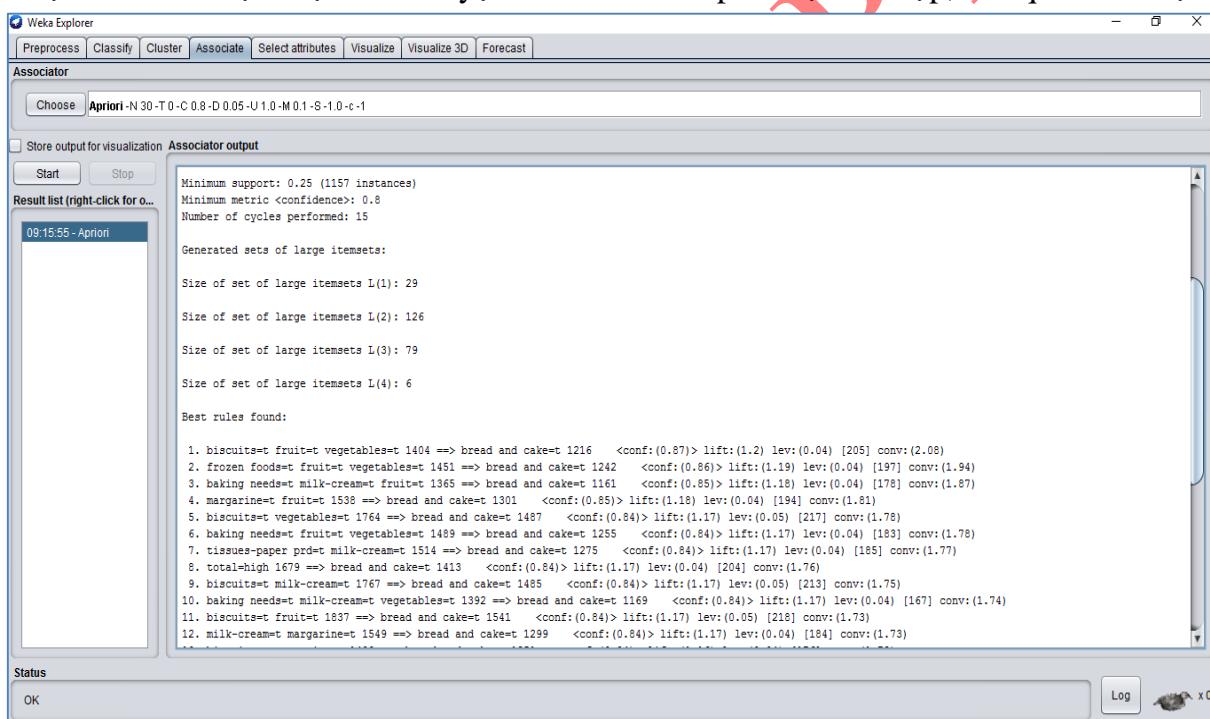
Minh họa việc xây dựng mô hình khai phá luật kết hợp với thuật toán Apriori cho tập dữ liệu supermarket.arff trong Weka Explorer GUI. Các bước thực hiện như sau:

Bước 1: Chọn tab Preprocess nạp tập dữ liệu supermarket.arff

Bước 2: Chuyển sang tab Associate: Associator->Apriori, thiết lập giá trị cần thiết cho tham số đầu vào



Chọn Start để thực hiện huấn luyện mô hình khai phá luật kết hợp, kết quả thu được:



### 3.3.3. Một số ứng dụng của bài toán khai phá luật kết hợp

- Supermarket
- Web mining
- Medical analysis
- Bioinformatics
- Network analysis
- ...

## TỔNG KẾT CHƯƠNG 3

Nội dung chương 3 đã trình bày cụ thể về việc xây dựng và triển khai một số mô hình khai phá dữ liệu điển hình, áp dụng cho các dữ liệu đa phương tiện. Đó là:

- Mô hình phân lớp dữ liệu
- Mô hình phân cụm dữ liệu
- Mô hình khai phá luật kết hợp

Cách tiếp cận trong nội dung chương này đã trình bày từ phát biểu bài toán, áp dụng một số thuật toán khai phá dữ liệu điển hình cho bài toán tiếp cận và đánh giá độ chính xác của mô hình áp dụng. Bên cạnh đó để có thể tiếp cận thực tế cho sinh viên, bài giảng đã đưa nội dung thực hành quá trình xây dựng và đánh giá mô hình khai phá dữ liệu tương ứng trong môi trường Weka theo cả hai hướng (Sử dụng tiện ích của Weka Explorer GUI và sử dụng thư viện Weka API trong lập trình với Java) tương ứng với từng nội dung liên quan.

## CÂU HỎI VÀ BÀI TẬP CHƯƠNG 3

1. Trình bày hiểu biết của em về bài toán phân lớp dữ liệu
2. Thực hiện xây dựng, triển khai và đánh giá mô hình phân lớp dữ liệu với thuật toán kNN trong phân loại văn bản.
3. Thực hiện xây dựng, triển khai và đánh giá mô hình phân lớp dữ liệu với thuật toán kNN trong phân loại hình ảnh.
4. Thực hiện xây dựng, triển khai và đánh giá mô hình phân lớp dữ liệu với thuật toán kNN trong phân loại âm thanh.
5. Thực hiện xây dựng, triển khai và đánh giá mô hình phân lớp dữ liệu với thuật toán kNN trong phân loại video.
6. Trình bày hiểu biết của em về bài toán phân cụm dữ liệu
7. Thực hiện xây dựng, triển khai và đánh giá mô hình phân cụm dữ liệu với thuật toán k-means trong phân cụm văn bản.
8. Thực hiện xây dựng, triển khai và đánh giá mô hình phân cụm dữ liệu với thuật toán k-means trong phân cụm hình ảnh.
9. Thực hiện xây dựng, triển khai và đánh giá mô hình phân cụm dữ liệu với thuật toán k-means trong phân cụm âm thanh.
10. Thực hiện xây dựng, triển khai và đánh giá mô hình phân cụm dữ liệu với thuật toán k-means trong phân cụm video.

11. Trình bày hiểu biết của em về bài toán khai phá luật kết hợp
12. Thực hiện xây dựng, triển khai mô hình khai phá luật kết hợp với thuật toán Apriori trong phân tích giỏ hàng.

THƯ VIỆN PTIT

## TÀI LIỆU THAM KHẢO

- [1] Jiawei Han, Micheline Kamber, Jian Pei, Data Mining – Concepts and Techniques, Third Edition, Morgan Kaufmann Publishers (2012).
- [2] Valery A.Petrushin, Latifur Khan, Multimedia Data Mining and Knowledge Discovery, Springer (2007)
- [3] Pang-ning Tan, Michael Steinbach, Vipin Kumar, Introduction to Data Mining, Pearson-AW (2006).
- [4] Ian H. Witten, Eibe Frank, Mark A. Hall, Christopher J. Pal, Data Mining - Practical Machine Learning Tool and Techniques, Fourth Edition, Morgan Kaufmann Publishers (2016).
- [5] Mehmed Kantardzic, Data Mining – Concepts, Models, Methods and Algorithms, IEEE Press, John Wiley and Sons, INC., Publication (2003).
- [6] Raphael Troncy, Benoit Huet, Multimedia Semantics (Metadata, Analysis and Interaction), Wiley (2011)
- [7] Nguyễn Hà Nam, Nguyễn Chí Thành, Hà Quang Thụy, Giáo trình Khai Phá Dữ Liệu.
- [8] [www.crisp-dm.org](http://www.crisp-dm.org)

THƯ VIỆN