ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA

TRẦN QUANG PHÚC

KHAI PHÁ QUAN ĐIỂM VỚI CÁC PHƯƠNG PHÁP QUẦN THỂ

Ngành: Khoa học máy tính
Mã số: 9480101

LUẬN ÁN TIẾN SĨ KỸ THUẬT

ĐÀ NẴNG – NĂM 2026

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA

TRẦN QUANG PHÚC

KHAI PHÁ QUAN ĐIỂM VỚI CÁC PHƯƠNG PHÁP QUẦN THỂ

LUẬN ÁN TIẾN SĨ KỸ THUẬT

Giảng viên hướng dẫn
GS.TS. Huỳnh Xuân Hiệp
TS. Lê Thị Mỹ Hạnh

ĐÀ NẴNG – NĂM 2026

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu do tôi thực hiện, dưới sự hướng dẫn khoa học của GS.TS Huỳnh Xuân Hiệp (Đại học Cần Thơ) và TS. Lê Thị Mỹ Hạnh (Trường Đại học Bách khoa, Đại học Đà Nẵng).

Tôi cam đoan các kết quả nghiên cứu được trình bày trong luận án là trung thực và không sao chép từ bất kỳ công trình nghiên cứu nào khác. Một số kết quả nghiên cứu là thành quả tập thể và đã được các đồng tác giả đồng ý cho sử dụng trong luận án. Mọi trích dẫn trong luận án đều có ghi nguồn gốc xuất xứ rõ ràng và đầy đủ.

Tác giả

NCS. Trần Quang Phúc

LỜI CẢM ƠN

Trước tiên, tôi xin chân thành cảm ơn Phòng Đào tạo và Bảo đảm chất lượng, Khoa Công nghệ thông tin cũng như các đơn vị có liên quan khác của Trường Đại học Bách khoa, Đại học Đà Nẵng đã luôn tạo điều kiện thuận lợi cho tôi trong thời gian làm nghiên cứu sinh tại đây.

Tôi xin bày tỏ lòng biết ơn sâu sắc và gửi lời tri ân đến thầy GS.TS Huỳnh Xuân Hiệp và cô TS. Lê Thị Mỹ Hạnh đã tận tình hướng dẫn, truyền đạt kiến thức và kinh nghiệm nghiên cứu khoa học cho tôi trong suốt quá trình học tập, nghiên cứu và hoàn thành luận án. Xin cảm ơn đồng nghiệp đã luôn hỗ trợ và tạo điều kiện tốt nhất để tôi hoàn thành nghiên cứu.

Cuối cùng, tôi xin được gửi lời cảm ơn sâu sắc đến gia đình và bạn bè, đồng nghiệp những người luôn bên cạnh, giúp đỡ và động viên tôi trong suốt thời gian học tập, nghiên cứu và hoàn thành luận án.

NCS. Trần Quang Phúc

MỤC LỤC

LỜI CAM ĐOAN ..... **I**
LỜI CẢM ƠN ..... **II**
MỤC LỤC ..... **III**
DANH MỤC CÁC TỪ VIẾT TẮT ..... **IX**
DANH MỤC BẢNG ..... **XIII**
DANH MỤC HÌNH VẼ ..... **XVII**
DANH MỤC KÝ HIỆU TOÁN HỌC ..... **XIX**
PHẦN MỞ ĐẦU **1**
1. Tính cấp thiết của luận án 1
2. Mục tiêu, đối tượng, phương pháp và phạm vi nghiên cứu 3
2.1 Mục tiêu nghiên cứu 3
2.2 Đối tượng nghiên cứu 4
2.3 Phương pháp nghiên cứu 4
2.4 Phạm vi nghiên cứu 4
3. Các đóng góp của luận án 4
4. Bố cục của luận án 5
CHƯƠNG 1. CƠ SỞ LÝ LUẬN VỀ KHAI PHÁ QUAN ĐIỂM VÀ HỌC QUẦN THỂ **8**
1.1. Giới thiệu 8
1.2. Khai phá quan điểm 8
1.2.1. Quan điểm 8
1.2.2. Tài liệu quan điểm 11
1.2.3. Các cấp độ khai phá quan điểm 11
1.2.3.1. Khai phá quan điểm cấp độ tài liệu 11
1.2.3.2. Khai phá quan điểm cấp độ câu 11
1.2.3.3. Khai phá quan điểm cấp độ khía cạnh 12
1.2.4. Các tác vụ khai phá quan điểm 14
1.3. Học quần thể 16
1.3.1. Cây quyết định 16
1.3.1.1. Cây hồi quy 17
1.3.1.2. Cây phân loại 19

1.3.2. Bootstrap 19
1.3.3. Các kỹ thuật học quần thể 20
1.3.3.1. Bagging 20
1.3.3.2. Boosting 21
1.3.3.3. Stacking 22
1.3.4. Mô hình học quần thể với các mô hình phân loại cơ sở giống nhau 23
1.3.4.1. Rừng phân loại 23
1.3.4.2. Rừng mật độ 24
1.3.4.3. Rừng đa tạp 26
1.4. Đánh giá mô hình 28
1.5. Các nghiên cứu liên quan và đề xuất 30
1.5.1. Các nghiên cứu liên quan 30
1.5.1.1. Về kỹ thuật khai phá quan điểm 30
1.5.1.2. Các lĩnh vực áp dụng 39
1.5.2. Một số hạn chế của các tiếp cận khai phá quan điểm hiện nay 47
1.5.2.1. Hạn chế của tiếp cận khai phá quan điểm thông thường 47
1.5.2.2. Một số vấn đề tiếp cận khai phá quan điểm với học quần thể 48
1.5.3. Xu hướng nghiên cứu hiện nay liên quan đến luận án 51
1.5.4. Cơ sở lựa chọn hướng tiếp cận học quần thể 53
1.5.5. Đề xuất nghiên cứu 54
1.6. Trích xuất bộ năm quan điểm 56
1.6.1. Mô hình trích xuất 56
1.6.2. Các giai đoạn trích xuất bộ năm quan điểm 59
1.6.3. Đánh giá 61
1.6.3.1. Các độ đo đánh giá 61
1.6.3.2. Độ đo đồng thuận chú thích 61
1.6.4. Thực nghiệm 62
1.6.4.1. Tập dữ liệu 62
1.6.4.2. Công cụ 62
1.6.4.3. Các kịch bản 63
1.7. Kết luận 65
CHƯƠNG 2. KHAI PHÁ QUAN ĐIỂM VỚI RỪNG PHÂN LOẠI 67

2.1. Giới thiệu 67
2.2. Ma trận bộ năm quan điểm 67
2.3. Mô hình rừng phân loại quan điểm 68
2.3.1. Cây phân loại quan điểm 68
2.3.2. Dựng rừng phân loại quan điểm 70
2.3.3. Dự đoán rừng phân loại quan điểm 72
2.4. Mô hình tổng quan khai phá quan điểm với rừng phân loại 73
2.4.1. Tổng quan mô hình 73
2.4.2. Giai đoạn mô hình hóa quan điểm 74
2.4.3. Giai đoạn khám phá và biểu diễn quan điểm 74
2.4.4. Giai đoạn định hướng quan điểm 75
2.5. Thuật toán và đồ thị tính toán mô hình rừng phân loại quan điểm 75
2.5.1. Thuật toán và độ phức tạp 75
2.5.2. Đồ thị tính toán mô hình rừng phân loại quan điểm 78
2.6. Diễn giải mô hình rừng phân loại quan điểm 79
2.6.1. Diễn giải cây phân loại quan điểm 79
2.6.2. Diễn giải rừng phân loại ngẫu nhiên 82
2.6.2.1. Độ sâu tối thiểu trung bình 82
2.6.2.2. Tầm quan trọng của đặc trưng dựa trên Information Gain 82
2.6.2.3. Tầm quan trọng của đặc trưng dựa trên hoán vị 83
2.6.2.4. Tương tác giữa các đặc trưng 83
2.6.2.5. Biểu đồ phụ thuộc từng phần 84
2.7. Thực nghiệm 85
2.7.1. Tập dữ liệu sử dụng 85
2.7.1.1. Tập dữ liệu Amazon Food Reviews 85
2.7.1.2. Tập dữ liệu TripAdvisor Hotel Reviews 86
2.7.2. Tiền xử lý dữ liệu 87
2.7.2.1. Tiền xử lý dữ liệu Amazon Food Reviews 87
2.7.2.2. Tiền xử lý dữ liệu TripAdvisor Hotel Reviews 88
2.7.3. Các kịch bản thực nghiệm 89
2.7.3.1. Khai phá quan điểm với rừng phân loại dựa trên bộ năm quan điểm trên tập dữ liệu đánh giá thực phẩm Amazon Food Reviews 89

2.7.3.2. Khai phá quan điểm với rừng phân loại dựa trên bộ năm quan điểm trên tập dữ liệu đánh giá khách sạn TripAdvisor Hotel Reviews 92
2.8. Công cụ sử dụng 99
2.9. Kết luận chương 99
CHƯƠNG 3. KHAI PHÁ QUAN ĐIỂM VỚI RỪNG MẬT ĐỘ 100
3.1. Giới thiệu 100
3.2. Biểu diễn và chuẩn hóa bộ năm quan điểm 101
3.2.1. Biểu diễn bộ năm quan điểm 101
3.2.2. Mã hóa và chuẩn hóa quan điểm 102
3.3. Mô hình hóa rừng mật độ quan điểm 103
3.3.1. Phân cụm mật độ quan điểm bằng DBSCAN 103
3.3.2. Cây mật độ quan điểm 104
3.3.3. Rừng mật độ quan điểm 106
3.4. Mã giả và đồ thị tính toán mô hình rừng mật độ quan điểm 106
3.4.1. Thuật toán và độ phức tạp 106
3.4.2. Đồ thị tính toán mô hình rừng mật độ quan điểm 110
3.5. Mô hình tổng quan khai phá quan điểm với rừng mật độ 111
3.6. Diễn giải rừng mật độ quan điểm 112
3.6.1. SHAP 112
3.6.2. Tầm quan trọng của đặc trưng SHAP 113
3.6.3. Giá trị tương tác SHAP 114
3.7. Thực nghiệm 114
3.7.1. Tập dữ liệu sử dụng 114
3.7.2. Tiền xử lý dữ liệu 114
3.7.2.1. Tiền xử lý dữ liệu yelpHotelData 114
3.7.2.2. Tiền xử lý dữ liệu yelpResData 116
3.7.3. Các kịch bản thực nghiệm 119
3.7.3.1. Mô hình khai phá quan điểm với rừng mật độ dựa trên bộ năm quan điểm trên tập dữ liệu khách sạn yelpHotelData 119
3.7.3.2. Mô hình khai phá quan điểm với rừng mật độ trên tập dữ liệu nhà hàng (yelpResData) 126
3.7.4. Thảo luận kết quả thực nghiệm 131

3.7.5. So sánh mô hình đề xuất 132
3.8. Công cụ sử dụng 136
3.9. Kết luận chương 137
CHƯƠNG 4. KHAI PHÁ QUAN ĐIỂM VỚI RỪNG ĐA TẠP 138
4.1. Giới thiệu 138
4.2. Phân cụm quan điểm trên không gian đa tạp 139
4.2.1. Biểu diễn đặc trưng quan điểm 139
4.2.2. Giả thuyết về đa tạp 139
4.2.3. Ý tưởng xây dựng ma trận quan hệ đa tạp 140
4.2.4. Phân cụm DBSCAN trên không gian đa tạp 140
4.3. Cây đa tạp quan điểm 141
4.4. Rừng đa tạp quan điểm 143
4.4.1. Ma trận quan hệ đa tạp 143
4.4.2. Khoảng cách đa tạp 143
4.5. Thuật toán và đồ thị tính toán mô hình rừng đa tạp quan điểm 145
4.5.1. Thuật toán và độ phức tạp 145
4.5.2. Đồ thị tính toán mô hình rừng đa tạp quan điểm 152
4.6. Mô hình tổng quan khai phá quan điểm với rừng đa tạp 153
4.7. Thực nghiệm 154
4.7.1. Tập dữ liệu sử dụng 154
4.7.2. Tiền xử lý dữ liệu 155
4.7.2.1. Tiền xử lý dữ liệu yelpHotelData 155
4.7.2.2. Tiền xử lý dữ liệu yelpResData 156
4.7.3. Các kịch bản thực nghiệm 157
4.7.3.1. Mô hình khai phá quan điểm với rừng đa tạp dựa trên bộ năm quan điểm trên tập dữ liệu khách sạn yelpHotelData 157
4.7.3.2. Mô hình khai phá quan điểm với rừng đa tạp dựa trên bộ năm quan điểm trên tập dữ liệu nhà hàng yelpResData 161
4.8. So sánh tổng hợp 163
4.9. Công cụ sử dụng 166
4.10. Kết luận chương 167
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 168

5.1. Kết luận 168
5.2. Hướng phát triển 173
DANH MỤC CÁC CÔNG TRÌNH KHOA HỌC ĐÃ CÔNG BỐ 176
TÀI LIỆU THAM KHẢO 179
PHỤ LỤC 198
Phụ lục A 198
Phụ lục B 200
Phụ lục C 202
Phụ lục D 206
Phụ lục E 210
Phụ lục F 219
Phụ lục G 230
Phụ lục H 236
Phụ lục I 245
Phụ lục J 252

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Nghĩa tiếng Anh	Nghĩa tiếng Việt
ABSA	Aspect-based sentiment analysis	Phân tích tình cảm dựa trên khía cạnh
Acc	Accuracy	Độ chính xác
AI	Artificial Intelligence	Trí tuệ nhân tạo
ARIMA	AutoRegressive Integrated Moving Average	Mô hình tự hồi quy tích hợp trung bình trượt
ASQP	Aspect Sentiment Quad Prediction	Dự đoán bộ tứ tình cảm
AUC	Area Under the Curve	Diện tích dưới đường cong
BERT	Bidirectional Encoder Representations from Transformers	Biểu diễn mã hóa hai chiều từ Transformers
CA-PCA	Covering Adjustment Principal Component Analysis	Phân tích thành phần chính có hiệu chỉnh vùng phủ
CF	Classification Forest	Rừng phân loại
CNN	Convolutional Neural Network	Mạng nơ-ron tích chập
CNN-1D	1-Dimensional Convolutional Neural Network	Mạng nơ-ron tích chập một chiều
CNN-LSTM	Convolutional Neural Network–Long Short-Term Memory	Mô hình lai CNN–LSTM
CRF	Conditional Random Fields	Trường ngẫu nhiên có điều kiện
CRFs	Conditional Random Fields	Trường ngẫu nhiên có điều kiện
DBSCAN	Density-Based Spatial Clustering of Applications with Noise	Phân cụm không gian dựa trên mật độ của các ứng dụng với nhiễu
DF	Density Forests	Rừng mật độ
DL	Deep Learning	Học sâu
DPC	Density Peaks Clustering	Phân cụm dựa trên đỉnh mật độ
DTC	Decision Tree Classifier	Phân loại cây quyết định
DTs	Decision Trees	Cây quyết định
EL	Ensemble Learning	Học quần thể

EM	Expectation-Maximization	Tối đa hóa kỳ vọng
F_{CF}	Opinion Classification Forest	Rừng phân loại quan điểm
F_{DF}	Opinion Density Forest	Rừng mật độ quan điểm
FinBERT	Financial Bidirectional Encoder Representations from Transformers	Phiên bản BERT dành cho lĩnh vực tài chính
F_{MF}	Opinion Manifold Forest	Rừng đa tạp quan điểm
GBC	Gradient Boosting Classifiers	Bộ phân loại tăng cường độ dốc
GPU	Graphics Processing Unit	Bộ xử lý đồ họa
HITS	Hyperlink-Induced Topic Search	Tìm kiếm chủ đề dựa trên liên kết
IMDb	Internet Movie Database	Cơ sở dữ liệu phim Internet
ISOMAP	Isometric Feature Mapping	Ánh xạ đặc trưng đẳng cự
KNN	K-nearest neighbor	K-hàng xóm gần nhất
LDA	Latent Dirichlet Allocation	Mô hình phân bổ Dirichlet ẩn
LLM	Large Language Model	Mô hình ngôn ngữ lớn
LLMs	Large Language Models	Các mô hình ngôn ngữ lớn
LR	Logistic Regression	Hồi quy Logistic
LSA	Latent Semantic Analysis	Phân tích ngữ nghĩa tiềm ẩn
LSTM	Long Short-Term Memory	Bộ nhớ ngắn hạn dài
MF	Manifold Forests	Rừng đa tạp
MinPts	Minimum Points (Minimum number of points)	Số điểm tối thiểu trong vùng lân cận
ML	Machine Learning	Học máy
MLE	Maximum Likelihood Estimation	Ước lượng hợp lý cực đại
MSE	Mean Squared Error	Sai số bình phương trung bình
NB	Naive Bayes	Thuật toán Naive Bayes
Neg	Negative	Tiêu cực
NER	Named Entity Recognition	Nhận dạng thực thể có tên
NEVs	New Energy Vehicles	Phương tiện sử dụng năng lượng mới
NLP	Natural Language Processing	Xử lý ngôn ngữ tự nhiên

NLTK	Natural Language Toolkit	Bộ công cụ xử lý ngôn ngữ tự nhiên
NMCE	Neural Manifold Clustering and Embedding	Phân cụm và nhúng đa tạp neural
OOB	Out-of-Bag	Ngoài túi mẫu
PCA	Principal Component Analysis	Phân tích thành phần chính
PhoBERT	Pre-trained BERT model for Vietnamese	Mô hình BERT được huấn luyện trước dành cho tiếng Việt
pLSA	Probabilistic Latent Semantic Analysis	Phân tích ngữ nghĩa tiềm ẩn xác suất
PMI	Pointwise Mutual Information	Thông tin tương hỗ theo từng điểm
POS	Part-of-Speech	Nhãn từ loại
Pos	Positive	Tích cực
RF	Random Forests	Rừng ngẫu nhiên
RoBERTa	Robustly Optimized BERT Pretraining Approach	Phương pháp tiền huấn luyện BERT tối ưu
ROC	Receiver Operating Characteristic	Đường cong đặc trưng hoạt động của bộ thu
ROC-AUC	Area Under the Receiver Operating Characteristic Curve	Diện tích dưới đường cong ROC
RSS	Residual Sum of Squares	Tổng bình phương còn lại
R^2	Coefficient of Determination (R^2)	Hệ số xác định
SHAP	SHapley Additive exPlanations	Giải thích cộng tính Shapley
SLM	Small Language Model	Mô hình ngôn ngữ nhỏ
SLMs	Small Language Models	Các mô hình ngôn ngữ nhỏ
SSL	Self-supervised learning	Học tự giám sát
SVM	Support Vector Machine	Máy vector hỗ trợ
TPR	True Positive Rate	Tỷ lệ dương tính đúng (Độ nhạy)
t-SNE	t-distributed Stochastic Neighbor Embedding	Nhúng lân cận ngẫu nhiên phân bố t
TwoNN	Two Nearest Neighbors	Phương pháp hai láng giềng gần nhất

UMAP	Uniform Manifold Approximation and Projection	Xấp xỉ và chiếu đa tạp đồng nhất
URL	Uniform Resource Locator	Định vị tài nguyên thống nhất
VADER	Valence Aware Dictionary and sEntiment Reasoner	Bộ phân tích cảm xúc dựa trên từ điển và quy tắc
VLSP	Vietnamese Language and Speech Processing	Xử lý Ngôn ngữ và Tiếng nói tiếng Việt
WSDEN	Weakly Supervised Discriminative Enhancement Network	Mạng tăng cường khả năng phân biệt dưới cơ chế giám sát yếu
XAI	Explainable Artificial Intelligence	Trí tuệ nhân tạo có thể giải thích
ε (epsilon)	Neighborhood radius (Epsilon radius)	Bán kính lân cận
DANCo	Dimensionality from Angle and Norm Concentration	Phương pháp ước lượng chiều nội tại dựa trên sự tập trung của góc và chuẩn

DANH MỤC BẢNG

Bảng 1.1: Ma trận nhầm lẫn 28
Bảng 1.2: Các mô hình khai phá quan điểm dựa trên từ vựng 32
Bảng 1.3: Các mô hình khai phá quan điểm dựa trên học máy 36
Bảng 1.4: Các mô hình khai phá quan điểm với học quần thể 48
Bảng 1.5: Thống kê tóm tắt của tập dữ liệu Twitter US Airline Sentiment 62
Bảng 1.6: Hiệu suất theo từng thành phần 64
Bảng 1.7: So sánh với các mô hình cơ sở 65
Bảng 2.1: Chi tiết dữ liệu Amazon 85
Bảng 2.2: Dữ liệu huấn luyện và thử nghiệm 86
Bảng 2.3: Thống kê tập dữ liệu 87
Bảng 2.4: Chi tiết tỷ lệ phần trăm của hạng dựa trên các đánh giá 87
Bảng 2.5: Dữ liệu huấn luyện và kiểm tra 88
Bảng 2.6: Thống kê các giá trị của đặc trưng 89
Bảng 2.7: Thiết lập tham số mô hình Rừng phân loại quan điểm trong thực nghiệm ..89
Bảng 2.8: Đánh giá mô hình F_{CF} theo khía cạnh thực phẩm 91
Bảng 2.9: Đánh giá mô hình F_{CF} theo thời gian và người đánh giá 91
Bảng 2.10: Đánh giá mô hình rừng ngẫu nhiên trên bộ dữ liệu khách sạn 92
Bảng 2.11: Thống kê tầm quan trọng của các đặc trưng trong rừng 94
Bảng 2.12: Thống kê sự tương tác của các đặc trưng trong rừng 94
Bảng 3.1: Thống kê các đặc trưng trong bộ dữ liệu yelpHotelData 115
Bảng 3.2: Thống kê các đặc trưng trong bộ dữ liệu yelpResData 117
Bảng 3.3: Chuyển đổi 5 mức sao sang 2 nhãn nhị phân 117
Bảng 3.4: Thiết lập tham số mô hình Rừng mật độ quan điểm trong thực nghiệm 120
Bảng 3.5: Kết quả phân loại tình cảm của rừng mật độ trên YelpHotelData 126
Bảng 3.6: Kết quả phân loại tình cảm của rừng mật độ trên YelpResData 131
Bảng 3.7: Kết quả so sánh trên tập dữ liệu YelpHotelData 132
Bảng 3.8: Kết quả so sánh trên tập dữ liệu YelpResData 133
Bảng 3.9: Tổng hợp kết quả trên hai tập dữ liệu 134
Bảng 4.1: Thiết lập tham số mô hình Rừng đa tạp quan điểm trong thực nghiệm 158
Bảng 4.2: Đánh giá hiệu suất của mô hình rừng đa tạp quan điểm khách sạn 159
Bảng 4.3: Đánh giá hiệu suất của mô hình rừng đa tạp quan điểm nhà hàng 162

Bảng 4.4: So sánh Rừng đa tạp với các mô hình chuẩn 164
Bảng 4.5: So sánh thiết lập tham số ba mô hình rừng 165
Bảng A.1: Cấu trúc bảng đánh giá nhà hàng 198
Bảng A.2: Phân bố điểm đánh giá nhà hàng 198
Bảng A.3: Cấu trúc bảng restaurant 199
Bảng A.4: Cấu trúc bảng *restaurant reviewer* 199
Bảng B.1: Cấu trúc bảng đánh giá khách sạn 200
Bảng B.2: Phân bố điểm đánh giá khách sạn 200
Bảng B.3: Cấu trúc bảng hotel 201
Bảng B.4: Cấu trúc bảng *hotel reviewer* 201
Bảng C.1: Cấu trúc dữ liệu đầu vào – Restaurant 202
Bảng C.2: Cấu trúc dữ liệu đầu ra – Restaurant 203
Bảng C.3: Cấu trúc dữ liệu đầu vào – Hotel 204
Bảng C.4: Cấu trúc dữ liệu đầu ra – Hotel 205
Bảng D.1: Danh sách chi tiết các thư viện R và công cụ sử dụng trong thực nghiệm 206
Bảng E.1: Kết quả thống kê tập dữ liệu yelpHotelData và yelpResData 210
Bảng E.2: Kết quả thống kê tập dữ liệu Amazon Reviews 211
Bảng E.3: Kết quả thống kê tập dữ liệu TripAdvisor Reviews 212
Bảng E.4: Thống kê tổng quan tập dữ liệu VLSP-2018 ABSA 213
Bảng E.5: Phân bố cảm xúc trên các tập dữ liệu 213
Bảng E.6: Top 5 cặp Entity#Attribute phổ biến nhất đối với tập Restaurant Train 214
Bảng E.7: Top 5 cặp Entity#Attribute phổ biến nhất đối với tập Hotel Train 214
Bảng E.8: Quy trình ánh xạ từ bộ ba VLSP-2018 sang bộ năm (e, a, s, h, t) 215
Bảng E.9: Tiêu chí chú thích *h* 216
Bảng E.10: Tiêu chí chú thích *t* theo mùa 216
Bảng E.11: Độ đo đánh giá chú thích và ngưỡng mục tiêu 217
Bảng E.12: Bảng tổng hợp các tập dữ liệu dùng trong thực nghiệm 217
Bảng F.1: Số cặp (đánh giá, khía cạnh) và số bộ năm sau khi loại nhãn neutral 225
Bảng F.2: Phân bố thành phần h và t suy từ từ khóa trên tập Train 225
Bảng F.3: Kết quả trên tập Test VLSP-2018 Restaurant (1.828 bộ năm; ngưỡng tối ưu BA trên Dev) 225
Bảng F.4: Đối chiếu tại ngưỡng cố định 0,5 – VLSP-2018 Restaurant 226

Bảng F.5: Kết quả trên tập Test VLSP-2018 Hotel (2.451 bộ năm; ngưỡng tối ưu BA trên Dev) 227
Bảng F.6: Đối chiếu tại ngưỡng cố định 0,5 – VLSP-2018 Hotel 228
Bảng F.7: Kết quả PhoBERT trên tập Test VLSP-2018 (Restaurant: 1.828 bộ năm; Hotel: 2.451 bộ năm) 229
Bảng G.1: So sánh 3 mô hình × 5 biến thể xử lý mất cân bằng lớp — Hotel 234
Bảng G.2: So sánh 3 mô hình × 5 biến thể xử lý mất cân bằng lớp — Restaurant 235
Bảng H.1: Ablation 2×2 — tập Hotel (mean ± SD, 5 lần lặp) 240
Bảng H.2: Ablation 2×2 — tập Restaurant (mean ± SD, 5 lần lặp) 240
Bảng H.3: Kiểm định McNemar — Hotel 241
Bảng H.4: Kiểm định McNemar — Restaurant 242
Bảng H.5: Kiểm định Wilcoxon — Hotel 243
Bảng H.6: Kiểm định Wilcoxon — Restaurant 243
Bảng I.1: Ablation đặc trưng — Hotel (mean ± SD, 3 lần lặp, một rừng ngẫu nhiên duy nhất, BalancedBoot) 249
Bảng I.2: Ablation đặc trưng — Restaurant (mean ± SD, 3 lần lặp) 249
Bảng I.3: Ước lượng chiều nội tại d′ — Hotel (D = 4 chiều quan sát) 250
Bảng I.4: Ước lượng chiều nội tại d′ — Restaurant (D = 4 chiều quan sát) 250
Bảng I.5: So sánh ba cách mã hóa đặc trưng định danh với F_{CF} trên tập kiểm tra (trung bình ± độ lệch chuẩn qua 3 hạt giống) 251
Bảng J.1: Kết quả Kịch bản 1 (có cân bằng bootstrap và tối ưu ngưỡng) trên Hotel và Restaurant 261
Bảng J.2: Ma trận nhầm lẫn trên tập Test — Hotel 261
Bảng J.3: Ma trận nhầm lẫn trên tập Test — Restaurant 262
Bảng J.4: So sánh thiết lập thực nghiệm giữa các công bố và mục J.1–J.6 262
Bảng J.5: So sánh chỉ số trên tập Hotel 262
Bảng J.6: So sánh chỉ số trên tập Restaurant 263
Bảng J.7: Ba mô hình rừng đề xuất khi chạy lại (trung bình ± độ lệch chuẩn qua 5 lần lặp, Phụ lục H) 263
Bảng J.8: Balanced Accuracy trên tập kiểm tra theo hai cách đưa đầu vào 263
Bảng J.9: Cấu hình huấn luyện của 10 mô hình trong thực nghiệm so sánh thống nhất (Bảng J.8) 264

Bảng J.10: Ảnh hưởng của cách lấy mẫu và cách chia dữ liệu đến chỉ số đánh giá (Random Forest, yelpHotelData, 112.275 bộ năm) 266
Bảng J.11: So sánh thống nhất 10 mô hình trên yelpHotelData (10.000 đánh giá, 5 hạt giống; trung bình ± độ lệch chuẩn trên tập kiểm tra) 267
Bảng J.12: Chênh lệch Balanced Accuracy so với Random Forest (ghép cặp theo hạt giống, số liệu Bảng J.11) 267
Bảng J.13: Các bản điều chỉnh của mô hình rừng đề xuất (chỉ dùng bộ năm; yelpHotelData, 10.000 đánh giá, 5 hạt giống) 268
Bảng J.14: So sánh thống nhất 10 mô hình theo Cách 2 – mọi mô hình chỉ dùng bộ năm (yelpHotelData, 10.000 đánh giá, 5 hạt giống; trung bình ± độ lệch chuẩn trên tập kiểm tra) 270
Bảng J.15: So sánh thống nhất 10 mô hình trên yelpResData theo Cách 1 (10.000 đánh giá, 5 hạt giống; trung bình ± độ lệch chuẩn trên tập kiểm tra) 272
Bảng J.16: So sánh thống nhất 10 mô hình trên yelpResData theo Cách 2 – mọi mô hình chỉ dùng bộ năm (10.000 đánh giá, 5 hạt giống; trung bình ± độ lệch chuẩn trên tập kiểm tra) 274
Bảng J.17: Các bản điều chỉnh của mô hình rừng đề xuất trên yelpResData (chỉ dùng bộ năm; 10.000 đánh giá, 5 hạt giống) 276

DANH MỤC HÌNH VẼ

Hình 1.1: Cây quyết định....16
Hình 1.2: Phân vùng trong không gian đặc trưng hai chiều....18
Hình 1.3: Cây quyết định sau khi phân vùng....18
Hình 1.4: Bootstrap....20
Hình 1.5: Sơ đồ phương pháp Bagging với ba mô hình cơ sở....21
Hình 1.6: Sơ đồ phương pháp Boosting với ba mô hình cơ sở....22
Hình 1.7: Sơ đồ phương pháp Stacking với ba mô hình cơ sở....23
Hình 1.8: Trích xuất bộ năm quan điểm....59
Hình 1.9: Trung bình đồng thuận chú thích....63
Hình 2.1: Phân tách biến dự báo....69
Hình 2.2: Cây quyết định quan điểm về thực phẩm....69
Hình 2.3: Khai phá quan điểm với rừng phân loại....74
Hình 2.4: Đồ thị tính toán của mô hình rừng phân loại quan điểm....78
Hình 2.5: Ví dụ minh họa cây quyết định cho phân loại quan điểm....79
Hình 2.6: Phân bố số lượng đánh giá có quan điểm là tích cực và tiêu cực....88
Hình 2.7: Phân loại quan điểm theo khía cạnh thực phẩm.....90
Hình 2.8: Phân loại quan điểm theo người đánh giá....91
Hình 2.9: Phân loại quan điểm theo thời gian đánh giá....92
Hình 2.10: Phân bố độ sâu tối thiểu của các đặc trưng trong rừng....92
Hình 2.11: Minh họa phân vùng không gian hai chiều{Time, Holder}....95
Hình 2.12: Hiển thị các giá trị dự đoán của rừng ngẫu nhiên đối với {Time, Holder}.96
Hình 2.13: Minh họa phân vùng không gian hai chiều{Entity, Time}....96
Hình 2.14: Hiển thị các giá trị dự đoán của rừng ngẫu nhiên đối với {Entity, Time}..97
Hình 3.1: Đồ thị tính toán của mô hình rừng mật độ quan điểm....111
Hình 3.2: Đề xuất mô hình rừng mật độ quan điểm....112
Hình 3.3: Kết quả phân cụm quan điểm trên tập dữ liệu yelpHotelData....121
Hình 3.4: Sự đóng góp của các đặc trưng....123
Hình 3.5: Phân bổ tầm quan trọng của đặc trưng trên dữ liệu YelpHotelData....123
Hình 3.6: Các cặp tương tác thường xuyên....124
Hình 3.7: Giá trị tương tác SHAP cho các đặc trưng trên dữ liệu YelpHotelData....124
Hình 3.8: Phân phối xác suất dự đoán trong tập huấn luyện....125

Hình 3.9: Phân phối xác suất dự đoán trong tập kiểm tra YelpHotelData 125
Hình 3.10: Kết quả phân cụm DBSCAN quan điểm trên tập dữ liệu yelpResData....127
Hình 3.11: Sự đóng góp của các đặc trưng trên dữ liệu YelpResData.....128
Hình 3.12: Tầm quan trọng của các đặc trưng trên dữ liệu YelpResData.....128
Hình 3.13: Phân phối độ sâu và đặc trưng tối thiểu trên dữ liệu YelpResData 129
Hình 3.14: Giá trị tương tác SHAP trên dữ liệu YelpResData.....129
Hình 3.15: Phân phối xác suất dự đoán trong tập huấn luyện trên YelpResData 130
Hình 3.16: Phân phối xác suất dự đoán trong tập kiểm tra trên YelpResData.....130
Hình 4.1: Đồ thị tính toán của mô hình rừng đa tạp quan điểm.....153
Hình 4.2: Sơ đồ hóa rừng đa tạp quan điểm.....153
Hình 4.3: Phân phối xếp hạng quan điểm khách sạn.....155
Hình 4.4: Phân phối phân cực quan điểm khách sạn.....156
Hình 4.5: Phân phối xếp hạng quan điểm trên bộ dữ liệu yelpResData.....156
Hình 4.6: Phân phối quan điểm nhà hàng là positive và negative.....157
Hình 4.7: So sánh độ chính xác các mô hình trên bộ dữ liệu yelpHotelData 160
Hình 4.8: So sánh độ chính xác các mô hình trên bộ dữ liệu yelpResData 162
Hình C.1: Đoạn mã chuyển đổi trên bộ dữ liệu yelpResData 203
Hình C.2: Đoạn mã chuyển đổi trên bộ dữ liệu yelpHotelData 204
Hình H.1: Thiết kế ablation 2×2 theo không gian phân cụm và cách kết hợp K rừng236

DANH MỤC KÝ HIỆU TOÁN HỌC

Ký hiệu	Chương	Ý nghĩa	Công thức
D	1	Tập hợp các tài liệu quan điểm	-
$o = (e, a, s, h, t)$	1, 2, 3, 4	Bộ năm quan điểm, $o \in \mathbb{R}^5$ *(d=5)*	-
$O = \{o_1, o_2, \dots, o_N\}$	1, 2, 3, 4	Tập dữ liệu quan điểm gồm *N* mẫu	-
N	1, 2, 3, 4	Số mẫu trong tập dữ liệu	-
$C = \{positive, negative\}$	1, 2, 3, 4	Tập nhãn tình cảm	-
n	2, 3, 4	Số cây trong một rừng	-
T_i	2, 3, 4	Cây quyết định/phân loại thứ i trong rừng	-
$\hat{y}$	2, 3, 4	Nhãn tình cảm dự đoán	-
$\mathbb{1}(\cdot)$	1, 2, 3, 4	Hàm chỉ thị: 1 nếu điều kiện đúng, 0 nếu sai	-
O_i^*	2, 3, 4	Tập Bootstrap thứ *i* (lấy mẫu có hoàn lại từ *O*)	-
$\Theta = (f, \theta)$	2, 3, 4	Một phép tách gồm đặc trưng f và ngưỡng θ	-
(g, s)	1	Bộ đôi mục tiêu (target) và tình cảm (sentiment)	-
R_j	1	Vùng phân chia không gian dự đoán trong cây hồi quy *(j=1…J)*	(1.2)
$\hat{y}_{Rj}$	1	Giá trị dự đoán trung bình tại vùng R_j	(1.2)
p_{mk}	1	Tỷ lệ ước lượng của lớp *k* tại nút *m*	(1.4)
H	1	Entropy Shannon	(1.4)
θ_j^*	1	Hàm tối ưu tại nút phân tách *j*	(1.5)
I	1	Tính mức tăng thông tin	(1.6)
$H(S)$	1	Entropy Shannon của tập *S*	(1.7)
Z_t	1	Hằng số chuẩn hóa của cây	(1.12)
$\mathbb{W}^t$	1	Ma trận mối quan hệ của cây phân cụm t	(1.17
κ	1	Độ đo đồng thuận giữa người gán nhãn (Cohen's Kappa)	(1.42)

Ký hiệu	Chương	Ý nghĩa	Công thức
F_{CF}	2	Mô hình rừng phân loại quan điểm (Opinion Classification Forest model)	-
O_{mat}	2	Ma trận N×5 biểu diễn O	(2.2)
$H(O)$	2	Entropy Shannon trên tập quan điểm O	(2.3)
$\chi = \{e, a, h, t\}$	2	Tập đặc trưng đầu vào cho phân tách (không gồm tình cảm s)	-
m_{try}	2	Số đặc trưng ngẫu nhiên chọn tại mỗi nút	(2.8)
$Depth(v)$	2	Độ sâu nút v trong cây	(2.15)
$IG(v)$	2	Độ lợi thông tin tại nút v	(2.18)
$MinDepth_T(f)$	2	Độ sâu tối thiểu của đặc trưng f trong cây T	-
$I_{T_i}(f)$	2	Tầm quan trọng đặc trưng f trong cây quyết định T_i (tổng IG tại các nút dùng f)	(2.20)
$\bar{I}(f)$	2	Tầm quan trọng trung bình của đặc trưng f trên toàn bộ rừng phân loại F_{CF}	(2.21)
$\bar{I}_{\text{norm}}(f)$	2	Tầm quan trọng được chuẩn hóa	(2.22)
$I^{\text{perm}}(f)$	2	Tầm quan trọng của đặc trưng dựa trên hoán vị (Permutation Importance)	(2.23)
$\text{Acc}_{T_i}^{\text{OOB}}$	2	Độ chính xác OOB của cây T_i	(2.23)
$\text{Acc}_i^{\text{OOB, perm}(f)}$	2	Độ chính xác của cây T_i sau khi hoán vị ngẫu nhiên các giá trị của đặc trưng f trên tập OOB	(2.23)
$\text{CDepth}(f_2 \mid f_1)$	2	Độ sâu có điều kiện của đặc trưng f_2 khi f_1 là nút gốc.	(2.24)
$\text{INT}(f_1, f_2)$	2	Độ đo tương tác giữa hai đặc trưng f_1, f_2	(2.26)
$\text{PD}_f(v)$	2	Hàm phụ thuộc từng phần của f tại giá trị v	(2.27)
T_i, T_ℓ	2	Cây phân loại thứ i và thứ ℓ	-
Θ	2	Độ phức tạp thuật toán	-
χ	2	Không gian đặc trưng	-
f_i	2	Đặc trưng thứ i	-
O_i^*	2	Mẫu bootstrap thứ i	-

Ký hiệu	Chương	Ý nghĩa	Công thức
F_{DF}	3	Mô hình rừng mật độ quan điểm (Opinion Density Forest model)	-
μ_k, σ_k	3	Trung bình, độ lệch chuẩn chiều k (chuẩn hóa Z-score)	(3.4)
$N_\varepsilon(o)$	3	Vùng lân cận bán kính ε của điểm o	(3.6)
C_k	3	Cụm mật độ thứ k từ DBSCAN	-
Θ_j^*	3	Phép tách tối ưu tại nút j	(3.9)
IG_D	3	Độ lợi thông tin mật độ	(3.10)
$H_D(O)$	3	Entropy Gaussian không giám sát trên tập quan điểm O	(3.11)
$\lvert\Lambda(O)\rvert$	3	Định thức của ma trận hiệp phương sai Λ được tính từ tập dữ liệu O	(3.12)
μ_l, Λ_l, π_l	3	Trung bình, hiệp phương sai, tỷ lệ điểm tại lá l	-
M	3	Số đặc trưng đầu vào (SHAP)	(3.25)
φ_j	3	Giá trị Shapley của đặc trưng j	(3.26)
F_{MF}	4	Mô hình rừng đa tạp quan điểm (Opinion Manifold Forest model)	-
d'	4	Chiều nội tại của đa tạp $\mathcal{M}$ (scalar) với $d' < 5$	(4.1)
$T_k^{(w)}$	4	Cây không giám sát giai đoạn 1, xây ma trận W (k =1…n_W)	-
n_W	4	Số cây không giám sát để xây W	-
W_{ij}	4	Phần tử ma trận quan hệ đa tạp	-
$IG_{\mathcal{M}}$	4	Độ lợi thông tin đa tạp	(4.6)
$H_{\mathcal{M}}(O)$	4	Entropy đa tạp trên tập quan điểm	(4.7)
$\Lambda_{\mathcal{M}}(O)$	4	Ma trận hiệp phương sai có trọng số đa tạp	(4.8)
$W_{ij}^{(k)}$	4	Ma trận quan hệ thành phần một cây	(4.10)
$dist_{\mathcal{M}}(o_i, o_j)$	4	Hàm khoảng cách đa tạp	(4.13)
$k_{rep} \ll N$	4	Số điểm đại diện khi ánh xạ mẫu mới	-

Ký hiệu	Chương	Ý nghĩa	Công thức
$T_\ell \in F_{MF}$	4	Cây phân loại giai đoạn 2 - biến đếm ℓ=1…n	-
$\hat{W}(o, o_i)$	4	Ước lượng W cho mẫu mới o ∉ O	(4.14)
$\mathcal{C} = \{C_1, \dots, C_K, Noise\}$	4	Kết quả phân cụm DBSCAN trên không gian đa tạp	-
$\mathcal{F}_k$	4	Rừng chuyên môn hóa thứ k trên cụm C_k (với k=1…K)	-
M	4	Mô hình học siêu phân tầng (Stacking Meta-learner)	-

PHẦN MỞ ĐẦU

1. Tính cấp thiết của luận án

Sự phát triển mạnh mẽ của World Wide Web, các nền tảng thương mại điện tử và mạng truyền thông xã hội như Twitter, Facebook, LinkedIn, YouTube,… đã tạo ra một khối lượng dữ liệu văn bản trực tuyến khổng lồ, trong đó chứa đựng nhiều thông tin quan điểm, đánh giá và cảm xúc của người dùng đối với sản phẩm, dịch vụ, sự kiện cũng như các vấn đề kinh tế - xã hội [1][15]. Các quan điểm trực tuyến ngày càng đóng vai trò quan trọng trong việc hỗ trợ ra quyết định của cá nhân và tổ chức.

Đối với doanh nghiệp, các nhận xét và đánh giá của người dùng trên các nền tảng trực tuyến là nguồn thông tin có giá trị để cải thiện chất lượng sản phẩm, dịch vụ và chiến lược kinh doanh. Nhiều nghiên cứu chỉ ra rằng, một tỷ lệ lớn khách hàng có xu hướng thay đổi quyết định mua sắm dựa trên các đánh giá tiêu cực hoặc tích cực được chia sẻ trên internet. Đối với người tiêu dùng, việc phân tích các quan điểm trực tuyến giúp họ hiểu rõ hơn về ưu điểm, hạn chế của sản phẩm, dịch vụ trước khi đưa ra lựa chọn. Trong thực tế, các nền tảng như TripAdvisor, Yelp hay Amazon đã chứng minh vai trò thiết yếu của khai phá quan điểm trong các lĩnh vực như nhà hàng, khách sạn, du lịch và thương mại điện tử [2][3]. Nghiên cứu phân tích tổng hợp (meta-analysis) gần đây của [168] trên 156 nghiên cứu với 69.006 quan sát khẳng định đánh giá trực tuyến có ảnh hưởng mạnh nhất đến ý định mua hàng so với các yếu tố khác, trong đó độ phân cực của đánh giá là yếu tố tác động lớn nhất [168].

Trước đây, nhiều tổ chức tiến hành khảo sát truyền thống để thu thập ý kiến khách hàng. Tuy nhiên, phương pháp này thường tốn kém về thời gian và chi phí, đặc biệt khi số lượng người tham gia lớn. Sự xuất hiện của dữ liệu quan điểm trực tuyến [5] đã tạo điều kiện cho việc tự động hóa quá trình thu thập và phân tích ý kiến người dùng thông qua các kỹ thuật khai phá dữ liệu và học máy. Từ đó, lĩnh vực khai phá quan điểm đã thu hút được sự quan tâm rộng rãi của cộng đồng nghiên cứu và đạt được nhiều kết quả ứng dụng thực tiễn.

Khai phá quan điểm (Opinion Mining), hay phân tích cảm xúc (Sentiment Analysis), là một lĩnh vực liên ngành của khoa học máy tính, kết hợp các kỹ thuật xử lý ngôn ngữ tự nhiên, học máy và khai phá dữ liệu nhằm phân tích, phân loại và suy

diễn thái độ, cảm xúc của con người được thể hiện trong dữ liệu [5][7][9][11]. Trong phạm vi luận án này, nghiên cứu tập trung vào khai phá quan điểm trên dữ liệu văn bản, do đây là dạng dữ liệu phổ biến nhất và có nhiều ứng dụng thực tế. Khảo sát của [170] đã hệ thống hóa các phương pháp xử lý ngôn ngữ tự nhiên (NLP - Natural Language Processing) hiệu quả về mặt tính toán, nhấn mạnh sự đánh đổi giữa độ chính xác và chi phí tài nguyên (thời gian, bộ nhớ, năng lượng), vấn đề liên quan trực tiếp đến lựa chọn mô hình rừng dựa trên cây quyết định thay vì mạng nơ-ron sâu trong luận án, vốn có chi phí tính toán thấp hơn đáng kể [166].

Mặc dù đã đạt được nhiều thành tựu, các hệ thống khai phá quan điểm hiện nay vẫn còn đối mặt với nhiều thách thức [6] như: dữ liệu không cân bằng, nhiễu cao, phụ thuộc miền và ngữ cảnh, cũng như hạn chế về khả năng khái quát hóa và diễn giải của mô hình. Đặc biệt, phần lớn các nghiên cứu hiện nay vẫn tập trung vào các mô hình học đơn lẻ hoặc các phương pháp học sâu, trong khi việc khai thác ảnh hưởng kết hợp của các quan điểm [148] - một đặc trưng tự nhiên của dữ liệu đánh giá trực tuyến - chưa được nghiên cứu một cách hệ thống. Bài viết định hướng có ảnh hưởng lớn [169] chỉ ra rằng mặc dù phân tích tình cảm đã phát triển gần 20 năm với nhiều ứng dụng thương mại, lĩnh vực này vẫn còn nhiều khía cạnh chưa được khai thác đầy đủ để đạt được sự hiểu biết tình cảm thực sự, đặc biệt là khả năng khai thác cấu trúc đa chiều của quan điểm (đối tượng, khía cạnh, người phát biểu, thời điểm) thay vì chỉ phân loại nhãn tình cảm đơn lẻ, đúng định hướng nghiên cứu mà luận án theo đuổi thông qua biểu diễn bộ năm quan điểm.

Khoảng trống này thể hiện rõ trên chính các tập dữ liệu đánh giá thực tế được sử dụng trong luận án. Trên Amazon Food Reviews (568.454 đánh giá), đánh giá 5★ chiếm 63,88% trong khi đánh giá 1★–2★ chỉ chiếm 14,42%; một mô hình chỉ học nhãn tình cảm tổng thể dễ thiên về lớp đa số và bỏ sót phần lớn ý kiến tiêu cực, vốn là thông tin doanh nghiệp cần nhất. Trên TripAdvisor Hotel Reviews (108.891 đánh giá của 1.850 khách sạn), mỗi đánh giá được người dùng chấm điểm riêng cho bảy khía cạnh (giá trị, phòng, vị trí, vệ sinh, lễ tân, dịch vụ, dịch vụ doanh nhân): cùng một đánh giá có thể hài lòng với vị trí nhưng không hài lòng với dịch vụ, và mức độ hài lòng còn khác nhau theo người đánh giá và thời điểm đánh giá. Một nhãn tình cảm duy nhất cho cả đánh giá không lưu giữ được các thông tin này, trong khi biểu

diễn bộ năm (thực thể, khía cạnh, tình cảm, người đánh giá, thời gian) giữ lại được. Thực nghiệm tại Chương 2 cũng cho thấy trên TripAdvisor, độ chính xác 0,91 đi kèm Recall của lớp negative chỉ 0,50 do mất cân bằng nhãn — biểu hiện điển hình của nghịch lý độ chính xác. Ba đặc điểm này của dữ liệu thực tế — cấu trúc đa chiều của quan điểm, mất cân bằng nhãn và quan hệ phi tuyến giữa các thành phần quan điểm — lần lượt là động lực trực tiếp cho ba mô hình rừng phân loại, rừng mật độ và rừng đa tạp được đề xuất trong luận án.

Trong bối cảnh đó, các phương pháp học quần thể [8], với khả năng kết hợp nhiều mô hình cơ sở để tạo ra mô hình mạnh hơn, có tiềm năng lớn trong việc nâng cao độ chính xác, tính ổn định và khả năng khái quát hóa của hệ thống khai phá quan điểm. Tuy nhiên, việc ứng dụng có hệ thống học quần thể [10], đặc biệt là các mô hình rừng, cho khai phá quan điểm vẫn còn là khoảng trống nghiên cứu. Khảo sát toàn diện và cập nhật của [171] đề xuất khung nghiên cứu bốn tầng cho học quần thể, khẳng định việc kết hợp nhiều mô hình cơ sở có tính đa dạng giúp giảm thiên lệch (bias) và phương sai (variance) trong dự đoán, từ đó cải thiện nhất quán độ chính xác và khả năng khái quát hóa so với mô hình đơn lẻ, đúng cơ sở lý luận cho ba mô hình rừng (phân loại, mật độ, đa tạp) được đề xuất xuyên suốt luận án.

Xuất phát từ những lý do trên, luận án với đề tài "Khai phá quan điểm với các phương pháp quần thể" được thực hiện nhằm nghiên cứu và đề xuất các mô hình khai phá quan điểm dựa trên học quần thể, góp phần nâng cao hiệu quả, độ chính xác và khả năng diễn giải của các hệ thống khai phá quan điểm hiện nay.

2. Mục tiêu, đối tượng, phương pháp và phạm vi nghiên cứu

2.1 Mục tiêu nghiên cứu

Mục tiêu tổng quát của luận án là nghiên cứu và đề xuất các mô hình khai phá quan điểm dựa trên các phương pháp học quần thể nhằm khắc phục những hạn chế của các mô hình khai phá quan điểm truyền thống.

Các mục tiêu cụ thể của luận án bao gồm:

- Khảo sát tổng quan các nghiên cứu và hệ thống khai phá quan điểm, làm rõ những thành tựu và hạn chế hiện nay.

- Nghiên cứu mô hình quan điểm có cấu trúc dưới dạng bộ năm (thực thể, khía cạnh, tình cảm, người đưa ra quan điểm, thời gian) và các kỹ thuật trích xuất bộ năm từ dữ liệu văn bản.

- Đề xuất mô hình khai phá quan điểm theo ảnh hưởng kết hợp nhằm nâng cao độ chính xác, khả năng khái quát hóa và khả năng diễn giải.

- Xây dựng mô hình khai phá quan điểm dựa trên rừng phân loại, rừng mật độ và rừng đa tạp, phù hợp với đặc trưng dữ liệu quan điểm có cấu trúc.

- Đánh giá và so sánh các mô hình đề xuất với các phương pháp khai phá quan điểm hiện có trên các tập dữ liệu thực tế.

2.2 Đối tượng nghiên cứu

Đối tượng nghiên cứu của luận án bao gồm:

- Dữ liệu văn bản chứa quan điểm của người dùng.

- Quan điểm ở các mức độ khác nhau, trong đó tập trung vào mức độ khía cạnh.

- Các kỹ thuật trích xuất và biểu diễn dữ liệu quan điểm dưới dạng bộ năm.

- Các phương pháp học quần thể, đặc biệt là các mô hình rừng, áp dụng cho khai phá quan điểm.

2.3 Phương pháp nghiên cứu

Luận án sử dụng kết hợp các phương pháp nghiên cứu sau:

- Phương pháp nghiên cứu tài liệu: phân tích, tổng hợp các công trình nghiên cứu liên quan đến khai phá quan điểm, học quần thể và xử lý ngôn ngữ tự nhiên.

- Phương pháp thực nghiệm: xây dựng, cài đặt và đánh giá các mô hình đề xuất trên dữ liệu quan điểm thực tế nhằm kiểm chứng hiệu quả của phương pháp.

2.4 Phạm vi nghiên cứu

Luận án được giới hạn trong các nội dung sau:

- Trích xuất và chuẩn hóa bộ năm quan điểm từ dữ liệu văn bản.

- Nghiên cứu các phương pháp học quần thể với các mô hình cơ sở đồng nhất.

- Áp dụng các mô hình rừng phân loại, rừng mật độ và rừng đa tạp cho khai phá quan điểm.

- Đánh giá mô hình trên dữ liệu đánh giá.

3. Các đóng góp của luận án

Trong luận án, bốn đóng góp khoa học chính được trình bày như sau:

- Thứ nhất, mô hình rừng phân loại quan điểm dựa trên ma trận bộ năm. Trong mô hình này, mỗi câu quan điểm được biểu diễn dưới dạng ma trận bộ năm (entity, aspect, sentiment, holder, time) thay vì vector đặc trưng đơn lẻ, qua đó cấu trúc ngữ nghĩa được bảo toàn khi các cây phân loại được xây dựng. Kết quả đã được công bố trong [Bài báo 2, Bài báo 3].

- Thứ hai, mô hình rừng mật độ quan điểm tích hợp phân cụm không gian dựa trên mật độ của các ứng dụng với nhiễu (DBSCAN - Density-Based Spatial Clustering of Applications with Noise) và phương pháp giải thích mô hình dựa trên giá trị Shapley (SHAP - SHapley Additive exPlanations). Trong mô hình này, thuật toán phân cụm DBSCAN được tích hợp tại các nút lá để dữ liệu mất cân bằng được xử lý theo mật độ phân bố; phương pháp SHAP và chỉ số tương tác Shapley được sử dụng để phân tích và diễn giải tầm quan trọng và mối tương tác giữa các đặc trưng của bộ năm. Kết quả đã được công bố trong [Bài báo 4, Bài báo 5].

- Thứ ba, mô hình rừng đa tạp quan điểm được đề xuất nhằm khai thác cấu trúc đa tạp thấp chiều của không gian đặc trưng. Thay vì sử dụng các phương pháp chiếu đa tạp truyền thống như phương pháp ánh xạ đặc trưng đẳng cự (ISOMAP - Isometric Feature Mapping) hay phương pháp xấp xỉ và chiếu đa tạp đồng nhất (UMAP - Uniform Manifold Approximation and Projection), luận án đề xuất học cấu trúc đa tạp trực tiếp từ rừng ngẫu nhiên không giám sát thông qua tính chất đồng xuất hiện tại nút lá, không cần giải bài toán trị riêng toàn cục. Tính phù hợp của giả thuyết đa tạp được kiểm chứng thông qua kết quả thực nghiệm so sánh với các mô hình khác tại §4.8. Kết quả đã được công bố trong [Bài báo 6].

- Thứ tư, hai tập dữ liệu YelpHotelData và YelpResData được sử dụng trong luận án là dữ liệu công bố [3]; dữ liệu tiếng Việt là dữ liệu của cộng đồng trên tập dữ liệu VLSP-2018 [197]. Đóng góp của luận án về mặt dữ liệu nằm ở quy trình chuyển đổi các tập dữ liệu này sang cấu trúc bộ năm quan điểm (entity, aspect, sentiment, holder, time), phục vụ huấn luyện và đánh giá các mô hình rừng đề xuất, được công bố trong [Bài báo 4, Bài báo 5, Bài báo 6, Bài báo 9]; mô hình trích xuất bộ năm quan điểm từ văn bản (mục 1.6) được công bố trong [Bài báo 7, Bài báo 8].

4. Bố cục của luận án

Luận án được tổ chức theo bố cục gồm 4 chương, cụ thể như sau:

- **Chương 1: Cơ sở lý luận về khai phá quan điểm và học quần thể**

Chương này trình bày tổng quan về lĩnh vực khai phá quan điểm, bao gồm các khái niệm cơ bản về quan điểm, bộ năm thành phần quan điểm và các cấp độ phân tích tình cảm. Chương cũng trình bày nền tảng lý thuyết về học quần thể, đặc biệt là phương pháp Bagging và các biến thể rừng ngẫu nhiên, cùng các phương pháp biểu diễn đặc trưng văn bản, kỹ thuật xử lý dữ liệu mất cân bằng, và tổng quan về các tập dữ liệu thực nghiệm được sử dụng trong luận án. Ngoài ra, chương khảo sát các hướng ứng dụng và hạn chế của các tiếp cận hiện tại, từ đó xác định rõ khoảng trống nghiên cứu mà luận án hướng đến giải quyết.

- **Chương 2: Khai phá quan điểm với rừng phân loại**

Chương này đề xuất mô hình Rừng phân loại quan điểm (Opinion Classification Forest - F_{CF}) dựa trên nguyên lý Bagging kết hợp với cây quyết định có cấu trúc quan điểm. Mô hình sử dụng bộ năm thành phần quan điểm *(e, a, s, h, t)* làm đơn vị phân loại cơ sở, tích hợp cơ chế giải thích mô hình qua tầm quan trọng đặc trưng (Feature Importance) và biểu đồ phụ thuộc từng phần (PDP) để tăng tính minh bạch (phương pháp SHAP được tích hợp trong mô hình rừng mật độ tại Chương 3). Thực nghiệm của chương được thực hiện trên hai tập dữ liệu Amazon Food Reviews và TripAdvisor Hotel Reviews theo ba kịch bản diễn giải (theo khía cạnh; theo thời gian và người đánh giá; theo thực thể và thời gian), tập trung vào độ chính xác và khả năng diễn giải của F_{CF}; trên hai tập này, chương không so sánh F_{CF} với các mô hình khác. Việc so sánh F_{CF} cùng hai mô hình đề xuất còn lại với các mô hình học máy (Random Forest, Gradient Boosting, SVM) và các mô hình học sâu (LSTM, BiLSTM-Attention, BERT, RoBERTa) được thực hiện trên dữ liệu Yelp tại Chương 3 và Chương 4 (Bảng 3.7–3.9, Bảng 4.4). Thực nghiệm bổ sung trên dữ liệu tiếng Việt VLSP-2018 với năm mô hình cơ sở (Logistic Regression, SVM tuyến tính, Naive Bayes, Random Forest, Gradient Boosting) và mô hình tham chiếu PhoBERT được trình bày tại Phụ lục F.

- **Chương 3: Khai phá quan điểm với rừng mật độ**

Chương này đề xuất mô hình Rừng mật độ quan điểm (Opinion Density Forest - F_{DF}), phù hợp với không gian đặc trưng có phân phối không đồng đều. Mô hình xây dựng cây mật độ quan điểm bằng cách sử dụng entropy Gaussian kết hợp phân cụm

mật độ DBSCAN làm tiêu chí phân chia nút, cho phép phát hiện các nhóm quan điểm thiểu số mà các phương pháp phân loại thông thường bỏ sót. Kết quả thực nghiệm trên các tập dữ liệu đánh giá khách sạn *YelpHotelData* và nhà hàng *YelpResData* cho thấy khả năng áp dụng của mô hình đối với dữ liệu thực tế có cấu trúc phức tạp.

- **Chương 4: Khai phá quan điểm với rừng đa tạp**

Chương này đề xuất mô hình Rừng đa tạp quan điểm (Opinion Manifold Forest - F_{MF}) khai thác giả thuyết đa tạp, rằng dữ liệu quan điểm thực tế nằm trên một đa tạp chiều thấp nhúng trong không gian đặc trưng chiều cao để học biểu diễn nén và phân loại quan điểm hiệu quả hơn. Mô hình xây dựng cây đa tạp quan điểm bằng cách học ma trận quan hệ đồng xuất hiện tại nút lá (co-occurrence) từ rừng ngẫu nhiên không giám sát, qua đó bảo toàn cấu trúc hình học cục bộ của dữ liệu mà không cần giải bài toán trị riêng toàn cục như ISOMAP hay UMAP. Thực nghiệm so sánh toàn diện trên hai tập dữ liệu *YelpResData* và *YelpHotelData* cho thấy, theo quy trình chia ngẫu nhiên theo dòng bộ năm của các bài báo đã công bố, mô hình đề xuất đạt độ chính xác trung bình 0,80 (Bảng 4.4), cao nhất trong nhóm các mô hình học quần thể được thực nghiệm, đồng thời cung cấp khả năng diễn giải qua cấu trúc cụm và ma trận quan hệ đa tạp; khi chia theo đánh giá gốc (Phụ lục J), độ chính xác của mọi mô hình dùng bộ năm giảm đáng kể và rừng đa tạp thấp hơn Random Forest do bước phân cụm đa tạp chỉ học trên mẫu con.

Phần nội dung cuối cùng của luận án là KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN sẽ tổng hợp, cô đọng các kết quả thu được của luận án, đồng thời nêu ra các hướng phát triển về nghiên cứu, mở rộng nội dung liên quan đến kết quả của luận án trong thời gian tiếp theo.

Mối liên hệ giữa 9 công trình đã công bố của nghiên cứu sinh với các chương, mục và số liệu của luận án (bao gồm vai trò của biểu diễn Doc2vec tại Bài báo 1 và sự khác biệt giữa Bài báo 4 và Bài báo 6) được tổng hợp tại "Bảng đối chiếu công trình – chương – nội dung sử dụng" được đặt sau Danh mục các công trình khoa học đã công bố.

CHƯƠNG 1. CƠ SỞ LÝ LUẬN VỀ KHAI PHÁ QUAN ĐIỂM VÀ HỌC QUẦN THỂ

1.1. Giới thiệu

Chương này đặt nền móng cho toàn bộ luận án bằng cách trình bày cái nhìn tổng quan về khai phá quan điểm và học quần thể, bao gồm cả lý thuyết và ứng dụng thực tế như: lý thuyết về khai phá quan điểm, các cấp độ khai phá quan điểm, mô hình hóa bộ năm quan điểm, các tác vụ khai phá quan điểm, các kỹ thuật trích xuất bộ năm quan điểm; tìm hiểu học quần thể; tìm hiểu các nghiên cứu có liên quan, xác định các vấn đề hạn chế trong các mô hình khai phá quan điểm trước đây để có cơ sở đề xuất hướng nghiên cứu của luận án. Chương này cũng trình bày kết quả nghiên cứu, phát triển và xây dựng mô hình, thuật toán trích xuất bộ năm cho khai phá quan điểm, đây là nền tảng cơ sở cho việc áp dụng học quần thể trong các chương tiếp theo. Mô hình trích xuất bộ năm quan điểm thực hiện phân tích dự đoán trích xuất bộ năm quan điểm và xác định mối quan hệ phụ thuộc lẫn nhau của các thành phần gồm thực thể, khía cạnh, cảm xúc, người giữ quan điểm và thời gian từ tài liệu quan điểm với độ chính xác tương đối, có khả năng diễn giải và sử dụng tài nguyên thấp.

Nội dung chính của chương bao hàm các kết quả nghiên cứu của Bài báo 1, Bài báo 7, Bài báo 8, Bài báo 9. Trong đó, Bài báo 1 khảo sát phương pháp biểu diễn đặc trưng văn bản bằng Doc2vec kết hợp với các bộ phân loại học quần thể (Gradient Boosting, Random Forest) để phân loại quan điểm trên tập dữ liệu Amazon Fine Food Reviews. Bài báo 1 là bước khảo sát ban đầu về khả năng biểu diễn đặc trưng cho bài toán khai phá quan điểm, làm cơ sở thực nghiệm để luận án lựa chọn hướng biểu diễn bộ năm quan điểm có cấu trúc (e, a, s, h, t) thay cho biểu diễn vector nhúng không tường minh về ngữ nghĩa như Doc2vec làm đặc trưng đầu vào thống nhất cho ba mô hình rừng học quần thể được đề xuất trong các chương tiếp theo.

1.2. Khai phá quan điểm

1.2.1. Quan điểm

Quan điểm là một khái niệm rộng bao gồm tình cảm, đánh giá, thẩm định hoặc thái độ và các thông tin có liên quan như mục tiêu quan điểm và người đưa ra quan điểm.

Quan điểm [1] được mô hình hóa có hai thành phần chính là mục tiêu và tình cảm đối với mục tiêu, (g, s), trong đó g có thể là bất kỳ thực thể hoặc khía cạnh nào của thực thể mà quan điểm đã được bày tỏ và s là tích cực, tiêu cực, hoặc tình cảm trung lập hoặc đánh giá tình cảm bằng số. Tích cực, tiêu cực và trung lập được gọi là định hướng tình cảm hoặc quan điểm. Mục tiêu tình cảm hay còn gọi là mục tiêu quan điểm của một quan điểm là thực thể hoặc một bộ phận hoặc thuộc tính của thực thể mà quan điểm đó được thể hiện. Một quan điểm được đưa ra bởi người giữ quan điểm là cá nhân hoặc tổ chức tại thời gian [1] là thời điểm đưa ra bởi người giữ quan điểm.

Cụ thể, quan điểm [1] được mô hình hóa thành một bộ tứ (g, s, h, t) trong đó g là mục tiêu tình cảm, s là tình cảm của quan điểm về mục tiêu g và h là người giữ quan điểm có nghĩa là người hoặc tổ chức có quan điểm, t là thời điểm bày tỏ quan điểm.

Tình cảm của quan điểm [1] là cảm giác, thái độ, đánh giá hoặc cảm xúc tiềm ẩn liên quan đến một quan điểm. Nó được biểu diễn dưới dạng bộ ba (y, o, i), trong đó y là loại tình cảm, o là định hướng của tình cảm, i là cường độ của tình cảm. Tình cảm có thể được phân thành nhiều loại. Đối với người tiêu dùng, phân loại tình cảm thành hai loại: tình cảm lý trí (rational sentiment) và tình cảm cảm xúc *(*emotional sentiment).

- *Tình cảm lý trí (Rational sentiment).* Tình cảm lý trí là từ lý luận hợp lý, niềm tin hữu hình và thái độ vị lợi. Tình cảm lý trí không bày tỏ cảm xúc.

- *Tình cảm cảm xúc (Emotional sentiment).* Tình cảm cảm xúc là từ những phản ứng vô hình, cảm tính đối với thực thể mà đi sâu vào các trạng thái tâm lý của con người.

Tình cảm cảm xúc mạnh hơn tình cảm lý trí và quan trọng hơn trong thực tế.

Định hướng tình cảm (Sentiment orientation). Định hướng tình cảm có thể tích cực, tiêu cực hoặc trung lập. Trung lập thường có nghĩa là sự vắng mặt tình cảm hoặc không có tình cảm. Định hướng tình cảm còn được gọi là phân cực, định hướng ngữ nghĩa hoặc giá trị trong các tài liệu nghiên cứu.

Cường độ tình cảm (Sentiment intensity). Tình cảm mỗi loại vẫn có thể có mức độ mạnh yếu khác nhau. Mọi người thường sử dụng hai cách để thể hiện cường độ cảm xúc của họ trong văn bản. Đầu tiên là lựa chọn những từ, cụm từ trong văn bản

là những biểu hiện tình cảm có thế mạnh phù hợp. Thứ hai là việc sử dụng các thuật ngữ thay đổi mức độ tình cảm được thể hiện theo hướng tăng cường và thu nhỏ.

Đánh giá tình cảm (Sentiment rating). Trong các ứng dụng thực tế, một số xếp hạng rời rạc thể hiện cường độ tình cảm. Năm cấp độ (ví dụ: 1–5 sao) thường được sử dụng, có thể được hiểu như sau dựa trên hai loại tình cảm.

Cảm xúc tích cực (+2 hoặc 5 sao).

Tích cực lý trí (+1 hoặc 4 sao).

Trung lập (0 hoặc 3 sao).

Tiêu cực lý trí (−1 hoặc 2 sao).

Cảm xúc tiêu cực (−2 hoặc 1 sao).

Quan điểm [1] cũng có thể biểu diễn bởi một bộ năm (e, a, s, h, t), trong đó e là thực thể đích, a là khía cạnh đích của thực thể e mà quan điểm hướng tới, s là tình cảm của quan điểm về khía cạnh a của đối tượng e, h là người đưa ra quan điểm, t là thời gian đăng quan điểm; s có thể là tích cực, tiêu cực hoặc trung lập hoặc xếp hạng (ví dụ: 1–5 sao). Trong trường hợp quan điểm hướng đến toàn bộ thực thể, khía cạnh đặc biệt GENERAL được sử dụng để thể hiện quan điểm đó. Ở đây e và a cùng đại diện cho mục tiêu quan điểm.

Thực thể e được biểu diễn bởi chính nó dưới dạng một tổng thể và một tập hữu hạn các khía cạnh $A = \{a_1, a_2, \dots, a_s\}$ trong đó e có thể được biểu thị bằng văn bản với bất kỳ một trong tập hữu hạn các biểu thức thực thể của nó $A = \{ee_1, ee_2, \dots, ee_n\}$. Mỗi khía cạnh $a \in A$ của thực thể e có thể được biểu diễn bằng bất kỳ một trong tập hữu hạn các biểu thức khía cạnh của nó $\{ae_1, ae_2, \dots, ae_m\}$.

Khía cạnh [12] của một thực thể e là các thành phần và thuộc tính của e. Biểu thức khía cạnh là một từ hoặc cụm từ thực tế đã xuất hiện trong văn bản biểu thị một khía cạnh.

Trong các ứng dụng thực tế, việc khám phá lý do cho mỗi quan điểm tích cực hoặc tiêu cực có thể rất quan trọng vì có thể chính những lý do này cho phép một người thực hiện các hành động để khắc phục tình huống và việc biết từ hạn định cũng rất quan trọng trong thực tế vì nó cho biết quan điểm đó phù hợp với điều gì.

Lý do (Reason). Lý do là nguyên nhân hoặc giải thích quan điểm.

Hạn định (Qualifier). Hạn định của một quan điểm là thành phần giới hạn hoặc thay đổi ý nghĩa của quan điểm đó. Nó làm cho quan điểm trở nên chính xác hơn hoặc hạn chế phạm vi áp dụng của quan điểm.

Ví dụ: Câu "The camera is quite good", từ "quite" là hạn định có tác động giảm mức độ tích cực (không hoàn toàn tốt).

1.2.2. Tài liệu quan điểm

Một tài liệu quan điểm [1] D chứa các quan điểm về một tập hợp hữu hạn các thực thể $\{e_1, e_2, \dots, e_r\}$ và một tập hợp con các khía cạnh của mỗi thực thể. Các quan điểm là từ một nhóm hữu hạn những người có quan điểm $\{h_1, h_2, \dots, h_p\}$ và được đưa ra tại một thời điểm cụ thể t.

1.2.3. Các cấp độ khai phá quan điểm

Khai phá quan điểm chủ yếu nghiên cứu ở ba cấp độ [1] chi tiết là cấp độ tài liệu, cấp độ câu và cấp độ khía cạnh.

1.2.3.1. Khai phá quan điểm cấp độ tài liệu

Phân loại tình cảm ở cấp độ tài liệu là phân loại một tài liệu có quan điểm thành thể hiện quan điểm tổng thể tích cực hoặc tiêu cực. Nó coi toàn bộ tài liệu là thông tin cơ bản và giả định rằng tài liệu là có quan điểm và chứa các quan điểm về thực thể duy nhất (ví dụ như một chiếc điện thoại cụ thể). Phân loại cảm tính ở cấp độ câu sẽ phân loại các câu riêng lẻ trong một tài liệu. Tuy nhiên, không phải câu nào cũng có quan điểm. Theo truyền thống, trước tiên thường phân loại một câu là có quan điểm hay không có quan điểm, được gọi là phân loại chủ quan. Sau đó, các câu đưa ra quan điểm được phân loại thành thể hiện quan điểm tích cực hoặc tiêu cực.

Khai phá quan điểm ở cấp độ tài liệu tập trung vào phân loại tổng thể tình cảm của toàn bộ văn bản trong tài liệu. Nhiệm vụ là xác định xem tài liệu là tích cực, tiêu cực hoặc trung lập về một đối tượng nhất định. Mức độ phân tích cấp độ tài liệu [1] mặc nhiên giả định rằng mỗi tài liệu bày tỏ quan điểm về một thực thể (ví dụ: một sản phẩm hoặc dịch vụ). Do đó, nó không thể áp dụng cho các tài liệu đánh giá hoặc so sánh nhiều thực thể mà cần phân tích chi tiết hơn.

1.2.3.2. Khai phá quan điểm cấp độ câu

Phân loại tình cảm ở cấp độ câu được phân loại hai lớp, nghĩa là phân loại một câu là tích cực hoặc tiêu cực.

Khai phá quan điểm cấp độ câu tập trung xác định xem mỗi câu thể hiện quan điểm tích cực hay tiêu cực. Mức độ khai phá quan điểm cấp độ câu [1] thường có liên quan chặt chẽ đến phân loại chủ quan, phân biệt câu thể hiện thông tin thực tế (được gọi là câu khách quan) với câu thể hiện quan điểm chủ quan. Trong nghiên cứu [13] đã đề xuất phương pháp bootstrapping để tự động xây dựng tập huấn luyện phân loại câu chủ quan/khách quan mà không cần dữ liệu được gán nhãn thủ công. Quy trình gồm hai mô hình phân loại hoạt động song song: mô hình phân loại chủ quan (subjective classifier) với độ chính xác cao nhưng độ phủ thấp, tìm kiếm các từ và mẫu ngôn ngữ trong danh sách từ điển chủ quan và mô hình phân loại khách quan (objective classifier), xác định các câu không chứa bất kỳ chỉ báo chủ quan nào. Các câu được gán nhãn tự động bởi hai mô hình phân loại này tạo thành tập huấn luyện mở rộng theo vòng lặp, qua đó tăng độ phủ trong khi duy trì độ chính xác cao. Hạn chế chính của hướng tiếp cận này là giả thiết rằng sự vắng mặt của dấu hiệu chủ quan đồng nghĩa với tính khách quan, dẫn đến lỗi dương tính giả đối với câu khách quan.

1.2.3.3. Khai phá quan điểm cấp độ khía cạnh

So với phân tích tình cảm ở cấp độ tài liệu và cấp độ câu, phân tích cảm xúc ở cấp độ khía cạnh hoặc phân tích tình cảm dựa trên khía cạnh sẽ chi tiết hơn. Nhiệm vụ của nó là trích xuất và tổng hợp các quan điểm về các thực thể và các khía cạnh/đặc trưng của các thực thể, còn gọi là mục tiêu. Toàn bộ nhiệm vụ của phân tích cảm tính dựa trên khía cạnh bao gồm một số nhiệm vụ phụ như trích xuất khía cạnh, trích xuất thực thể và phân loại tình cảm theo khía cạnh của thực thể hoặc theo thực thể. Khảo sát hệ thống chuyên biệt về phân tích tình cảm theo khía cạnh tiếng Việt của [188] tổng hợp trên 200 công trình, cho thấy phân tích tình cảm dựa trên khía cạnh (ABSA) tiếng Việt còn ở giai đoạn đầu với nhiều khoảng trống về dữ liệu và phương pháp, đặc biệt thiếu cách tiếp cận khai thác cấu trúc đa chiều (thực thể, khía cạnh, tình cảm, người phát biểu, thời điểm). Đặc biệt với dữ liệu tiếng Việt, ngôn ngữ luận án tập trung trong bộ dữ liệu xử lý ngôn ngữ và tiếng nói tiếng Việt (VLSP - Vietnamese Language and Speech Processing) 2018, khảo sát và so sánh toàn diện của [188][189] về tinh chỉnh hiệu năng các mô hình ngôn ngữ tiền huấn luyện trên dữ liệu phân tích tình cảm tiếng Việt cho thấy các mô hình đơn ngữ tiếng Việt như mô hình BERT được huấn luyện trước dành cho tiếng Việt (PhoBERT-Pre-trained BERT model for Vietnamese) vượt trội mô hình đa ngữ. Tuy nhiên, việc chỉ tinh chỉnh mô hình ngôn

ngữ tiền huấn luyện trên dữ liệu tiếng Việt vẫn chưa khai thác được cấu trúc đặc trưng bộ năm quan điểm tiếng Việt.

Khai phá quan điểm cấp độ khía cạnh tập trung vào xác định chi tiết của văn bản. Ý tưởng khai phá quan điểm dựa trên khía cạnh lần đầu tiên được gọi là khai phá quan điểm dựa trên đặc trưng. Mục tiêu của cấp độ phân tích khía cạnh [1] là khám phá các cảm xúc trên thực thể hoặc khía cạnh của thực thể. Trên cơ sở của phân tích này có thể đưa ra một bản tổng hợp các quan điểm về thực thể và các khía cạnh của thực thể, nhiệm vụ cơ bản là trích xuất và tổng hợp quan điểm được thể hiện trên các thực thể và các khía cạnh của thực thể. Phân tích cấp độ khía cạnh là cần thiết trong các ứng dụng và hầu như tất cả các hệ thống khai phá quan điểm đời thực trong ngành đều dựa trên cấp độ phân tích này. Khai phá quan điểm ở cấp độ khía cạnh được thực hiện trên ba nhiệm vụ sau:

(1) Xác định và trích xuất các thực thể trong văn bản đánh giá.

(2) Xác định và trích xuất các khía cạnh của các thực thể.

(3) Xác định phân cực tình cảm trên thực thể hoặc khía cạnh của thực thể.

Ngoài ra, khảo sát mới của [177][78] mở rộng phân tích sang bài toán trích xuất bộ tứ quan điểm (quadruple extraction) gồm khía cạnh, danh mục, từ quan điểm và cực tính tình cảm, kể cả với khía cạnh và quan điểm ẩn (implicit), đồng thời đánh giá khả năng của mô hình ChatGPT [127] trong nhiệm vụ này, cho thấy mô hình ngôn ngữ lớn vẫn còn hạn chế với các quan hệ ngầm định phức tạp.

Khai phá quan điểm là một nhiệm vụ quan trọng trong việc hiểu ngôn ngữ tự nhiên. Khai phá quan điểm xử lý tính toán các quan điểm và tính chủ quan của văn bản. Khai phá quan điểm nhận sự quan tâm sâu rộng từ giới học thuật và ngành công nghiệp. Khai phá quan điểm ở mức độ bình thường ngày càng không thể đáp ứng được nhu cầu ra quyết định của mọi người và với các nhiệm vụ phức tạp ở cấp độ khía cạnh, học máy thuần túy vẫn chưa thể đạt được sự hiểu biết ngôn ngữ thực sự một cách linh hoạt. Hiện tại, độ chính xác của phân tích tình cảm do máy cung cấp kém hơn rất nhiều so với con người. Để đạt được hiệu suất giống như con người đối với máy cần phải kết hợp kiến thức thông thường của con người và kiến thức lĩnh vực, cũng như các định nghĩa cơ bản về các khái niệm, để máy có thể hiểu được ngôn ngữ tự nhiên ở mức độ sâu hơn. Những điều này, kết hợp với các quy tắc lý luận quan điểm để bổ sung thông tin có thể giải thích được, sẽ có hiệu quả trong việc cải thiện hiệu suất khai phá quan điểm.

1.2.4. Các tác vụ khai phá quan điểm

Cho tập hợp các tài liệu quan điểm *D*, khai phá quan điểm gồm tám tác vụ chính sau [1]:

Tác vụ 1. *(Trích xuất và phân giải thực thể).* Trích xuất tất cả biểu thức thực thể trong tập dữ liệu *D* và nhóm các biểu thức đồng nghĩa thành các cụm thực thể (hoặc danh mục). Mỗi cụm biểu thức thực thể đại diện cho một thực thể duy nhất *e*.

Tác vụ 2. *(Trích xuất và phân giải khía cạnh).* Trích xuất tất cả biểu thức khía cạnh của các thực thể và nhóm các biểu thức này thành các cụm. Mỗi cụm biểu thức khía cạnh đại diện cho một khía cạnh duy nhất *a*.

Tác vụ 3. *(Trích xuất và phân giải chủ thể nêu quan điểm).* Trích xuất biểu thức chủ thể của mỗi quan điểm từ văn bản hoặc dữ liệu có cấu trúc và nhóm chúng lại. Tác vụ này tương tự như tác vụ 1 và 2.

Tác vụ 4. *(Trích xuất và chuẩn hóa thời gian).* Trích xuất thời gian đăng tải của mỗi quan điểm và chuẩn hóa các định dạng thời gian khác nhau.

Tác vụ 5. *(Phân loại/hồi quy tình cảm theo khía cạnh)*. Xác định xem một quan điểm về khía cạnh *a* (hoặc thực thể *e*) là tích cực, tiêu cực hay trung lập (phân loại), hoặc gán điểm số tình cảm bằng số cho khía cạnh (hoặc thực thể) đó (hồi quy).

Tác vụ 6. *(Tạo bộ năm thành tố quan điểm).* Tạo ra tất cả các bộ năm thành tố quan điểm *(e, a, s, h, t)* được biểu đạt trong *D* dựa trên kết quả từ tác vụ 1-5.

Trong trường hợp phân tích nâng cao, có thể thực hiện thêm hai tác vụ sau (tương tự tác vụ 2):

Tác vụ 7. *(Trích xuất và phân giải lý do quan điểm)*. Trích xuất các biểu thức lý do cho mỗi quan điểm và nhóm tất cả các biểu thức lý do thành các cụm. Mỗi cụm đại diện cho một lý do duy nhất của quan điểm.

Tác vụ 8. *(Trích xuất và phân giải định lượng quan điểm)*. Trích xuất các biểu thức định lượng cho mỗi quan điểm và nhóm tất cả các biểu thức định lượng thành các cụm. Mỗi cụm đại diện cho một định lượng duy nhất của quan điểm.

Ví dụ: minh họa các tác vụ (các câu được đánh số ID) và kết quả phân tích của đánh giá.

Đánh giá A*: Đăng bởi bigJohn Ngày: 15 tháng 9 năm 2011*

(1) I bought a Samsung camera and my friend brought a Canon camera yesterday. (2) In the past week, we both used the cameras a lot. (3) The photos from my Samy are not clear for night shots, and the battery life is short too. (4) My friend

was very happy with his camera and loves its picture quality. (5) I want a camera that can take good photos. (6) I am going to return it tomorrow.

Tác vụ 1. Trích xuất các biểu thức thực thể *Samsung*, *Samy, Canon* và nhóm *Samsung* với *Samy* vì chúng đại diện cho cùng một thực thể.

Tác vụ 2. Trích xuất biểu thức khía cạnh *picture*, *photo*, *battery life* và nhóm *picture* với *photo* vì chúng là từ đồng nghĩa cùng chỉ ảnh chụp.

Tác vụ 3. Chủ thể quan điểm trong câu 3 là *bigJohn (tác giả blog)* và chủ thể quan điểm trong câu 4 là *bạn của bigJohn*.

Tác vụ 4. Xác định thời gian đăng blog là *ngày 15 tháng 9 năm 2011.*

Tác vụ 5. Xác định câu 3 đưa ra quan điểm tiêu cực về chất lượng của máy Samsung và quan điểm tiêu cực về thời lượng pin. Câu 4 đưa ra quan điểm tích cực về máy Canon nói chung và chất lượng ảnh của nó. Câu 5 thể hiện quan điểm trung lập. Để tạo ra bộ năm thành tố quan điểm cho câu 4, cần biết "his camera" và "its" tham chiếu đến máy Canon.

Tác vụ 6. Tạo ra các bộ năm thành tố quan điểm sau:

1. (Samsung, picture_quality, negative, bigJohn, 15-09-2011)

2. (Samsung, battery_life, negative, bigJohn, 15-09-2011)

3. (Canon, GENERAL, positive, bigJohn's_friend, Sept-15-2011)

4. (Canon, picture_quality, positive, bigJohn's_friend, Sept-15-2011)

Ngoài ra, để khai phá và phân tích nâng cao hơn, cần tìm ra lý do và định lượng của các quan điểm. "*None"* có nghĩa là không xác định.

1. (Samsung, picture_quality, negative, bigJohn, 15-09-2011)

Lý do*: ảnh không rõ nét.*

Định lượng: *chụp ban đêm.*

2. (Samsung, battery_life, negative, bigJohn, Sept-15-2011)

Lý do*: tuổi thọ pin ngắn.*

Định lượng: *none.*

3. (Canon, GENERAL, positive, bigJohn's_friend, Sept-15-2011)

Lý do: *none.*

Định lượng: *none.*

4. (Canon, picture_quality, positive, bigJohn's_friend, Sept-15-2011)

Lý do: *none.*

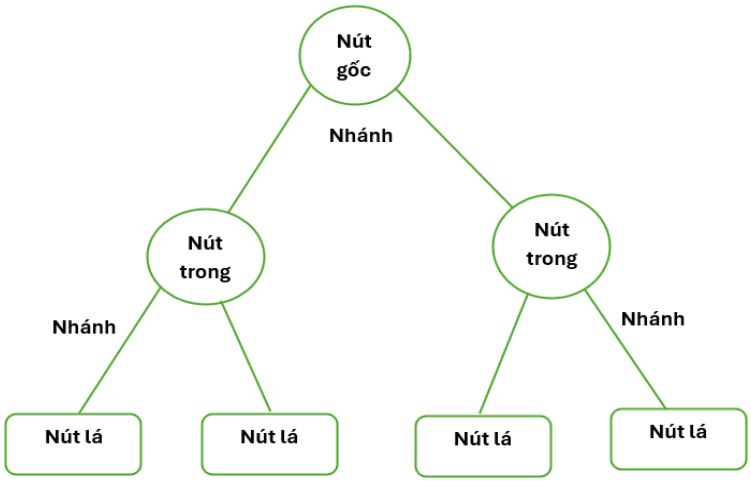
Định lượng: *none.*

1.3. Học quần thể

1.3.1. Cây quyết định

Cây quyết định [59][60] là thuật toán học máy có giám sát được sử dụng cho các bài toán phân loại và hồi quy. Thuật toán cây quyết định là thuật toán học có giám sát sử dụng cấu trúc dạng cây để đưa ra quyết định. Quy trình học của cây quyết định bao gồm một loạt các bước trong đó dữ liệu được chia thành các tập con đồng nhất, như Hình 1.1. Nút gốc, là điểm bắt đầu của cây, đại diện cho toàn bộ tập dữ liệu. Thuật toán xác định đặc trưng và ngưỡng dẫn đến sự phân chia tốt nhất dựa trên một tiêu chí cụ thể. Quá trình này tiếp tục đệ quy, với mỗi tập hợp con dữ liệu được phân chia thêm ở mỗi nút con. Điều này tiếp tục cho đến khi đạt đến tiêu chí dừng, thường là khi các nút thuần túy, nghĩa là tất cả các điểm dữ liệu trong một nút thuộc về cùng một lớp hoặc khi đạt đến độ sâu được xác định trước của cây. Các nút ở nơi cây kết thúc, được gọi là nút lá hoặc nút cuối, đại diện cho kết quả hoặc nhãn lớp. Quyết định phân chia tại mỗi nút được thực hiện bằng cách sử dụng các công thức toán học như độ lợi thông tin, tạp chất Gini hoặc giảm phương sai.

Ngoài ra, hiệu quả của kỹ thuật cây quyết định phụ thuộc vào nhiều yếu tố ảnh hưởng trực tiếp đến hiệu suất, khả năng diễn giải và tính ứng dụng của mô hình đối với các bài toán khác nhau. Các yếu tố quan trọng bao gồm chất lượng dữ liệu, độ sâu của cây, tiêu chí phân chia nút và phương pháp tỉa cây. Trong đó, chất lượng dữ liệu huấn luyện đóng vai trò then chốt, vì dữ liệu càng đầy đủ, chính xác và nhất quán thì mô hình cây quyết định càng đạt được độ chính xác và khả năng dự đoán cao hơn.



Hình 1.1: Cây quyết định

Mô hình cây quyết định có cấu trúc dạng cây ở Hình 1.1 [92], trong đó mỗi nút đại diện cho một đặc trưng, mỗi nhánh đại diện cho một quyết định và mỗi nút lá đại diện cho kết quả. Thuật toán sử dụng một loạt câu lệnh nếu - thì để tạo cây quyết định tối ưu có thể được sử dụng để đưa ra dự đoán hoặc quyết định. Đây là một trong những thuật toán được sử dụng rộng rãi nhất trong khai thác dữ liệu và học máy.

1.3.1.1. Cây hồi quy

Trong không gian đặc trưng, có 2 bước để xây dựng một cây hồi quy [91]:

Bước 1. Chia không gian dự đoán - nghĩa là tập hợp các giá trị có thể cho X_1, $X_2, \dots, X_p$ – thành J vùng riêng biệt và không chồng lấp $R_1, R_2, \dots, R_J$.

Bước 2. Đối với mọi quan sát rơi vào vùng R_j, giá trị dự đoán được xác định là giá trị trung bình của các giá trị phản hồi của các quan sát huấn luyện thuộc vùng R_j.

Xây dựng các vùng $R_1, R_2, \dots, R_J$. Về lý thuyết, các vùng có thể có hình dạng bất kỳ. Tuy nhiên, phương pháp chia không gian dự báo thành các vùng thường được sử dụng nhằm đơn giản hóa mô hình và tăng khả năng diễn giải kết quả dự đoán. Mục tiêu là tìm các vùng $R_1, R_2, \dots, R_J$ giảm thiểu tổng bình phương còn lại (RSS - Residual Sum of Squares), được đưa ra bởi công thức (1.1).

$$\sum_{j=1}^{J} \sum_{i \in R_j} \left(y_i - \hat{y}_{R_j} \right)^2 \tag{1.1}$$

Trong đó $\hat{y}_{R_j}$ là phản hồi trung bình cho các quan sát huấn luyện trong vùng thứ j. Thực hiện tiếp cận tham lam từ trên xuống được gọi là phân tách nhị phân đệ quy.

Để thực hiện tách nhị phân đệ quy, trước tiên, chọn bộ dự đoán X_j và điểm cắt s sao cho việc chia không gian dự đoán thành các vùng $\{X | X_j < s\}$ và $\{X | X_j \geq s\}$ dẫn đến mức giảm RSS lớn nhất. Xem xét tất cả các yếu tố dự đoán X_1, $X_2, \dots, X_p$ và tất cả các giá trị có thể có của các điểm cắt s đối với mỗi yếu tố dự đoán. Sau đó, chọn bộ dự đoán và điểm cắt sao cho cây kết quả có RSS thấp nhất. Cụ thể, với bất kỳ j và s nào, cặp nửa mặt phẳng được xác định trong công thức (1.2).

$$R_1(j, s) = \{X | X_j < s\} \text{ và } R_2(j, s) = \{X | X_j \geq s\} \tag{1.2}$$

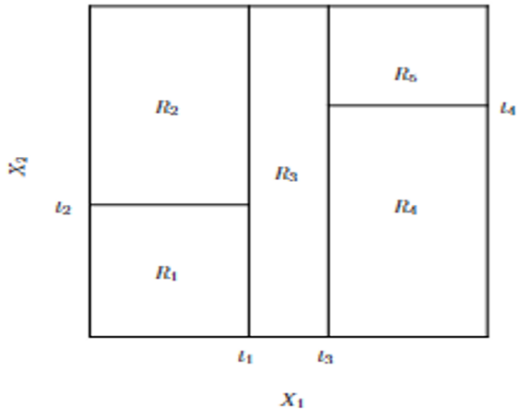
và tìm kiếm giá trị của j và s để giảm thiểu phương trình

$$\sum_{i:\, x_i \in R_1(j,s)} \left(y_i - \hat{y}_{R_1}\right)^2 + \sum_{i:\, x_i \in R_2(j,s)} \left(y_i - \hat{y}_{R_2}\right)^2 \tag{1.3}$$

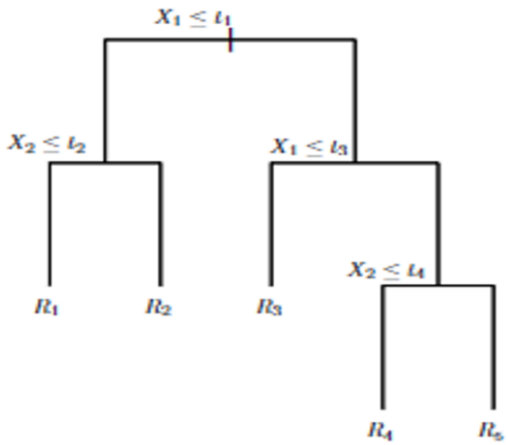
Trong đó $\hat{y}_{R_1}$ là dự đoán trung bình cho các quan sát huấn luyện trong $R_1(j, s)$ và $\hat{y}_{R_2}$ là dự đoán trung bình cho các quan sát huấn luyện trong $R_2(j, s)$.

Tìm các giá trị của j và s để giảm thiểu công thức (1.3) có thể thực hiện khá nhanh, nhất là khi số đặc trưng p không quá lớn.

Khi các vùng $R_1, R_2, \ldots, R_J$ được tạo, dự đoán phản hồi cho một quan sát thử nghiệm nhất định bằng cách sử dụng giá trị trung bình của các quan sát huấn luyện trong vùng mà quan sát thử nghiệm đó thuộc vùng đang xét. Ví dụ về năm vùng của phương pháp này được thể hiện trong Hình 1.2 [91], Hình 1.3 [91].



Hình 1.2: Phân vùng trong không gian đặc trưng hai chiều



Hình 1.3: Cây quyết định sau khi phân vùng

1.3.1.2. Cây phân loại

Cây phân loại [59][60] giống như cây hồi quy nhưng được sử dụng để dự đoán một phản ứng định tính hơn là một phản ứng định lượng. Đối với cây phân loại, việc dự đoán mỗi quan sát thuộc về loại quan sát huấn luyện thường xảy ra nhất trong vùng mà nó thuộc về. Khi giải thích kết quả của cây phân loại, thông thường không chỉ quan tâm đến dự đoán lớp tương ứng với một vùng nút lá cụ thể mà còn quan tâm đến tỷ lệ lớp giữa các quan sát huấn luyện rơi vào vùng đó.

Nhiệm vụ xây dựng cây phân loại tương tự như xây dựng cây hồi quy, đều sử dụng phân tách nhị phân đệ quy để phát triển cây. Tuy nhiên, trong bài toán phân loại, RSS không thể được dùng làm tiêu chí phân tách. Thay vào đó, chỉ số Entropy được sử dụng để đánh giá chất lượng phân tách, được xác định theo công thức (1.4):

$$H = -\sum_{k=1}^{K} p_{mk} \log p_{mk} \tag{1.4}$$

Vì $0 \leq p_{mk} \leq 1$, nên $0 \leq -p_{mk} \log p_{mk}$. Ngoài ra, có thể chỉ ra rằng Entropy sẽ nhận một giá trị gần bằng 0 nếu các p_{mk} đều gần bằng 0 hoặc gần bằng 1.

1.3.2. Bootstrap

Bootstrap [13] là một kỹ thuật thống kê được sử dụng để ước tính phân phối lấy mẫu của một công cụ ước tính bằng cách lấy mẫu lại một tập dữ liệu đã cho có hoàn lại. Nó là một công cụ mạnh mẽ để ước tính độ chính xác của một mô hình hoặc để xây dựng khoảng tin cậy. Phương pháp bootstrap hoạt động bằng cách lấy mẫu ngẫu nhiên từ tập dữ liệu gốc có hoàn lại, tạo một tập dữ liệu mới có cùng kích thước. Quá trình này được lặp lại nhiều lần và các kết quả sau đó được tính trung bình để đưa ra ước tính về phân phối lấy mẫu.

Ví dụ trên Hình 1.4 [91], với các mẫu ban đầu là 1, 2, 3

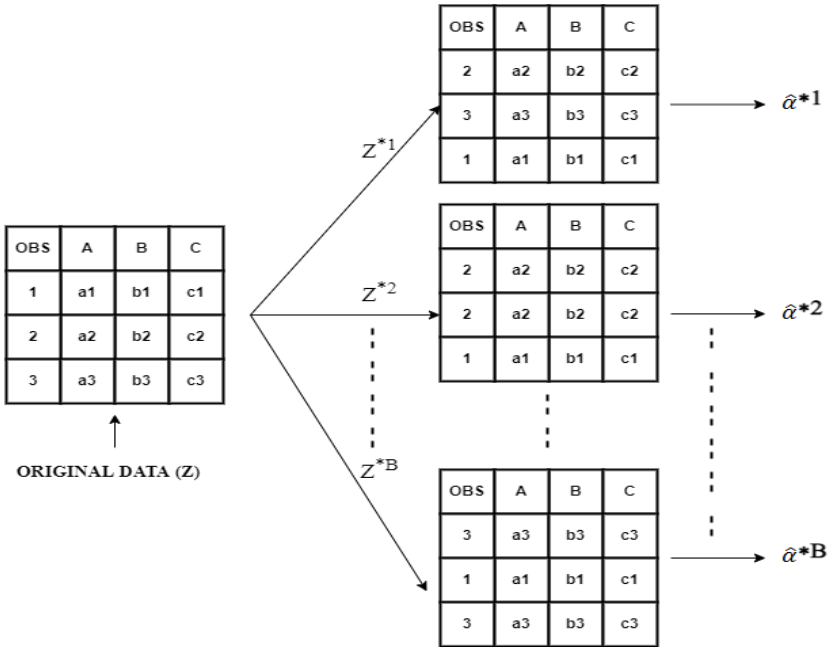
Mẫu 1 có các giá trị của A là a1, giá trị của B là b1, giá trị của C là c1.

Mẫu 2 có các giá trị của A là a2, giá trị của B là b2, giá trị của C là c2.

Mẫu 3 có các giá trị của A là a3, giá trị của B là b3, giá trị của C là c3.

Áp dụng phương pháp bootstrap, kết quả đầu tiên là 2; 3; 1 tương ứng với các giá trị: a2, b2, c2; a3, b3, c3; a1, b1, c1; và trả mẫu. Tiếp tục lấy mẫu lần 2, kết quả lần 2: 2; 2; 1 tương ứng với các giá trị: a2, b2, c2; a2, b2, c2; a1, b1, c1; và trả mẫu. Tiếp tục lấy mẫu lần B, kết quả lần B: 3; 1; 3 tương ứng với các giá trị a3, b3, c3; a1,

b1, c1; a3, b3, c3. Từ bootstrap Z^{*1}, có thể tạo ước tính bootstrap mới cho giá trị giảm rủi ro α được gọi là α^{*1}.



Hình 1.4: Bootstrap

1.3.3. Các kỹ thuật học quần thể

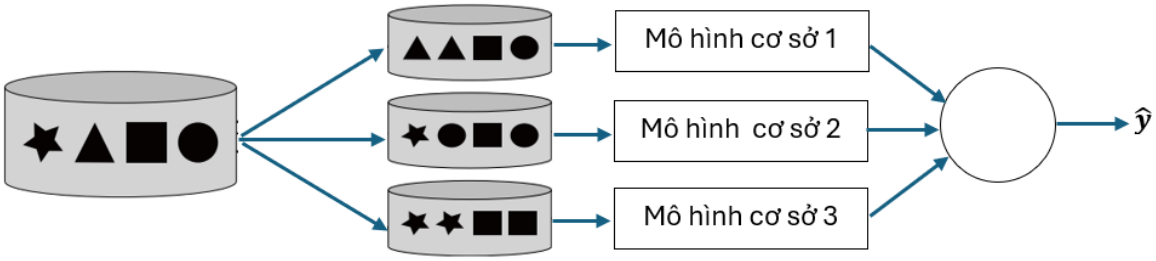
1.3.3.1. Bagging

Phương pháp Bagging, hay tổng hợp bootstrap, là một phương pháp học tập kết hợp được phát triển lần đầu bởi [62]. Kỹ thuật này bao gồm việc tạo ra một tập hợp (nhóm hoặc rừng) các cây quyết định, mỗi cây được huấn luyện dựa trên một tập dữ liệu huấn luyện khác nhau. Các tập dữ liệu này được tạo ra bằng phương pháp lấy mẫu bootstrap, bao gồm việc lấy mẫu đồng nhất và có hoàn trả từ tập dữ liệu gốc. Một tập dữ liệu bootstrap được kỳ vọng sẽ có một phần 1−1e (≈ 63,2%) mẫu duy nhất, phần còn lại bao gồm các mục trùng lặp [62]. Một dự đoán được đưa ra dựa trên tập hợp này bằng cách lấy trung bình kết quả của tất cả các cây quyết định (đối với hồi quy) hoặc bằng cách bỏ phiếu (đối với phân loại). Phương pháp Bagging thường được thực hiện bằng cách sử dụng các bộ học cơ sở cùng loại, do đó được coi là một tập hợp đồng nhất. Hơn nữa, mỗi mô hình được xây dựng độc lập. Nghiên cứu hệ thống và so sánh toàn diện nhất gần đây về các kỹ thuật học quần thể cho phân tích tình cảm trên mạng xã hội của [164] kết luận rằng các kỹ thuật ensemble dựa trên Bagging (bao gồm Random Forest) vượt trội so với các kỹ thuật Boosting trên bốn tập dữ liệu chuẩn, đo trên tỷ lệ dương tính đúng (TPR - True Positive Rate), F-

Measure (F1-Score), Precision, Recall và diện tích dưới đường cong ROC (ROC-AUC - Area Under the Receiver Operating Characteristic Curve). Kết quả này phù hợp với lựa chọn xây dựng mô hình rừng phân loại dựa trên Bagging.

Để cung cấp một tập con dữ liệu đủ lớn cho mỗi bộ học cơ sở, phương pháp lấy mẫu bootstrap được sử dụng, bao gồm việc lấy mẫu đồng nhất có hoàn trả từ tập dữ liệu gốc cho đến khi một tập dữ liệu mới có cùng kích thước với tập dữ liệu gốc được xây dựng, chứa một số điểm dữ liệu lặp lại. Sau đó, đầu ra của mỗi bộ học cơ sở được tổng hợp bằng cách lấy trung bình đơn giản hoặc bỏ phiếu đa số.

Quá trình này được minh họa sơ đồ trong Hình 1.5 [93] trong trường hợp ba mô hình cơ sở, trong đó các hình trụ đại diện cho các kho dữ liệu, các ký hiệu chứa trong các kho dữ liệu này đại diện cho các mẫu huấn luyện khác nhau và vòng tròn màu trắng đại diện cho sự tổng hợp các đầu ra của mô hình bằng cách sử dụng trọng số. Như có thể thấy rõ từ hình, mỗi mẫu bootstrap chứa một hoặc hai phần tử lặp lại từ tập dữ liệu gốc và đại diện cho các phân bố khác nhau của cùng một dữ liệu. Trong cấu hình trong hình, mô hình cơ sở 3 có khả năng phân biệt chính xác hơn giữa các ngôi sao và hình vuông, trong khi mô hình cơ sở 1 và 2 có khả năng xác định tốt hơn các hình tam giác và hình tròn tương ứng. Các mô hình được kết hợp với mục đích khai thác các thế mạnh bổ sung của các mô hình để dự đoán chính xác vectơ đầu ra $\hat{y}$.



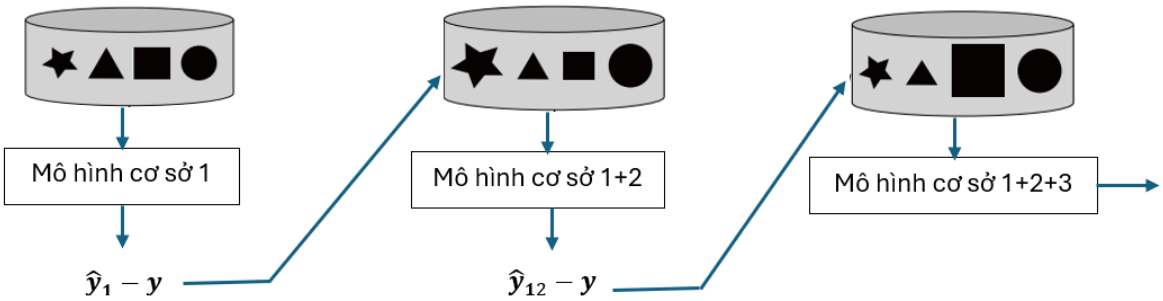
Hình 1.5: Sơ đồ phương pháp Bagging với ba mô hình cơ sở

1.3.3.2. Boosting

Phương pháp Boosting là một quá trình, được nêu ra lần đầu tiên trong [94], mà qua đó các bộ học yếu có thể được chuyển đổi thành bộ học mạnh. Giống như Bagging, tính đa dạng được đảm bảo bằng thao tác đầu vào, theo đó mỗi bộ học cơ sở được huấn luyện trên một tập dữ liệu được phân bổ khác nhau. Tuy nhiên, không giống như Bagging, Boosting là một quá trình tuần tự và phụ thuộc. Cụ thể hơn, tập

dữ liệu cho mỗi bộ học cơ sở mới được tạo ra để nhắm đến các lỗi do bộ học cơ sở trước đó mắc phải. Quy trình Boosting chung được minh họa trong Hình 1.6 [93] bằng một ví dụ với ba mô hình cơ sở.

Mô hình cơ sở 1 được huấn luyện dựa trên dữ liệu huấn luyện ban đầu. Sau đó, sai số của nó được xác định bằng sự khác biệt giữa kết quả dự đoán $\hat{\boldsymbol{y}}$ và nhãn thực $\boldsymbol{y}$ được quan sát. Trong ví dụ này, mô hình được phát hiện phân loại sai các trường hợp thuộc lớp hình sao và hình tròn. Phân bố dữ liệu sau đó được điều chỉnh để trọng số hóa nhiều hơn cho các lớp bị phân loại sai, như được chỉ ra bởi các hình dạng lớn hơn trong phân bố dữ liệu thứ hai. Một mô hình cơ sở thứ hai sau đó được huấn luyện dựa trên phân bố mới, và các dự đoán của nó được trọng số hóa với các dự đoán của Mô hình cơ sở 1 để tạo thành một tập hợp hai mô hình ban đầu. Kết quả của tập hợp này sau đó được sử dụng để hướng dẫn phân bố dữ liệu tiếp theo, và quá trình này được lặp lại một số lần nhất định. Bằng cách thêm lặp đi lặp lại các mô hình mới nhằm giải quyết những điểm yếu của các mô hình trước đó, các mô hình yếu được tăng cường để tạo thành một mô hình mạnh duy nhất.



Hình 1.6: Sơ đồ phương pháp Boosting với ba mô hình cơ sở

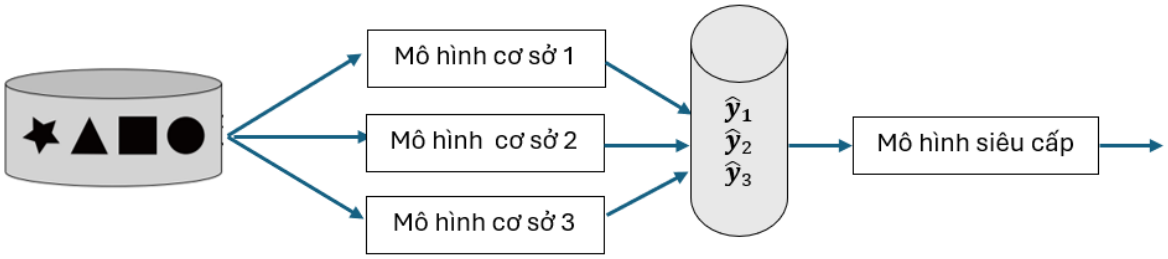
Các thuật toán mạnh mẽ của Boosting bao gồm: Adaptive Boosting (AdaBoost), Gradient Boosting và Extreme gradient boosting (XGBoost).

1.3.3.3. Stacking

Stacking [64][63] là phương pháp kết hợp trong đó mô hình siêu cấp được huấn luyện để kết hợp một số mô hình cơ sở độc lập. Phương pháp Stacking được minh họa trong Hình 1.7, trong đó cùng một bộ dữ liệu được sử dụng để huấn luyện những mô hình cơ sở (được biểu thị bằng các sắc thái xám khác nhau). Tương tự như với các mô hình cơ sở trong quần thể, bất kỳ thuật toán học nào cũng có thể được sử

dụng làm thuật toán siêu học. Tuy nhiên, thông thường, một số dạng hồi quy tuyến tính hoặc logistic được áp dụng.

Như hình minh họa Hình 1.7 [93], thuật toán học siêu cấp được huấn luyện trên một tập dữ liệu mới được hình thành từ các đầu ra dự đoán $\boldsymbol{y}$ của mỗi mô hình học cơ sở $i \in \{1, 2, \dots, K\}$ để trả về một dự đoán tổng hợp $\boldsymbol{y}$. Nếu các dự đoán tạo thành dữ liệu huấn luyện cho thuật toán học siêu cấp được thực hiện trên cùng một dữ liệu được sử dụng để huấn luyện các mô hình cơ sở, sẽ có nguy cơ quá khớp cao. Do đó, phương pháp kiểm định chéo k-fold thường được áp dụng, trong đó các mô hình học cơ sở được huấn luyện trên $k-1$ fold và các dự đoán trên fold thứ k được đưa vào tập dữ liệu huấn luyện cho thuật toán học siêu cấp. Quá trình này được lặp lại k lần, tạo ra một tập huấn luyện mới có cùng kích thước với tập dữ liệu gốc. Sau đó, các mô hình cơ bản sẽ được huấn luyện lại dựa trên toàn bộ tập dữ liệu huấn luyện để sử dụng trong giai đoạn triển khai.



Hình 1.7: Sơ đồ phương pháp Stacking với ba mô hình cơ sở

1.3.4. Mô hình học quần thể với các mô hình phân loại cơ sở giống nhau

1.3.4.1. Rừng phân loại

Rừng phân loại [65][66] là thuật toán học quần thể với mô hình cơ sở giống nhau là cây phân loại và thêm bước chọn m_{try} đặc trưng ngẫu nhiên tại mỗi nút phân tách, tạo sự đa dạng giữa các cây. Tổng hợp dự đoán các cây trong rừng có tính khái quát tốt.

Rừng phân loại được huấn luyện bằng cách tối ưu hóa các tham số của các cây phân loại tại mỗi nút phân tách j thông qua hàm tối ưu công thức (1.5).

$$\theta_j^* = \underset{\theta_j \in \tau_j}{\operatorname{argmax}} I_j \tag{1.5}$$

Trong đó I_j là hàm mục tiêu đo lường mức tăng thông tin và được tính bằng công thức (1.6).

$$I_j = H\left(S_j\right) - \sum_{i \in \{L,R\}} \frac{\left|S_j^i\right|}{\left|S_j\right|} H\left(S_j^i\right) \tag{1.6}$$

Với i lập chỉ mục hai nút con. Entropy của tập mẫu S gồm các mẫu huấn luyện được định nghĩa theo công thức (1.7):

$$H(S) = -\sum_{c \in C} p(c) \log p(c) \tag{1.7}$$

Trong đó, $p(c)$ được tính là xác suất thực nghiệm chuẩn hóa của nhãn tương ứng với các điểm huấn luyện trong S. Cây phân loại được huấn luyện bằng cách tối đa hóa mức tăng thông tin có xu hướng tạo ra cây trong đó entropy của phân phối lớp liên quan đến các nút sẽ giảm, có nghĩa là độ tin cậy dự đoán tăng khi đi từ gốc đến phía lá.

Thử nghiệm rừng phân loại bằng cách cho dữ liệu thử nghiệm đầu vào là một mẫu dữ liệu $s = (x_1, x_2, x_3, \ldots, x_d)$, với d các yếu tố dự đoán và đầu ra là nhãn lớp c sao cho $c \in C$, với $C = \{c_k\}$, thông qua phân phối xác suất $p(c|s)$.

1.3.4.2. Rừng mật độ

Rừng mật độ [66][68][69][70] là một thuật toán học máy không giám sát, sử dụng cấu trúc dựa trên cây để xác định các cụm điểm dữ liệu có đặc điểm tương tự. Nó hoạt động bằng cách xây dựng cấu trúc cây từ các điểm dữ liệu, sau đó sử dụng mật độ của các điểm dữ liệu trong mỗi nhánh của cây để xác định cụm nào sẽ được hình thành. Thuật toán hữu ích cho việc phân cụm các điểm dữ liệu có nhiều đặc điểm và có thể được sử dụng để xác định các cụm điểm dữ liệu tương tự trong tập dữ liệu lớn. Rừng mật độ là thuật toán học quần thể bao gồm các cây phân cụm được huấn luyện ngẫu nhiên. Các lá của cây chứa các mô hình dự đoán như Gaussian.

Trong không gian vectơ đặc trưng, cho một tập hợp các điểm $\{v\}$ không được dán nhãn, có thể huấn luyện từng cây riêng lẻ trong rừng theo cách song song và sử dụng tối ưu hóa nút ngẫu nhiên thứ j thông qua hàm tối ưu trong công thức (1.5) và hàm mục tiêu đo lường mức tăng thông tin được tính trong công thức (1.6).

Để xác định mô hình rừng mật độ, cần xác định dạng chính xác Entropy $H(S)$ của một tập hợp các điểm huấn luyện S. Trong trường hợp rừng mật độ, dữ liệu không được dán nhãn nên việc xác định Entropy $H(S)$ không giống như phân loại và hồi quy. Do đó, cần xác định một Entropy $H(S)$ không giám sát, tức là một

Entropy $H(S)$ áp dụng cho dữ liệu không dán nhãn. Sau đó, sử dụng các phân phối Gaussian đa biến tại các nút. Entropy của một phân phối Gaussian d chiều được tính theo công thức (1.8).

$$H(S) = \frac{1}{2}\log((2\pi e)^d|\Lambda(S)|) \tag{1.8}$$

Trong đó Λ liên quan đến ma trận hiệp phương sai $d \times d$. Do đó, mức tăng thông tin được tính theo công thức (1.9).

$$I_j = \log\left(\left|\Lambda\left(S_j\right)\right|\right) - \sum_{i\in\{L,R\}} \frac{\left|S_j^i\right|}{\left|S_j\right|}\log\left(\left|\Lambda\left(S_j^i\right)\right|\right) \tag{1.9}$$

Với $|.|$ là định thức của ma trận

Tập hợp các lá trong cây thứ t trong rừng xác định một phân vùng dữ liệu tại công thức (1.10).

$$l(\upsilon): \mathbb{R}^d \rightarrow \ L \subset \mathbb{N} \tag{1.10}$$

Trong đó, $l(\upsilon)$ biểu diễn lá xác định bởi điểm đầu vào υ và L là tập hợp tất cả các lá trong một cây nhất định. Thống kê của tất cả các điểm huấn luyện đến từng nút lá được tóm tắt bằng một phân phối Gaussian đa biến duy nhất $\mathcal{N}\left(\upsilon, \mu_{l(\upsilon)}, \Lambda_{l(\upsilon)}\right)$. Sau đó, đầu ra của cây thứ t được mô hình hóa trong công thức (1.11).

$$p_t(\upsilon) = \frac{\pi_{l(\upsilon)}}{Z_t}\mathcal{N}\left(\upsilon; \mu_{l(\upsilon)}, \Lambda_{l(\upsilon)}\right) \tag{1.11}$$

Vectơ μ_l biểu diễn giá trị trung bình của tất cả các điểm đạt tới lá l và Λ_l là ma trận hiệp phương sai liên quan. Vô hướng π_l là tỷ lệ của tất cả các điểm huấn luyện đạt tới lá l, nghĩa là $\pi_l = \frac{|S_l|}{|S_0|}$

Trong công thức (1.11), mỗi Gaussian bị cắt bớt bởi các ranh giới của ô phân vùng được liên kết với lá tương ứng. Vì vậy, để đảm bảo chuẩn hóa xác suất cần kết hợp hàm phân vùng Z_t, được định nghĩa tại công thức (1.12)

$$Z_t = \int_{\upsilon}\left(\sum_l \pi_l \mathcal{N}(\upsilon; \mu_l, \Lambda_l)\mathrm{p}(l|\upsilon)\right)d\upsilon \tag{1.12}$$

Tuy nhiên, trong rừng mật độ, mỗi điểm dữ liệu đạt đến chính xác một nút đầu cuối trên mỗi cây. Như vậy, điều kiện $p(l|\upsilon)$ là một hàm delta $p(l|\upsilon) = \delta(l = l(\upsilon))$ do đó công thức (1.12) trở thành công thức (1.13).

$$Z_t = \int_{\upsilon} \pi_{l(\upsilon)} \mathcal{N}\left(\upsilon; \mu_{l(\upsilon)}, \Lambda_{l(\upsilon)}\right) d\upsilon \tag{1.13}$$

Như thường gặp khi xử lý các mô hình sinh (generative models), việc tính Z_t trong không gian nhiều chiều có thể là một thách thức.

Đối với các bộ học yếu phức tạp hơn, có thể dễ dàng ước tính Z_t bằng tích phân số trên một lưới thông thường tại công thức (1.14).

$$Z_t \approx \Delta . \sum_i \pi_{l(\upsilon_i)} \mathcal{N}\left(\upsilon_i; \mu_{l(\upsilon_i)}, \Lambda_{l(\upsilon_i)}\right) \tag{1.14}$$

Trong đó, Δ biểu diễn độ dài, diện tích, thể tích,... tùy thuộc vào chiều của miền; các điểm υ_i được tạo trên mạng lưới thông thường hữu hạn với khoảng cách Δ.

Tương tự như phân loại và hồi quy, mật độ rừng được tính bằng trung bình cộng của tất cả các mật độ cây tại công thức (1.15).

$$p(\upsilon) = \frac{1}{T} \sum_{t=1}^{T} p_t(\upsilon) \tag{1.15}$$

Với mật độ mỗi cây là $p_t(\upsilon)$

Thử nghiệm rừng mật độ bằng cách cho mỗi điểm dữ liệu đầu vào υ là vector đa chiều của các phản hồi đặc trưng, $\upsilon = (x_1, x_2, x_3, \dots, x_d) \in \mathbb{R}^d$. Đầu ra là toàn bộ hàm mật độ xác suất $p(\upsilon) \geq 0, \quad \int p(\upsilon) d\upsilon = 1$ cho bất kỳ đầu vào υ.

1.3.4.3. Rừng đa tạp

Rừng đa tạp [66][71] là thuật toán học quần thể bao gồm các cây phân cụm. Rừng đa tạp có liên quan chặt chẽ với rừng mật độ. Tuy nhiên, không giống như rừng mật độ, mô hình rừng đa tạp có thêm các thành phần bổ sung như mô hình mối quan hệ, thuật toán để ước lượng các ánh xạ tối ưu. Hướng tiếp cận này khác với các phương pháp phân cụm đa tạp dựa trên học sâu gần đây như mạng nơ-ron tích chập kết hợp embedding [36], vốn đòi hỏi huấn luyện end-to-end và khối lượng dữ liệu lớn, trong khi rừng đa tạp đề xuất trong luận án xây dựng cấu trúc đa tạp tường minh qua ma trận quan hệ W, không cần huấn luyện mạng nơ-ron sâu.

Cho một tập hợp các quan sát không ghi nhãn $\{\upsilon_1, \upsilon_2, \dots, \upsilon_k\}$ với $\upsilon_i \in \mathbb{R}^d$, tìm một ánh xạ $f: \mathbb{R}^d \to \mathbb{R}^{d'}$, $f(\upsilon_i) = \upsilon_i'$, sao cho $d' \ll d$ và bảo toàn quan hệ khoảng

cách giữa các quan sát. Mỗi quan sát đầu vào v được biểu diễn dưới dạng vectơ đặc trưng đa chiều $v = (x_1, x_2, \dots, x_d) \in \mathbb{R}^d$.

Trong không gian vectơ đặc trưng, cho một tập hợp các điểm $\{v\}$ không được dán nhãn được sử dụng để huấn luyện rừng đa tạp. Sử dụng tối ưu hóa nút ngẫu nhiên, quá trình huấn luyện rừng đa tạp đạt được bằng cách tối đa hóa phép đo mức tăng thông tin liên tục. Do đó, việc tối ưu hóa nút tách thứ j được thực hiện tương tự như rừng mật độ và bổ sung thêm mô hình mối quan hệ khoảng cách giữa các điểm dữ liệu.

Mô hình mối quan hệ. Theo [66], một số thước đo về mối quan hệ giữa các điểm dữ liệu được ước tính nhằm duy trì khoảng cách tương đối giữa các điểm sau khi ánh xạ. Theo công thức (1.16), các lá của cây phân cụm t xác định một phân vùng của dữ liệu đầu vào.

$$l(v): \mathbb{R}^d \rightarrow \mathfrak{L} \subset \mathbb{N} \tag{1.16}$$

Với chỉ số nút lá l và tập tất cả các lá $\mathfrak{L}$ trong một cây cho trước. Đối với cây phân cụm t, tính ma trận mối quan hệ $\mathbb{W}^t$ có kích thước $\mathtt{k} \times \mathtt{k}$ với các phần tử trong công thức (1.17).

$$\mathbb{W}_{ij}^t = e^{-\mathbb{Q}^t(v_i, v_j)} \tag{1.17}$$

trong đó $\mathbb{Q}$ là khoảng cách được xác định bởi quan hệ nhị phân của cặp (v_i, v_j) trong công thức (1.18).

$$\mathbb{Q}^t(v_i, v_j) = \begin{matrix} 0 & if\ l(v_i) = l(v_j) \\ \infty & otherwise \end{matrix} \tag{1.18}$$

trong đó $d_{ij} = v_i - v_j$, và $\Lambda_{l(v_i)}$ là ma trận hiệp phương sai liên kết với các lá đạt tới điểm v_i.

Trong công thức (1.18), cho một cây t và một cặp điểm (v_i, v_j) và chỉ định mối quan hệ hoàn hảo bằng 1, khoảng cách bằng 0 cho cặp (v_i, v_j), nếu hai điểm đó kết thúc trong cùng một cụm hay lá; ngược lại, mối quan hệ bằng 0 và khoảng cách bằng ∞.

Mô hình học quần thể. Ma trận mối quan hệ quan điểm đại diện cho toàn bộ khu rừng có kích thước được tính bằng cách lấy trung bình [64] trên tất cả các ma trận mối quan hệ từ mỗi cây đơn lẻ trong công thức (1.19).

$$\mathbb{W} = \frac{1}{T}\sum_{t=1}^{T} \mathbb{W}^t \tag{1.19}$$

1.4. Đánh giá mô hình

Mô hình đề xuất được đánh giá thông qua Bảng 1.1 được gọi là ma trận nhầm lẫn [202][16], bao gồm bốn kết hợp khác nhau giữa giá trị dự đoán quan điểm và giá trị thực tế quan điểm. Hiệu suất phân loại của mô hình trong trường hợp này được đánh giá là phân loại tiêu cực hoặc phân loại tích cực.

Bảng 1.1: Ma trận nhầm lẫn

	Dự đoán tích cực	**Dự đoán tiêu cực**
Thật sự tích cực	*TP*	*FN*
Thật sự tiêu cực	*FP*	*TN*

Trong đó *TP* là số lượng nhóm quan điểm được phân loại quan điểm thật sự là "Tích cực", *TN* là số lượng nhóm quan điểm được phân loại quan điểm thật sự "Tiêu cực", *FP* là số lượng nhóm quan điểm được phân loại quan điểm dự đoán "Tích cực" nhưng thực tế là "Tiêu cực", FN là số lượng nhóm quan điểm được phân loại quan điểm dự đoán "Tiêu cực" nhưng thực tế là "Tích cực".

Dựa trên ma trận nhầm lẫn trong Bảng 1.1, đánh giá bộ phân loại quan điểm thông qua các độ đo là Accuracy, Precision, Recall và F-Measure.

Accuracy là tỷ lệ dự đoán chính xác của tất cả quan điểm thật sự tích cực (TP) và quan điểm thật sự tiêu cực (TN) trên tổng số bộ năm quan điểm của các mẫu được kiểm tra được cho bởi công thức (1.20).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{1.20}$$

Precision là tỷ lệ dự đoán chính xác được tính giữa số lượng thật sự tích cực (TP) trên tổng số thật sự tích cực (TP) và tích cực giả (FP) được cho bởi công thức (1.21).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{1.21}$$

Recall là tỷ lệ dự đoán chính xác được tính giữa số lượng thật sự tích cực (TP) trên tổng số thật sự tích cực (TP) và tiêu cực giả (FN) trong công thức (1.22).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{1.22}$$

F-Measure (F1-score) hay F1 đo lường hiệu suất tổng thể của hệ thống bằng cách kết hợp hài hòa hai chỉ số Recall và Precision. Công thức (1.23) xác định độ đo F1.

$$\mathrm{F1} = \frac{2 * (\mathrm{Recall} * \mathrm{Precision})}{\mathrm{Recall} + \mathrm{Precision}}, \tag{1.23}$$

Các công thức (1.20) - (1.23) thiết lập bộ độ đo chuẩn được áp dụng nhất quán cho các thực nghiệm chính của luận án (Chương 2, 3 và 4). Trong trường hợp dữ liệu có mức mất cân bằng lớp rất cao, đặc biệt khi tỷ lệ ρ = |positive|/|negative| lớn, Accuracy và F1 tính cho lớp positive có thể phản ánh sai lệch hiệu năng. Để xử lý trường hợp này, ba độ đo [201] sau được sử dụng trong Phụ lục F, cụ thể:

F1-macro là trung bình cộng F1 của từng lớp k, cho mỗi lớp trọng số bằng nhau bất kể kích thước. Với bài toán phân loại C lớp, được xác định theo công thức (1.24):

$$\mathrm{F1-macro} = \frac{1}{\mathrm{C}} \times \sum_{\mathrm{k}=1} \mathrm{F1_k} \quad , \quad \mathrm{k} = 1, 2, \dots, \mathrm{C} \quad , \tag{1.24}$$

với:

$$\mathrm{F1}_k = \frac{2 \times Precision_k \times Recall_k}{Precision_k + Recall_k}$$

trong đó C là số lớp phân loại (trong luận án C = 2 gồm positive và negative), $\mathrm{F1_k}$ là F1 của lớp thứ k. Độ đo F1-macro không phụ thuộc vào kích thước lớp, lớp thiểu số có đóng góp ngang bằng lớp đa số.

F1-positive đo hiệu suất phát hiện đúng lớp đa số (positive, ký hiệu +). Với TP, FP, FN tính trên lớp positive:

$$\mathrm{Precision^+} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{1.25}$$

$$\mathrm{Recall^+} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \quad , \tag{1.26}$$

$$\mathrm{F1-pos} = \frac{2 \times Precision^+ \times Recall^+}{Precision^+ + Recall^+}, \tag{1.27}$$

F1-negative đo hiệu suất phát hiện đúng lớp thiểu số (negative, ký hiệu -), là độ đo quan trọng nhất khi dữ liệu mất cân bằng. Với TN, FP, FN tính trên lớp negative:

$$\text{Precision}^{-} = \frac{\text{TN}}{\text{TN} + \text{FN}} \tag{1.28}$$

$$\text{Recall}^{-} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{1.29}$$

$$\text{F1} - \text{neg} = \frac{2 \times Precision^{-} \times Recall^{-}}{Precision^{-} + Recall^{-}} \tag{1.30}$$

Quan hệ giữa 3 độ đo khi C = 2 (binary classification):

$$\text{F1} - \text{macro} = (\text{F1} - \text{pos} + \text{F1} - \text{neg}) / 2 \tag{1.31}$$

Công thức (1.31) cho thấy F1-macro là trung bình cộng của F1-positive và F1-negative, đây là lý do F1-macro phản ánh đầy đủ cả hai lớp, không thiên lệch về lớp đa số như Accuracy hay F1-weighted.

Trong nghiên cứu [201] cho thấy độ đo F1-macro ít bị chi phối bởi lớp đa số hơn Accuracy khi phân bố lớp thay đổi, do đó được chọn làm độ đo chính trong thực nghiệm trên tập dữ liệu mất cân bằng nghiêm trọng. Độ đo F1-neg bổ sung giúp đánh giá khả năng phát hiện lớp thiểu số – điều mà Accuracy và F1-pos không phản ánh được. Bên cạnh F1-macro, độ chính xác cân bằng (Balanced Accuracy, BA) được dùng làm độ đo chính trong các thực nghiệm so sánh bổ sung tại Phụ lục G–J: BA = (Recall + Specificity)/2, trong đó Specificity = TN/(TN + FP) là độ thu hồi của lớp negative. Do là trung bình độ thu hồi của hai lớp, BA không phụ thuộc tỷ lệ lớp trong tập kiểm tra; mô hình dự đoán toàn bộ về một lớp luôn có BA = 0,5.

1.5. Các nghiên cứu liên quan và đề xuất

1.5.1. Các nghiên cứu liên quan

1.5.1.1. Về kỹ thuật khai phá quan điểm

- Khai phá quan điểm dựa trên từ vựng

Các phương pháp tiếp cận dựa trên khai thác từ vựng tình cảm được xây dựng từ các từ điển hiện có hoặc được tạo ra từ kho ngữ liệu. Mục tiêu của những từ vựng này là lập chỉ mục những từ quan điểm. Các từ quan điểm được sử dụng trong nhiệm vụ phân loại tình cảm. Các từ quan điểm tích cực được sử dụng để thể hiện một số trạng thái tình cảm mong muốn. Các từ quan điểm tiêu cực được sử dụng để thể hiện một số trạng thái tình cảm không mong muốn. Các cụm từ quan điểm và các thành ngữ cũng được gọi là từ vựng quan điểm (opinion lexicon). Có ba cách tiếp cận chính

để tạo hoặc thu thập danh sách các từ quan điểm. Phương pháp thủ công là rất tốn thời gian. Nó thường được kết hợp với hai phương pháp tự động khác như một bước kiểm tra cuối cùng để tránh những sai lầm do các phương pháp tự động gây ra. Hai cách tiếp cận tự động là tiếp cận dựa trên từ điển và tiếp cận dựa trên kho ngữ liệu**.**

Trong nghiên cứu [16] đã tiếp cận phân tích tình cảm dựa trên tiếp cận từ điển. Một tập hợp nhỏ các từ quan điểm được thu thập thủ công cùng với định hướng tình cảm đã biết. Sau đó tập hợp này được phát triển bằng cách tìm kiếm trong WordNet hoặc từ điển đồng nghĩa (thesaurus) để tìm các từ đồng nghĩa và trái nghĩa. Quá trình lặp lại và dừng khi không tìm thấy từ mới. Sau khi quá trình hoàn thành, có thể tiến hành kiểm tra thủ công để loại bỏ hoặc sửa chữa các sai sót. Phương pháp dựa trên từ điển có một nhược điểm lớn là không thể tìm thấy các từ quan điểm với các định hướng cụ thể về miền và ngữ cảnh. Trong nghiên cứu [17] đã sử dụng phương pháp dựa trên từ điển để xác định các câu tình cảm trong quảng cáo theo ngữ cảnh. Chiến lược quảng cáo để cải thiện mức độ liên quan của quảng cáo và trải nghiệm người dùng được đề xuất. Phân tích cú pháp và từ điển tình cảm được sử dụng, đồng thời đề xuất phương pháp tiếp cận dựa trên quy tắc để xử lý việc trích xuất từ theo chủ đề và xác định thái độ của người tiêu dùng trong việc trích xuất từ khóa quảng cáo. Kết quả thực nghiệm đã chứng minh hiệu quả của cách tiếp cận được đề xuất trong việc trích xuất từ khóa quảng cáo và lựa chọn quảng cáo.

Phương pháp dựa trên kho ngữ liệu giúp giải quyết vấn đề tìm kiếm các từ quan điểm với các định hướng cụ thể theo ngữ cảnh. Các phương pháp của nó phụ thuộc vào các mẫu hoặc mẫu cú pháp xảy ra cùng với danh sách các từ quan điểm để tìm các từ quan điểm khác trong một kho ngữ liệu lớn. Một phương pháp tiêu biểu theo hướng này thiết kế danh sách các tính từ quan điểm gốc và sử dụng chúng cùng với tập hợp các ràng buộc ngôn ngữ để xác định các từ quan điểm bổ sung và định hướng của chúng. Các ràng buộc dành cho các từ kết nối như: AND, OR, BUT, EITHER - OR, … Ví dụ từ AND thể hiện các tính từ liền nhau thường có cùng hướng ngữ nghĩa. Ý tưởng này có tính nhất quán theo cảm tính, không phải lúc nào cũng nhất quán trên thực tế. Cũng có những biểu hiện mâu thuẫn, chẳng hạn như từ: but, however được thể hiện sự thay đổi quan điểm. Để xác định xem hai tính từ liền nhau có cùng hướng ngữ nghĩa hay khác nhau, việc học được áp dụng cho một kho ngữ

liệu lớn. Các liên kết giữa các tính từ tạo thành một đồ thị và việc phân cụm được thực hiện trên đồ thị để tạo ra hai tập hợp từ: tích cực và tiêu cực.

Phương pháp trường ngẫu nhiên có điều kiện (CRFs - Conditional Random Fields) [18] được sử dụng như một kỹ thuật học trình tự để trích xuất các biểu thức có quan điểm. Nó cũng đã được sử dụng để phân biệt đối cực tình cảm bằng thuật toán so khớp mẫu. Thuật toán được áp dụng trên các bài đánh giá trực tuyến. Từ điển cảm xúc được thành lập. Thực nghiệm trên các đánh giá trực tuyến về xe hơi, khách sạn và máy tính. Kết quả cho thấy phương pháp đề xuất đã đạt được hiệu quả cao. Trong nghiên cứu [19] đã sử dụng mô hình CRFs hai cấp với sự phụ thuộc lẫn nhau không cố định để trích xuất các quan hệ so sánh. Điều này được thực hiện bằng cách sử dụng các phụ thuộc phức tạp giữa các quan hệ, thực thể và từ, và sự phụ thuộc lẫn nhau không cố định giữa các quan hệ. Mục đích tạo ra một mô hình đồ họa để trích xuất và phản ánh mối quan hệ so sánh giữa các sản phẩm từ đánh giá của khách hàng. Kết quả được hiển thị dưới dạng bản đồ quan hệ so sánh để hỗ trợ quyết định trong quản lý rủi ro doanh nghiệp. Thực nghiệm trên các đánh giá của khách hàng trên thiết bị di động từ amazon.com, epinions.com, blog, dịch vụ mạng xã hội (SNS - Social Networking Services) và emails. Kết quả cho thấy phương pháp đề xuất có thể trích xuất các quan hệ so sánh chính xác hơn các phương pháp khác và bản đồ quan hệ so sánh của chúng có khả năng là một công cụ rất hiệu quả để hỗ trợ việc ra quyết định và quản lý rủi ro doanh nghiệp.

Chỉ sử dụng cách tiếp cận dựa trên ngữ liệu không hiệu quả bằng cách tiếp cận dựa trên từ điển vì khó có thể chuẩn bị một kho ngữ liệu lớn để bao gồm tất cả các từ tiếng Anh, nhưng cách tiếp cận này có một ưu điểm chính là có thể giúp tìm các từ quan điểm cụ thể về miền và ngữ cảnh. Sau đó, sử dụng một kho dữ liệu miền để định hướng phân cực quan điểm. Các mô hình khai phá quan điểm dựa trên từ vựng được thống kê trong Bảng 1.2.

Bảng 1.2: Các mô hình khai phá quan điểm dựa trên từ vựng

Nghiên cứu	Phương pháp	Dữ liệu thực nghiệm	Accuracy	Precision	Recall	F1
Alshutayri et al. (2017) [20]	Corpus-based	Twitter Reviews	79,00	-	-	-

Khoo et al. (2018) [21]	Dictionary-based	Amazon product reviews	69,00	-	-	-
Wook et al. (2020) [22]	Dictionary-based	Students' feedback	79,40	-	-	-
Grljević et al. (2020) [23]	Dictionary-based	HiEd-Sen Reviews	-	-	-	89,75

Các phương pháp biểu diễn từ (word representation) đã phát triển từ các mô hình dựa trên từ điển cố định sang các mô hình biểu diễn vector học được từ dữ liệu và các biểu diễn ngữ cảnh hóa. Cụ thể, vector từ (Word2Vec-Word to Vector) [97] được đề xuất nhằm học các biểu diễn vector phân tán của từ từ tập dữ liệu văn bản quy mô lớn. Tiếp theo, các vectơ toàn cục để biểu diễn từ (GloVe - Global Vectors for Word Representation) [98] kết hợp thống kê đồng xuất hiện toàn cục của từ với phương pháp học vector để tạo ra các biểu diễn từ giàu ngữ nghĩa. Sau đó, các phép nhúng từ mô hình ngôn ngữ (ELMo - Embeddings from Language Models) [104] giới thiệu cơ chế biểu diễn ngữ cảnh hóa phụ thuộc vào ngữ nghĩa của câu. Các nghiên cứu sau đó tiếp tục mở rộng biểu diễn ngôn ngữ sang môi trường đa ngôn ngữ [99]. Những biểu diễn này đã trở thành nền tảng quan trọng cho các mô hình phân tích tình cảm hiện đại dựa trên học chuyển giao và học sâu [100][101][102]. Khảo sát hệ thống hóa của [161] về kỹ thuật biểu diễn văn bản trong NLP tổng hợp toàn diện quá trình tiến hóa từ biểu diễn tĩnh sang ngữ cảnh hóa, cung cấp cơ sở lựa chọn phương pháp trích xuất đặc trưng phù hợp.

- Khai phá quan điểm dựa trên học máy

Hướng tiếp cận xây dựng mô hình học máy [96] để giải quyết bài toán khai phá quan điểm bao gồm học máy có giám sát, học máy bán giám sát và học máy không giám sát.

Các phương pháp học máy có giám sát phụ thuộc vào tập dữ liệu huấn luyện đã được gán nhãn trước. Trong phân loại quan điểm, nhiều thuật toán đã được áp dụng như Naive Bayes, Maximum Entropy và Support Vector Machines [87], bên cạnh các phương pháp khác như cây quyết định và bộ phân loại dựa trên luật. Khảo sát toàn diện của [165] về các mô hình học sâu cho phân loại văn bản cung cấp nền

tảng lý thuyết cho các phương pháp ABSA. Nghiên cứu [24] cung cấp đánh giá so sánh toàn diện các phương pháp học sâu tiên tiến để giải quyết bài toán ABSA, tổng hợp các tập dữ liệu chuẩn, số liệu đánh giá và phân tích hiệu suất của các mô hình mạng nơ-ron sâu, từ các mô hình đệ quy học cấu trúc thành phần ngữ nghĩa [14] đến các kiến trúc học sâu hiện đại. Nền tảng kỹ thuật của các tiếp cận này là cơ chế self-attention trong Transformer [103] và biểu diễn từ ngữ cảnh hóa sâu [104], tạo tiền đề cho các mô hình tiền huấn luyện quy mô lớn. Khảo sát hệ thống của [25] về học chuyển giao tuần tự (sequential transfer learning) xem xét sự phát triển của các mô hình tiền huấn luyện và ứng dụng của chúng vào các nhiệm vụ phân tích tình cảm đa dạng, bao gồm ABSA, phân tích tình cảm đa phương thức, phát hiện mỉa mai, phân loại tình cảm đa miền, phân tích tình cảm đa ngôn ngữ và phát hiện cảm xúc. Nghiên cứu về tính toán và phân tích tình cảm chuyển từ văn bản sang hình ảnh cũng được đề xuất trong [26]. Phân tích tình cảm trực quan với mạng tăng cường phân biệt được giám sát yếu (WSDEN - Weakly Supervised Discriminative Enhancement Network) được sử dụng để xây dựng khung dự đoán tình cảm. Các thử nghiệm mở rộng trên ba cơ sở dữ liệu điểm chuẩn cho thấy phương pháp được đề xuất vượt trội hơn các phương pháp phân tích tình cảm trực quan hiện đại. Trong lĩnh vực giao thông, nghiên cứu [27] đề xuất đánh giá các phương tiện sử dụng năng lượng mới (NEVs - New Energy Vehicles) bằng tập hợp hình ảnh mờ dựa trên phân tích tình cảm từ các đánh giá trực tuyến. Trong những năm gần đây nhiều nhà nghiên cứu đã tìm cách khám phá cách biểu diễn khía cạnh cụ thể thông qua cơ chế chú ý [28]. Mặc dù đã đạt được sự cải thiện đáng kể trong việc phân loại tình cảm ở cấp độ khía cạnh nhưng các phương pháp này vẫn có hiệu suất kém trong một số trường hợp nhất định vì nhiều lý do. Thứ nhất là do áp dụng cơ chế chú ý ở mức độ thô, có thể gây mất thông tin khi câu theo ngữ cảnh dài hoặc bao gồm nhiều tình cảm và khía cạnh chứa nhiều từ. Thứ hai là chỉ xem xét ảnh hưởng của các từ khóa theo ngữ cảnh đến sự phân cực tình cảm mà bỏ qua tầm quan trọng của các cụm từ khóa trong câu. Trong nghiên cứu [7] đã phân loại các phương pháp phân loại tình cảm dựa trên khía cạnh có ba loại chính: mô hình dựa trên tri thức, học máy và mô hình hỗn hợp. Phương pháp tiếp cận dựa trên tri thức bao gồm phương pháp dựa trên từ điển, phương pháp dựa trên ontology và phương pháp dựa trên diễn ngôn. Trái ngược với các phương pháp dựa

trên tri thức để khai phá các cơ sở tri thức để phân loại tình cảm, phương pháp tiếp cận dựa trên học máy sử dụng tập dữ liệu huấn luyện gồm các vectơ đặc trưng tương ứng với các nhãn chính xác. Mô hình học máy được huấn luyện để trích xuất các mẫu từ dữ liệu có thể được sử dụng để phân biệt giữa các lớp tình cảm, có nhiều loại mô hình học máy, có thể được nhóm thành các loại: mô hình học máy vector hỗ trợ (SVM- Support Vector Machine), mô hình dựa trên cây, mô hình học sâu và mô hình học sâu dựa trên cơ chế chú ý. Khi dữ liệu hạn chế, các mô hình học máy có thể không mang lại hiệu quả khả quan. Vì vậy, cần có sự kết hợp thêm tri thức từ cơ sở tri thức để bù đắp cho việc thiếu dữ liệu, việc kết hợp cơ sở tri thức mới với mô hình học máy được gọi là mô hình hỗn hợp (hybrid models). Có nhiều cách khác nhau để kết hợp cơ sở tri thức với học máy. Một cách tiếp cận phổ biến là sử dụng cơ sở tri thức để xác định các đặc trưng mà thuật toán học máy sử dụng để dự đoán tình cảm. Một cách khác là triển khai cả bộ phân loại dựa trên tri thức và bộ phân loại học máy theo trình tự hoặc song song. Trong nghiên cứu [6] đã đánh giá toàn diện về vấn đề phân tích tình cảm dựa trên khía cạnh, bao gồm các nhiệm vụ, phương pháp, thách thức hiện tại và các giải pháp tương ứng, đưa ra các hướng đi tiềm năng khác nhau, cụ thể là sắp xếp các nghiên cứu theo thứ tự các yếu tố tình cảm có liên quan, tập trung vào những tiến bộ gần đây trên cơ sở các nhiệm vụ phân tích tình cảm đơn giản. Tuy nhiên, việc tìm kiếm một yếu tố tình cảm duy nhất vẫn chưa thỏa đáng để hiểu được quan điểm ở cấp độ khía cạnh đầy đủ hơn, điều này đòi hỏi không chỉ việc trích xuất nhiều yếu tố tình cảm mà còn phải thừa nhận sự tương ứng và phụ thuộc giữa chúng. Nhằm giải quyết bài toán dự đoán đồng thời nhiều thành phần của quan điểm, nghiên cứu [29] đã đề xuất nhiệm vụ dự đoán bộ tứ tình cảm (Sentiment Quad Prediction) cùng với bộ dữ liệu chuẩn tương ứng. Khác với các phương pháp trích xuất tuần tự từng thành phần, nghiên cứu tiếp cận bài toán theo hướng sinh văn bản (text generation) để dự đoán đồng thời bốn yếu tố gồm danh mục khía cạnh (aspect category), thuật ngữ khía cạnh (aspect term), thuật ngữ quan điểm (opinion term) và cực tính cảm xúc (sentiment polarity). Đồng thời, các bộ dữ liệu chuẩn được giới thiệu trong những năm gần đây đã tạo điều kiện thuận lợi cho việc nghiên cứu các phương pháp dự đoán đồng thời nhiều yếu tố tình cảm [138]. Các phương pháp tiên tiến nhất về phân tích tình cảm dựa trên khía cạnh có thể được tóm lược như sau. Đầu

tiên là sử dụng các phương pháp thiết kế để phát hiện các khía cạnh như phương pháp dựa trên mẫu, học máy và phương pháp học sâu. Các phương pháp này đều hiệu quả để phát hiện khía cạnh trong phân tích tình cảm và tính linh hoạt của miền. Để phát huy hiệu quả của các phương pháp trên đồng thời tăng cường khả năng diễn giải của các mô hình học máy thì cây quyết định là mô hình dự đoán trong học tập có giám sát, được biết đến không chỉ vì tiện ích trong nhiều ứng dụng mà còn có khả năng diễn giải và độ tin cậy cao. Cây quyết định được nghiên cứu cách đây khoảng 60 năm và trong thập kỷ qua các nhà nghiên cứu đã giải quyết các vấn đề quan trọng trong lĩnh vực này. Trong nghiên cứu đã đánh giá những tiến bộ gần đây trong nghiên cứu về cây quyết định, tập trung vào ba mục tiêu chính của bộ học dự đoán như các vấn đề về khớp với dữ liệu huấn luyện, khái quát hóa và khả năng diễn giải. Tuy nhiên, cây quyết định có những hạn chế đối với dữ liệu lớn và nguy cơ mô hình huấn luyện bị quá khớp (overfitting). Đây là vấn đề cần phải giải quyết để khắc phục hạn chế của cây quyết định. Để khắc phục các hạn chế này, học quần thể đã được đề xuất và có xu hướng phát triển trong các mô hình học máy hiện đại. Gần đây, học quần thể với các bộ phân loại cơ sở giống nhau như học quần thể dựa trên cây quyết định và học quần thể dựa trên mạng nơ-ron sâu đã cho thấy hiệu quả đáng kể trong việc cải thiện tính khái quát hóa của hệ thống học tập. Tuy nhiên, mạng nơ-ron sâu hiện đại thường có hàng triệu đến hàng tỷ tham số, chi phí về thời gian và không gian để huấn luyện nhiều bộ học sâu cơ bản và thử nghiệm với bộ học sâu kết hợp lớn hơn nhiều so với phương pháp học quần thể truyền thống. Các mô hình khai phá quan điểm dựa trên học máy được thống kê tại Bảng 1.3.

Bảng 1.3: Các mô hình khai phá quan điểm dựa trên học máy

Nghiên cứu	Phương pháp	Dữ liệu thực nghiệm	Accuracy	Precision	Recall	F1
S. Kumar et al. (2020) [30]	NB ME SVM	Facebook reviews	75,00 76,00 78,00	- - -	- - -	- - -
J. Samuel et al. (2020) [31]	NB LR	COVID-19 tweets	91,00 74,00	- -	- -	- -
S. Zahoor and R. Rohilla (2020) [32]	NB SVM RF LSTM	Tweets reviews	96,80 96,00 95,94 95,28	- - - -	- - - -	- - - -

P. Mukherjee et al. (2021) [33]	RNN ANN SVM NB	Amazon reviews	95,80 96,32 85,54 82,15	92,95 94,36 85,80 83,60	99,19 98,59 85,43 80,32	95,97 96,43 85,61 81,93
B. Noori et al (2021) [34]	NB SVM DT KNN	Customer reviews	70,60 77,52 98,90 72,00	71,20 78,63 99,10 75,10	70,30 77,91 98,90 72,70	70,75 78,27 99,00 73,88
S. A. Arif et al. (2021) [35]	LR SVM	Tweet reviews	88,00 87,00	- -	- -	- -
C. Messaoudi et al. (2022) [37]	LSTM CNN CRF GRU	Twitter reviews	78,40 78,20 83,20 78,80	79,40 79,00 82,40 79,40	79,10 78,73 83,20 78,60	78,50 78,30 82,70 78,60
B. AlBadani et al. (2022) [38]	SVM	Twitter reviews	99,78 99,71 95,78	- - -	- - -	- - -
A. Adak et al (2022) [39]	LSTM LSTM-CNN	Product reviews	96,07 96,33	- -	- -	- -
J. Ahmed and M. Ahmed (2023) [40]	NB	Articles reviews	89,30	-	-	-
P. Gaur et al. (2023) [40]	NB	Twitter sentiment	84,44	-	-	-
Y. Qi and Z. Shabrina (2023) [41]	SVC	Tweets COVID-19	71,00	-	-	-
M. Susmitha et al. (2023) [42]	NB LR SVM DT RF	E-commerce reviews	93,40 93,70 93,80 89,99 93,50	- - - - -	- - - - -	- - - - -

Tiếp cận học máy bán giám sát, nghiên cứu [45] đã lập mô hình phương pháp học bán giám sát để nghiên cứu vấn đề phân cụm đặc trưng của sản phẩm để khai phá quan điểm và đề xuất sử dụng thuật toán tối đa hóa kỳ vọng (EM-Expectation Maximization) để giải quyết vấn đề. Thuật toán học bán giám sát đã được sử dụng để

học từ một tập hợp nhỏ các câu được gắn nhãn và một tập hợp lớn các câu không được gắn nhãn. Thuật toán học bán giám sát được xây dựng dựa trên thuật toán cực đại hóa kỳ vọng, trong đó mô hình phân loại Bayes ngây thơ (NB - Naïve Bayes) được sử dụng làm bộ phân loại cơ sở (base classifier). Mô hình thực hiện phân loại văn bản thành tích cực (positive), tiêu cực (negative). Một phương pháp học bán giám sát được đề xuất để nhóm các biểu thức khía cạnh thành một số loại khía cạnh do người dùng chỉ định, nhằm phản ánh nhu cầu của người dùng. Trước tiên, gắn nhãn thủ công cho một số lượng nhỏ biểu thức khía cạnh gốc cho mỗi danh mục. Sau đó, hệ thống sẽ gán phần còn lại của các biểu thức khía cạnh cho các danh mục phù hợp bằng cách sử dụng phương pháp học bán giám sát làm việc với các mẫu được gắn nhãn và không được gắn nhãn.

Tiếp cận học máy không giám sát được sử dụng để tự động khai thác các khía cạnh trong các bài phê bình trên mạng xã hội và những tình cảm được thể hiện ở các khía cạnh khác nhau. Mô hình chủ đề (LDA - Latent Dirichlet Allocation) [105] được sử dụng để khám phá các chủ đề chung về nhiều khía cạnh của các bài đánh giá trên mạng xã hội. Sau đó, các chủ đề cụ thể được trích xuất và tình cảm liên quan dựa trên bối cảnh văn bản đánh giá. Các bài đánh giá xã hội được trích xuất từ tập dữ liệu Blog và từ điển. Cách tiếp cận này đã thu được kết quả phân vùng chủ đề tốt và giúp cải thiện độ chính xác của phân tích tình cảm. Mô hình giúp khám phá các chủ đề được tinh chỉnh nhiều khía cạnh và tình cảm liên quan. Ngoài ra, cách tiếp cận không giám sát khác phụ thuộc vào định hướng ngữ nghĩa bằng cách sử dụng độ đo thông tin tương hỗ theo từng điểm (PMI-Pointwise Mutual Information) hoặc liên kết từ vựng sử dụng PMI, không gian ngữ nghĩa và sự tương tự trong phân phối để đo mức độ giống nhau giữa các từ và các nguyên mẫu phân cực.

Trong nghiên cứu [46] cũng đề xuất một cách tiếp cận không giám sát để xác định các câu đánh giá trong các cuộc thảo luận trực tuyến. Nghiên cứu đã triển khai một hệ thống dựa trên quy tắc để khai thác các đặc trưng, quan điểm của sản phẩm và điểm số độ tin cậy của chúng. Hệ thống được đề xuất sử dụng phân tích ngôn ngữ và ngữ nghĩa của văn bản để khai thác các cặp quan điểm đặc trưng từ các tài liệu đánh giá. Sau đó, thuật toán xếp hạng tìm kiếm chủ đề dựa trên liên kết (HITS-Hyperlink-Induced Topic Search) đã được sử dụng để thu được điểm độ tin cậy cho

cặp đặc trưng-quan điểm và kho dữ liệu. Phương pháp không giám sát để phân loại các bài đánh giá là thích (được khuyến nghị) hoặc không thích (không được khuyến nghị) [106]. Phương pháp không giám sát được sử dụng để phân loại tình cảm ở cấp độ tài liệu và thông tin tương hỗ điểm để có được định hướng ngữ nghĩa trung bình của các bài đánh giá. Thuật toán đạt độ chính xác trung bình là 74% cho 410 lượt đánh giá. Sau đó, các tác giả đã mở rộng công việc bằng cách trình bày một cách tiếp cận để tìm ra định hướng ngữ nghĩa của văn bản bằng cách tính toán sự liên kết thống kê của nó với một tập hợp các từ tích cực và tiêu cực bằng cách sử dụng phương pháp PMI và phân tích ngữ nghĩa tiềm ẩn (LSA- Latent Semantic Analysis). Phương pháp này khi thử nghiệm với 3.596 từ (1.614 từ tích cực và 1.982 từ tiêu cực) đạt độ chính xác 82,8%.

1.5.1.2. Các lĩnh vực áp dụng

Khai phá quan điểm có thể được áp dụng cho nhiều lĩnh vực khác nhau như du lịch, giáo dục, an ninh quốc gia, quốc phòng, kinh doanh, chính trị, công cộng, tài chính, mối nguy hiểm, sức khỏe và an toàn [47]. Phạm vi ứng dụng rộng rãi sẽ nhằm mục đích đạt được sự kết hợp tốt nhất các điểm mạnh của các tiếp cận khai phá quan điểm. Tùy thuộc vào yêu cầu/mục đích/khuôn khổ của mô hình khai phá quan điểm, các ứng dụng có thể khác nhau, từ dự đoán đơn giản về tính phân cực của một từ đến khám phá thông tin nhạy cảm hoặc ẩn giấu, hoặc thậm chí là mô hình bảo vệ một quốc gia khỏi bất kỳ cuộc tấn công khủng bố tiềm tàng nào hoặc thảm họa. Nhiều công trình nghiên cứu đề cập đến các lĩnh vực ứng dụng khác nhau dựa trên các lĩnh vực hoặc tiếp cận khác nhau được sử dụng. Kiến thức về các lĩnh vực ứng dụng đa dạng hoàn toàn dựa trên tập dữ liệu hiện có là một thách thức khó tìm thấy trong các tài liệu nghiên cứu hiện có. Mục này nhằm phác thảo một số lĩnh vực ứng dụng khai phá quan điểm dựa trên dữ liệu/nội dung/tài liệu có sẵn mà các nhà nghiên cứu có thể sử dụng để khai phá quan điểm. Trong lĩnh vực giáo dục cụ thể, khai phá quan điểm đã được ứng dụng để phân tích đánh giá khóa học trực tuyến và phản hồi của người học, hỗ trợ giảng viên và đơn vị đào tạo cải thiện chất lượng nội dung giảng dạy thông qua các kỹ thuật học sâu kết hợp phân loại chủ đề [44].

Trong lĩnh vực kinh doanh và thương mại điện tử, khai phá quan điểm từ các đánh giá trực tuyến của người dùng đã trở thành một trong những hướng ứng dụng

phổ biến và có giá trị thực tiễn cao nhất [1]. Với sự bùng nổ của các nền tảng mua sắm trực tuyến, số lượng sản phẩm, thương hiệu và nhà cung cấp ngày càng gia tăng, khiến người tiêu dùng phải đối mặt với khối lượng thông tin khổng lồ khi đưa ra quyết định mua hàng [43]. Đây là một trong những bài toán được nghiên cứu sớm nhất của lĩnh vực, trong nghiên cứu [49] đã đặt nền móng cho hướng trích xuất và phân loại ngữ nghĩa các đánh giá sản phẩm trực tuyến. Mặc dù các trang thương mại điện tử cung cấp hệ thống xếp hạng sao và cho phép khách hàng để lại nhận xét, việc xử lý thủ công hàng nghìn phản hồi là bất khả thi trong thực tế. Phân tích tình cảm tự động giúp khắc phục hạn chế này bằng cách tổng hợp và cung cấp cái nhìn ngắn gọn, có cấu trúc về toàn bộ sản phẩm, thương hiệu, hoặc từng khía cạnh kỹ thuật cụ thể của sản phẩm [1]. Đặc biệt với lĩnh vực khách sạn và nhà hàng, trọng tâm dữ liệu thực nghiệm của luận án, khảo sát hệ thống của [167] qua 68 công trình về ứng dụng học máy cho phân tích tình cảm khách hàng trong lĩnh vực khách sạn và du lịch xác nhận xu hướng sử dụng phổ biến các thuật toán cây quyết định, rừng ngẫu nhiên và mạng nơ-ron, đồng thời chỉ ra khoảng trống về kỹ thuật học quần thể chưa được khai thác đầy đủ trong lĩnh vực này, đúng với định hướng mà luận án theo đuổi. Khảo sát toàn diện của [119] về phân tích tình cảm đã tổng hợp các phương pháp xử lý ngôn ngữ tự nhiên, học máy, học sâu và mô hình ngôn ngữ lớn trong phân tích đánh giá khách hàng trực tuyến, đồng thời cho thấy các nền tảng thương mại điện tử như Amazon và Yelp là những nguồn dữ liệu được sử dụng phổ biến trong nghiên cứu và đánh giá các mô hình khai phá quan điểm. Trong khảo sát [195] cung cấp một đánh giá tổng quan có hệ thống về các phương pháp, ứng dụng và thách thức trong phân tích cảm xúc; phân tích tổng hợp quy mô lớn của [168] tổng hợp 156 nghiên cứu với 69.006 quan sát khẳng định đánh giá trực tuyến có ảnh hưởng đáng kể đến ý định mua hàng, đặc biệt nhấn mạnh vai trò của nội dung đánh giá, là lý do trực tiếp cho sự cần thiết của các mô hình khai phá quan điểm hiệu quả từ đánh giá khách hàng.

Các ứng dụng của khai phá quan điểm trong lĩnh vực này rất đa dạng và bao phủ nhiều đối tượng khác nhau. Từ góc độ người mua, hệ thống có thể hỗ trợ phân tích và so sánh sản phẩm giữa các nhà cung cấp. Từ góc độ người bán và nhà sản xuất, khai phá quan điểm cung cấp thông tin để nghiên cứu đối thủ cạnh tranh, theo dõi thương hiệu, quản lý danh tiếng và hoạch định chiến lược quảng cáo [1]. Đối với

các nền tảng thương mại điện tử, phân tích tình cảm hỗ trợ dịch vụ khách hàng và cá nhân hóa trải nghiệm người dùng.

Bên cạnh đó, tính tin cậy của dữ liệu đánh giá là một thách thức không thể bỏ qua. Sự xuất hiện ngày càng nhiều của các đánh giá giả mạo trên các nền tảng trực tuyến đòi hỏi phải có các cơ chế phát hiện hiệu quả trước khi sử dụng dữ liệu cho mục đích phân tích [3]. Các nghiên cứu đã đề xuất nhiều phương pháp tiếp cận, bao gồm bộ phân loại dựa trên quy tắc, bản thể luận đặc trưng theo miền và các độ đo khoảng cách, nhằm lọc và xác minh tính xác thực của đánh giá trước khi thực hiện khai phá quan điểm dựa trên khía cạnh [3]. Do đó, sự lựa chọn giữa các chiến lược phân tích dự đoán hay chẩn đoán sẽ là yếu tố then chốt định hướng việc phát triển các công nghệ xử lý đánh giá thương mại điện tử trong tương lai [1]. Hai khảo sát toàn diện gần đây hệ thống hóa lĩnh vực này: [6][174] phân tích các đặc trưng ngôn ngữ và hành vi để phân biệt đánh giá thật/giả, trong khi [175] cập nhật các kỹ thuật mới nhất từ năm 2019 đến năm 2023 bao gồm cả mô hình Transformer, đồng thời chỉ ra mất cân bằng dữ liệu thật/giả vẫn là thách thức chưa giải quyết triệt để, liên quan trực tiếp đến vấn đề mất cân bằng nhãn mà luận án xử lý thông qua mô hình rừng mật độ và rừng đa tạp.

Trong lĩnh vực chính trị, sự phát triển mạnh mẽ của các nền tảng truyền thông xã hội như Twitter, Facebook, blog và YouTube đã tạo ra một không gian công khai rộng lớn, nơi người dùng tự do bày tỏ quan điểm về các đảng phái chính trị, ứng cử viên bầu cử và chính sách của chính phủ [48]. Khối lượng dữ liệu khổng lồ từ các nền tảng này đã thu hút sự chú ý của các nhà nghiên cứu trong việc khai thác quan điểm phục vụ phân tích chính trị. Một trong những ứng dụng nổi bật nhất là dự đoán kết quả bầu cử dựa trên phân tích tình cảm từ dữ liệu mạng xã hội. Nghiên cứu tiên phong [107] đã chỉ ra rằng tỷ lệ đề cập và xu hướng tình cảm trong các tweet có thể phản ánh xu hướng bầu cử với độ chính xác đáng kể. Ngoài dự đoán bầu cử, khai phá quan điểm còn được ứng dụng để theo dõi cảm nhận của công chúng đối với các phong trào chính trị, đánh giá phản ứng của người dân trước các dự luật hoặc đạo luật được đề xuất, và giám sát diễn biến dư luận theo thời gian thực [48]. Một phát hiện quan trọng và nhất quán trong các nghiên cứu về truyền thông chính trị trực tuyến là hiện tượng phân cực tình cảm trong lan truyền thông tin: các tweet mang nội dung tiêu cực

có xu hướng được lan truyền rộng rãi hơn so với các tweet tích cực [108]. Điều này đặt ra những thách thức đáng kể cho các nhà phân tích khi sử dụng dữ liệu mạng xã hội để đánh giá dư luận khách quan, đồng thời mở ra hướng nghiên cứu về tác động của cảm xúc tiêu cực trong định hình nhận thức chính trị của công chúng. Khảo sát hệ thống gần đây của [172] về dự đoán bầu cử dựa trên dữ liệu Twitter tổng hợp các phương pháp tiếp cận, phân loại theo cách tiếp cận, trích dẫn và công nghệ sử dụng, đồng thời chỉ ra rằng hiệu quả dự đoán phụ thuộc mạnh vào đặc điểm quốc gia và nền tảng, minh chứng thêm cho tầm quan trọng của việc mô hình hóa các thành phần ngữ cảnh (người phát biểu, thời điểm) trong bộ năm quan điểm thay vì chỉ phân loại nhãn tình cảm đơn thuần. Khảo sát toàn diện và cập nhật nhất về phân tích tình cảm Twitter trong dự đoán bầu cử của [172] đã tổng kết tiến độ trong hai thập kỷ qua phân loại theo phương pháp, trích dẫn và công nghệ, xác nhận phân tích tình cảm từ mạng xã hội là công cụ hiệu quả để đo lường quan điểm cử tri với các phương pháp học máy như mô hình Naive Bayes, SVM và mô hình học sâu như biểu diễn mã hóa hai chiều từ Transformers (BERT - Bidirectional Encoder Representations from Transformers) là cách tiếp cận phổ biến nhất.

Trong lĩnh vực truyền thông xã hội, sự bùng nổ của các nền tảng kỹ thuật số đã tạo ra nguồn dữ liệu quan điểm khổng lồ, phản ánh phản ứng của công chúng trước các sự kiện và vấn đề xã hội theo thời gian thực [48]. Khai phá quan điểm trên mạng xã hội không chỉ đơn thuần là phân loại tình cảm mà còn mở rộng sang các ứng dụng phức tạp hơn: giám sát và theo dõi diễn biến dư luận xung quanh một sự kiện cụ thể, dự đoán các kết quả có thể xảy ra, và hỗ trợ ra quyết định trong các tình huống khẩn cấp hoặc hỗn loạn xã hội [65]. Các hệ thống phân tích tiên tiến có khả năng xác định thành phần nào của một sự kiện thu hút sự chú ý cao nhất từ người dùng, phát hiện nguồn nội dung, cụ thể là các blog hoặc tài khoản có ảnh hưởng chịu trách nhiệm gây ra sự thay đổi đột biến trong cảm xúc tập thể, đồng thời làm rõ các khía cạnh của sự kiện có tác động lớn nhất đến nhận thức cộng đồng [109].

Một minh chứng điển hình về ứng dụng của khai phá quan điểm trong bối cảnh chính sách mạng xã hội là nghiên cứu [110]. Nghiên cứu này phân tích ngôn ngữ và cảm xúc trong các tweet trước và sau lệnh cấm đăng ảnh phản cảm về hành vi tự làm hại bản thân trên Facebook và Instagram, sử dụng kho ngữ liệu gồm 8.013 tweet và

mô hình thống kê để trích xuất các chủ đề trừu tượng trong diễn ngôn [110]. Kết quả cho thấy lệnh cấm tạo ra sự chuyển dịch đáng kể trong cảm xúc người dùng, với sự gia tăng của các cảm xúc tức giận và buồn bã sau khi chính sách có hiệu lực. Thông qua việc làm rõ nội dung mang tính thời sự bằng mô hình chủ đề thống kê, nghiên cứu cung cấp cái nhìn sâu sắc về cách những người có quyền lợi liên quan nhìn nhận và phản ứng trước một thay đổi chính sách quan trọng, qua đó khẳng định tiềm năng của khai phá quan điểm trong việc đánh giá hiệu quả và tác động xã hội của các quyết định quản trị nội dung trực tuyến.

Trong lĩnh vực giải trí, khai phá quan điểm từ đánh giá của khán giả đã trở thành công cụ quan trọng hỗ trợ nhiều bên liên quan trong ngành điện ảnh [1]. Khối lượng đánh giá ngày càng tăng trên các nền tảng đánh giá phim trực tuyến như Cơ sở dữ liệu phim Internet (IMDb - Internet Movie Database), Rotten Tomatoes và các mạng xã hội đã tạo ra nguồn dữ liệu phong phú phục vụ phân tích cảm xúc và khai phá quan điểm của công chúng đối với các bộ phim sắp ra mắt hoặc đang công chiếu. Các kết quả phân tích này được ứng dụng trực tiếp vào việc dự đoán doanh thu phòng vé, một bài toán quan trọng đối với nhà phân phối và nhà sản xuất phim trong quá trình ra quyết định [111]. Bên cạnh đó, hệ thống gợi ý phim cá nhân hóa dựa trên phân tích tình cảm từ đánh giá của người dùng cũng đang được phát triển rộng rãi, giúp kết nối người xem với các bộ phim phù hợp với sở thích cá nhân [112]. Đối với các nhà phân phối và nhà sản xuất, việc phân tích các khía cạnh được đề cập nhiều nhất trong đánh giá như diễn xuất, kịch bản, kỹ xảo hoặc âm nhạc cung cấp cơ sở để điều chỉnh chiến lược quảng bá và cải thiện chất lượng sản phẩm trong các dự án tiếp theo. Khảo sát hệ thống về phân tích văn bản trong du lịch và khách sạn của [196] đã tổng kết 10 năm nghiên cứu từ năm 2013 đến năm 2023 trên 5 tạp chí hàng đầu của ngành, xác định các nền tảng đánh giá trực tuyến (TripAdvisor, Yelp) là nguồn dữ liệu chủ đạo và phân tích tình cảm theo khía cạnh là phương pháp được áp dụng nhiều nhất để hiểu mức độ hài lòng của khách hàng.

Trong lĩnh vực y tế và môi trường, khai phá quan điểm trên mạng xã hội đóng vai trò đặc biệt quan trọng trong việc nắm bắt phản ứng của công chúng trước các khủng hoảng và sự kiện đột ngột [10][31]. Khi đối mặt với các thảm họa như bão, động đất hay đại dịch, người dùng có xu hướng chia sẻ cảm xúc và thông tin theo thời

gian thực trên Twitter và Facebook, tạo ra nguồn dữ liệu quý giá để theo dõi diễn biến dư luận. Nghiên cứu [31] đã phân tích dữ liệu tweet liên quan đến COVID-19 để theo dõi sự tiến triển của tâm lý sợ hãi theo thời gian khi đại dịch đạt đỉnh ở Hoa Kỳ, áp dụng phân tích văn bản mô tả kết hợp trực quan hóa dữ liệu và phân loại trong học máy. Trong nghiên cứu [10] đã so sánh hiệu quả của các bộ phân loại học máy kết hợp trong việc phân tích tình cảm từ tweet COVID-19, đặt nền tảng cho việc lựa chọn phương pháp phù hợp trong bối cảnh dữ liệu y tế không cân bằng. Tuy nhiên, một thách thức đáng kể trong lĩnh vực này là việc nắm bắt các biểu đạt mỉa mai và ngôn ngữ ẩn dụ phổ biến trong các thảo luận về sức khỏe trực tuyến và để giải quyết vấn đề này, các phương pháp tiếp cận đa nền tảng kết hợp nhiều nguồn dữ liệu đang được nghiên cứu và phát triển nhằm tạo điều kiện thuận lợi cho việc phân tích truyền thông về y tế toàn diện hơn [10][31]. Khảo sát hệ thống quy mô lớn của [176] phân tích 139 nghiên cứu đã xác định mạng xã hội là nguồn dữ liệu hữu ích cho quản lý rủi ro thiên tai dù còn hạn chế về đa dạng ngôn ngữ và nền tảng dữ liệu, gợi ý hướng phát triển trong tương lai cho việc mở rộng mô hình bộ năm quan điểm sang dữ liệu đa ngôn ngữ và đa nền tảng. Nghiên cứu hệ thống toàn diện nhất gần đây về phân tích tình cảm trong lĩnh vực y tế cộng đồng của [120] đã xem xét 83 nghiên cứu từ 2020-2025, xác định mạng xã hội (như Twitter, Reddit) là nguồn dữ liệu chủ đạo và chỉ ra hướng phát triển mạnh sang các mô hình học quần thể, mô hình ngôn ngữ lớn (LLM - Large Language Model) và học sâu.

Trong lĩnh vực tài chính, phân tích tình cảm đã trở thành công cụ quan trọng hỗ trợ dự đoán thị trường chứng khoán. Phân tích tình cảm từ tweet kết hợp với dữ liệu giá lịch sử của năm công ty lớn, sử dụng mô hình bộ nhớ ngắn hạn dài (LSTM-Long Short-Term Memory) và mô hình mạng nơ-ron tích chập một chiều (CNN-1D-1-Dimensional Convolutional Neural Network), cho thấy khả năng dự đoán biến động giá cổ phiếu hiệu quả hơn so với phân tích kỹ thuật thuần túy [113]. Hướng tiếp cận này giúp nhà đầu tư xác định cổ phiếu tiềm năng và tối ưu hóa thời điểm giao dịch dựa trên tín hiệu cảm xúc thị trường. Về lựa chọn kiến trúc, CNN đảm nhận trích xuất đặc trưng cục bộ từ văn bản tin tức tài chính, trong khi LSTM chịu trách nhiệm mô hình hóa chuỗi thời gian, kết hợp hai kiến trúc trong mô hình CNN-LSTM lai cho thấy hiệu suất vượt trội so với mô hình đơn lẻ trên cả ba chỉ số đánh giá gồm: sai số

bình phương trung bình (MSE- Mean Squared Error), hệ số xác định (R^2 - Coefficient of Determination) và F1-score [114]. Phân tích tình cảm tích hợp vào mô hình CNN-LSTM lai cải thiện đáng kể độ chính xác dự báo chứng khoán, đặc biệt trong việc nắm bắt các biến động thị trường do cảm xúc dẫn dắt, so với mô hình chuỗi thời gian truyền thống ARIMA (AutoRegressive Integrated Moving Average) [115]. Ngoài ra, mô hình FinBERT kết hợp LSTM đạt hiệu suất cao nhất trong dự đoán biến động thị trường tài chính, vượt trội so với cả ARIMA lẫn LSTM [116]. Nghiên cứu [88] đề xuất mạng bộ nhớ ngắn hạn dài hai chiều dựa trên cơ chế chú ý (Att-BLSTM) để nắm bắt thông tin ngữ nghĩa quan trọng nhất trong câu. Khảo sát toàn diện của [173] về tích hợp phân tích tình cảm với mạng nơ-ron đồ thị (GNN) cho dự đoán chứng khoán, tổng hợp 76 công trình, khẳng định xu hướng kết hợp đa nguồn thông tin như tin tức, mạng xã hội, dữ liệu giá, mang lại cải thiện hiệu suất đáng kể so với phương pháp đơn lẻ, cùng nguyên lý kết hợp đa thành phần mà luận án áp dụng thông qua biểu diễn bộ năm quan điểm. Trong bài báo [137], tác giả đề xuất Sentic GCN (Sentic Graph Convolutional Network), một mô hình Graph Convolutional Network được tăng cường bằng tri thức cảm xúc từ SenticNet nhằm khai thác đồng thời quan hệ cú pháp và quan hệ cảm xúc giữa các từ để nâng cao hiệu quả phân tích cảm xúc theo khía cạnh. Đánh giá toàn diện của [142] trên 13 tác vụ phân tích tình cảm và 26 tập dữ liệu cho thấy mô hình LLM như ChatGPT [127] vượt trội ở các tác vụ đơn giản, nhưng còn hạn chế với các tác vụ phức tạp có cấu trúc như trích xuất bộ năm quan điểm, vì vậy luận án giải quyết khoảng trống nghiên cứu bằng học quần thể (Ensemble Learning) với đặc trưng bộ năm quan điểm tường minh thay vì fine-tuning mô hình LLM trực tiếp.

Ngoài ra, khảo sát quy mô lớn của [162] qua 154 công trình giai đoạn 2013–2023 về phân tích đánh giá khách hàng trực tuyến bằng NLP đề xuất một hệ thống phân loại ứng dụng theo miền dữ liệu, cho thấy thương mại điện tử và du lịch là hai lĩnh vực có khối lượng nghiên cứu lớn nhất, phù hợp với việc luận án lựa chọn dữ liệu Yelp và TripAdvisor cho thực nghiệm. Khảo sát [163] hệ thống hóa các phương pháp, tập dữ liệu phổ biến và hướng nghiên cứu tương lai của phân tích tình cảm nói chung, trong khi khảo sát so sánh thực nghiệm của [182] trên mạng xã hội khẳng định các kỹ thuật học quần thể dựa trên Bagging và Boosting tiếp tục là phương pháp phổ

biến và hiệu quả, là cơ sở thực nghiệm bổ sung cho việc lựa chọn hướng tiếp cận học quần thể của luận án.

Tóm lại, lĩnh vực nghiên cứu khai phá quan điểm và phân tích tình cảm đã trải qua quá trình phát triển dài từ khi những nghiên cứu tiên phong về tổng hợp đánh giá khách hàng và nghiên cứu [119] về trích xuất quan điểm sản phẩm đặt nền móng cho toàn bộ lĩnh vực. Từ đó đến nay, đã có nhiều hướng tiếp cận nghiên cứu khác nhau được đề xuất, và sự quan tâm của cộng đồng nghiên cứu ngày càng tăng để giải quyết các thách thức thực tiễn ngày càng lớn từ lượng dữ liệu quan điểm khổng lồ trên Internet [1][4]. Khảo sát toàn diện của [187] cho thấy xu hướng phát triển của phân tích tình cảm đang chuyển từ các mô hình học máy truyền thống sang các mô hình học sâu, Transformer, mô hình ngôn ngữ lớn và các kiến trúc lai. Đồng thời, khảo sát nhấn mạnh rằng những thách thức cốt lõi hiện nay không chỉ nằm ở việc nâng cao độ chính xác mà còn ở khả năng tổng quát hóa trên các miền dữ liệu khác nhau, tính bền vững trước sự thay đổi của ngôn ngữ, khả năng diễn giải và tính công bằng của mô hình. Các hướng nghiên cứu như học quần thể, học chuyển giao, học tự giám sát và các mô hình lai được xem là những giải pháp quan trọng nhằm nâng cao hiệu quả của các hệ thống khai phá quan điểm trong tương lai. Đặc biệt trong kỷ nguyên mô hình ngôn ngữ lớn LLM, nghiên cứu của [142] đánh giá toàn diện trên 13 tác vụ phân tích tình cảm và 26 tập dữ liệu cho thấy mô hình LLMs tuy vượt trội ở các tác vụ phân loại đơn giản nhưng thua kém các mô hình nhỏ chuyên biệt (SLMs) ở các tác vụ có cấu trúc phức tạp như ABSA, khẳng định hướng tiếp cận có cấu trúc (như bộ năm quan điểm) của luận án là bổ sung cần thiết ngay cả trong bối cảnh LLM.

Về phương pháp, hai hướng tiếp cận chính đã được phát triển song song. Hướng tiếp cận dựa trên từ vựng bao gồm các phương pháp dựa trên từ điển cảm xúc và phương pháp dựa trên kho ngữ liệu, tận dụng các nguồn tri thức ngôn ngữ có sẵn như SentiWordNet và WordNet [21]. Hướng tiếp cận học máy đa dạng hơn với ba nhánh chính: học có giám sát áp dụng cho dữ liệu có nhãn với các kỹ thuật phổ biến như Naive Bayes, SVM, cây quyết định, mạng nơ-ron và học quần thể [8], học bán giám sát kết hợp dữ liệu có nhãn và không có nhãn thông qua các thuật toán như EM với Naive Bayes [45], học không giám sát cho dữ liệu không có nhãn, thường sử dụng các mô hình chủ đề như pLSA/LDA [105].

Mặc dù đã đạt được nhiều tiến bộ đáng kể, các hệ thống khai phá quan điểm vẫn đối mặt với nhiều thách thức chưa được giải quyết triệt để, bao gồm xử lý ngôn ngữ mỉa mai, phụ thuộc ngữ cảnh văn hóa, mất cân bằng nhãn và khả năng tổng quát hóa sang các miền ứng dụng mới [4][147][6]. Điều này khẳng định khai phá quan điểm vẫn là một lĩnh vực nghiên cứu phong phú và nhiều tiềm năng phát triển.

1.5.2. Một số hạn chế của các tiếp cận khai phá quan điểm hiện nay

Mặc dù lĩnh vực khai phá quan điểm đã đạt được nhiều tiến bộ đáng kể, mỗi hướng tiếp cận vẫn tồn tại những hạn chế cố hữu cần được nhận diện rõ để định hướng nghiên cứu cải tiến. Các khảo sát toàn diện gần đây xác nhận rằng từ phương pháp dựa trên từ vựng, học máy truyền thống đến học sâu và mô hình ngôn ngữ lớn, mỗi nhóm đều có điểm mạnh và điểm yếu riêng biệt về độ chính xác, khả năng diễn giải, yêu cầu dữ liệu và chi phí tính toán [119].

1.5.2.1. Hạn chế của tiếp cận khai phá quan điểm thông thường

- *Tiếp cận dựa trên từ vựng (Lexicon-based):* Phương pháp này sử dụng từ điển tình cảm với điểm số phân cực được gán cố định cho từng từ. Hạn chế cốt lõi của tiếp cận từ vựng là không thể nắm bắt tình cảm phụ thuộc ngữ cảnh, bao gồm mỉa mai và ẩn dụ. Hiệu suất phụ thuộc nặng nề vào chất lượng, độ bao phủ và tính phù hợp miền của từ điển được chọn; từ điển đa năng thường hoạt động kém trong các lĩnh vực chuyên biệt nơi ý nghĩa từ có thể khác biệt và thuật ngữ chuyên ngành vắng mặt hoặc bị phân loại sai [120]. Ngoài ra, việc liên tục cập nhật từ điển để theo kịp ngôn ngữ mạng xã hội đang biến đổi nhanh là tốn kém và khó duy trì [121].

- *Tiếp cận học máy (Machine learning-based):* Các phương pháp học máy có giám sát từ Naive Bayes, SVM, cây quyết định đến mạng nơ-ron cho thấy khả năng thích ứng tốt với từng miền cụ thể nhưng đối mặt với hai thách thức lớn: (1) sự phụ thuộc vào dữ liệu gán nhãn số lượng lớn vừa tốn kém vừa hạn chế khả năng mở rộng sang miền mới; (2) các kỹ thuật học máy dựa trên đặc trưng văn bản hoặc từ điển tình cảm đặc biệt dễ bị tổn thương trước các vấn đề liên quan đến miền dữ liệu, nhất là khi xử lý dữ liệu lớn với phân phối lệch [121]. Đối với tiếp cận lai (hybrid) kết hợp học máy và từ vựng, các tài liệu đánh giá chứa nhiều nhiễu thường được gán điểm trung tính vì hệ thống không phát hiện được quan điểm rõ ràng.

- *Học sâu và mô hình tiền huấn luyện:* Sự xuất hiện của BERT [81], RoBERTa [83] và các mô hình Transformer đã đẩy độ chính xác phân loại tình cảm lên mức cao chưa từng có. Tuy nhiên, mạng nơ-ron sâu gặp khó khăn với sự dịch chuyển phân phối và dữ liệu gán nhãn hạn chế, dẫn đến hiện tượng quá khớp (overfitting) và khả năng tổng quát hóa kém sang các miền và nhiệm vụ khác nhau [122]. Hơn nữa, chi phí tính toán và yêu cầu GPU lớn cản trở việc triển khai thực tế trong môi trường tài nguyên hạn chế [81][83][100]. Các nghiên cứu gần đây về phân tích tình cảm dựa trên khía cạnh [131][132] cho thấy học sâu cần lượng dữ liệu và tài nguyên lớn.

1.5.2.2. Một số vấn đề tiếp cận khai phá quan điểm với học quần thể

Học quần thể (Ensemble Learning) đã được chứng minh hiệu quả trong cải thiện độ chính xác và tính ổn định của khai phá quan điểm [8][117][59][164][200][141]. Một số mô hình khai phá quan điểm với học quần thể được thống kê tại Bảng 1.4.

Bảng 1.4: Các mô hình khai phá quan điểm với học quần thể

Nghiên cứu	Phương pháp	Dữ liệu thực nghiệm	Accuracy	Precision	Recall	F1
A. Onan et al. (2015) [8]	Bagging (Naïve Bayes, SVM, C4.5, KNN, RBF)	Bộ dữ liệu đánh giá quan điểm	78,0	-	-	-
R. Alnashwan et al. (2015) [50]	Stacking (SVM, BPM, LR, DF)	Stanford, SemEval-2016, Health Care Reform (HCR)	78,3 87,3 83,2	77,7 88,9 78,8	78,4 85,8 81,8	78,0 87,3 80,3
M. M. Fouad et al. (2018) [51]	Stacking (SVM, LR, NB)	Stanford Twitter Corpus, Sanders, Health Care Reform	78,70 93,94 84,75	- - -	- - -	- - -
S. Sharma et al. (2018) [52]	Bagging (Random Forests)	Movie Review	77,48	-	-	-

H.-Q. Nguyen et al. (2018) [53]	Stacking (SVM, LR+ LSTM, CNN)	Travel Services Reviews	92,80	-	-	-
A. Alrehili et al. (2019) [54]	Stacking (NB, SVMs, Random forest, Bagging, Boosting)	Electronic	89,4	-	-	-
D. Tiwari et al. (2019) [55]	Bagging Boosting	US-Airlines reviews	76,0 74,0	- -	- -	- -
M. Khalid et al (2020) [56]	Stacking (Gradient Boosted, Support Vector Machine)	Mobile reviews	93,0	-	-	-
A. Mohammadi di et al. (2021) [57]	Stacking (CNN, LSTM, BiLSTM and GRU)	Restaurant reviews Laptop reviews	- -	69,3 67,5	- -	- -
R. V. A. Ogutu et al. (2022) [58]	Stacking (Naïve Bayes, SVM)	OnePlus 7 Pro reviews	96,5	-	-	-
M. M. Rahman et al (2021) [10]	Stacking (DT, SVC, LR); Bagging (SVC, 10 trees); Stacking (VC, BC, LR)	Twitter reviews	- - -	83,8 84,6 84,7	83,4 83,3 83,6	83,3 83,2 83,5
M. S. Basarslan	Stacking	Trip Advisor	84,1	-	-	79,6

et al. (2023) [59]	(NB, kNN, SVM, and LR)	reviews Rotten Tomatoes reviews Hotel reviews	85,8 84,7	- -	- -	87,4 80,1

Tuy nhiên, một số hạn chế đặc thù vẫn tồn tại:

- *Vấn đề dữ liệu mất cân bằng*: Hầu hết các thuật toán học quần thể truyền thống trong khai phá quan điểm gặp khó khăn với dữ liệu mất cân bằng khi các lớp tình cảm phân bố không đều, dẫn đến mô hình thiên lệch về phía lớp đa số [122].

- *Vấn đề mối quan hệ giữa các thành phần bộ năm:* Hầu hết các nghiên cứu hiện có về học quần thể trong khai phá quan điểm chưa khai thác đầy đủ mối quan hệ cấu trúc giữa các thành phần *(e, a, s, h, t)* của bộ năm quan điểm [79]. Mỗi thành phần thường được xử lý độc lập, làm mất đi thông tin tương quan quan trọng.

- *Vấn đề khả năng diễn giải:* Một thách thức lớn với mô hình học quần thể là thiếu khả năng diễn giải, việc sử dụng đồng thời nhiều mô hình khiến quá trình ra quyết định phức tạp hơn và khó hiểu hơn. Trong khi có thể truy vết đường quyết định trong một cây quyết định đơn lẻ, mức độ rõ ràng này không đạt được trong các mô hình rừng cây kết hợp [123].

Tổng hợp các phân tích trên cho thấy nghiên cứu khai phá quan điểm cần tiếp tục được phát triển theo hướng giải quyết đồng thời ba thách thức cốt lõi: cấu trúc đa chiều của bộ năm quan điểm, dữ liệu mất cân bằng và nhu cầu về mô hình có khả năng diễn giải cao, là động lực thúc đẩy hướng tiếp cận rừng kết hợp được đề xuất trong luận án này.

Ngoài ra, hướng nghiên cứu mới nổi gần đây đáng chú ý là việc khai thác cấu trúc đa tạp (manifold structure) trong không gian đặc trưng quan điểm. Trong nghiên cứu [118] đề xuất biểu diễn thưa dựa trên đa tạp (manifold-based sparse representation) cho khai phá quan điểm, khai thác giả thuyết đa tạp thông qua đồ thị láng giềng gần nhất (k-nearest neighbor graph) để học cấu trúc nội tại của dữ liệu quan điểm nhiều chiều, đây là bước tiến quan trọng so với các phương pháp Euclidean truyền thống. Tiếp nối hướng này, nghiên cứu [129] đề xuất khung đa mục tiêu biểu diễn đa tạp (MOMR - Multi-Objective Manifold Representation) tích hợp đặc trưng

toàn cục sâu với biểu diễn đa tạp cục bộ, kết hợp cơ chế tự chú ý (self-attention) để tăng khả năng diễn giải mô hình. Tuy nhiên, cả hai hướng tiếp cận trên đều dựa trên học sâu hoặc biểu diễn thưa, đòi hỏi lượng lớn dữ liệu gán nhãn và chi phí tính toán cao, đồng thời hạn chế khả năng diễn giải. Do đó, đây là khoảng trống nghiên cứu mà luận án hướng đến giải quyết. Khác biệt của đề xuất trong luận án so với [118] và [129] là rừng ngẫu nhiên không giám sát được sử dụng để ước tính ma trận quan hệ đa tạp trực tiếp từ dữ liệu bộ năm có cấu trúc *(e, a, s, h, t)* thay vì học sâu hay biểu diễn thưa. Cụ thể, hai quan điểm thường xuyên rơi vào cùng nút lá trong rừng được coi là gần nhau trên đa tạp dữ liệu quan điểm. Cách tiếp cận này có ba ưu điểm nổi bật: (1) tính diễn giải cao, cấu trúc cây quyết định cho phép truy vết rõ ràng cơ sở phân cụm quan điểm; (2) không đòi hỏi dữ liệu gán nhãn lớn, học không giám sát trực tiếp từ bộ năm quan điểm; (3) chi phí tính toán thấp, phù hợp triển khai trong các ứng dụng thực tế mà không yêu cầu phần cứng tăng tốc như GPU.

1.5.3. Xu hướng nghiên cứu hiện nay liên quan đến luận án

Hiện nay có ba xu hướng nghiên cứu mới liên quan trực tiếp đến hướng tiếp cận của luận án, cụ thể như sau:

Thứ nhất, các khảo sát quy mô lớn về phân tích tình cảm dựa trên khía cạnh [152] và phân tích tình cảm nói chung [146][147] đã hệ thống hóa lại các tác vụ, phương pháp và thách thức còn tồn đọng, trong đó nhấn mạnh khoảng trống về khả năng diễn giải kết quả theo từng thành phần quan điểm [150], vấn đề mà biểu diễn bộ năm quan điểm trong luận án hướng đến giải quyết.

Thứ hai, phương pháp luận học quần thể, khảo sát [149] tổng hợp các khái niệm và ứng dụng, khẳng định xu hướng kết hợp nhiều mô hình cơ sở đồng nhất tiếp tục hiệu quả với dữ liệu có cấu trúc bảng. Phát hiện thực nghiệm của [124] trên 48 tập dữ liệu chuẩn cho thấy mô hình dựa trên cây quyết định vẫn vượt trội hơn mạng nơ-ron sâu trên dữ liệu bảng cỡ vừa, được các khảo sát [154][155] đồng thuận. Đối với vấn đề dữ liệu mất cân bằng, khảo sát [156] khẳng định kết hợp nhiều cây quyết định tiếp tục là hướng tiếp cận hiệu quả.

Thứ ba, khả năng diễn giải mô hình, các khảo sát về trí tuệ nhân tạo có thể giải thích (XAI) [150][151] hệ thống hóa các phương pháp diễn giải mô hình học máy, trong đó SHAP được xác định là kỹ thuật model-agnostic phổ biến và đáng tin cậy.

Đồng thời, khảo sát cập nhật về cây quyết định [153] hệ thống hóa các tiến bộ gần đây theo ba mục tiêu: khớp dữ liệu, khái quát hóa và diễn giải, phù hợp với các mục tiêu mà các mô hình rừng đề xuất trong luận án hướng đến cải thiện. Nghiên cứu [179] nhằm mục đích cung cấp một đánh giá toàn diện về các phương pháp được sử dụng trong tài liệu để giải thích các mô hình thu được từ rừng ngẫu nhiên và đã đi sâu vào những tác động của sự mất cân bằng lớp đối với khả năng giải thích của các mô hình rừng ngẫu nhiên [178]. Trong khảo sát [157] về phân cụm dựa trên mật độ (density-based clustering) hệ thống hóa các thuật toán thuộc họ DBSCAN, phân tích vai trò của hai tham số ε và *MinPts* trong việc tự động xác định số cụm mà không cần giả định trước để xử lý dữ liệu mất cân bằng theo cấu trúc mật độ thực tế thay vì theo phân phối nhãn. Đồng thời, khảo sát so sánh thực nghiệm của [158] trên 45 nghiên cứu giai đoạn từ năm 2016–2023 cho thấy kỹ thuật Stacking, dù được sử dụng ít hơn Bagging và Boosting, lại đạt hiệu suất cao nhất trong phần lớn trường hợp so sánh.

Trong khảo sát thống kê toàn diện của [159] về học đa tạp (manifold learning) hệ thống hóa cơ sở toán học và thống kê của các phương pháp giảm chiều phi tuyến, làm rõ sự đánh đổi giữa bảo toàn cấu trúc cục bộ và toàn cục [193], cơ sở lý thuyết cho giả thuyết đa tạp được luận án áp dụng trên dữ liệu bộ năm quan điểm. Về xu hướng so sánh mô hình học sâu/LLM với các phương pháp khác cho phân tích tình cảm dựa trên khía cạnh, nghiên cứu thực nghiệm của [160] trên nhiều kiến trúc mạng nơ-ron sâu và mô hình ngôn ngữ lớn cho thấy hiệu suất giữa các nhóm phương pháp không có khoảng cách tuyệt đối, gợi ý rằng các mô hình dựa trên cấu trúc dữ liệu tường minh như rừng đa tạp và các phương pháp học sâu hiện đại có thể bổ trợ cho nhau, khi xét thêm yếu tố chi phí tính toán và khả năng diễn giải. Các phương pháp học đa tạp (manifold learning) được sử dụng phổ biến hiện nay là t-SNE (t-distributed Stochastic Neighbor Embedding) và UMAP (Uniform Manifold Approximation and Projection). Nghiên cứu phân tích và so sánh toán học của [193] đã làm rõ cơ sở lý thuyết và cơ chế hoạt động của hai phương pháp, cho thấy t-SNE tập trung tối ưu hóa việc bảo toàn cấu trúc cục bộ của dữ liệu, trong khi UMAP có khả năng bảo toàn tốt hơn các quan hệ toàn cục, đồng thời đạt hiệu quả tính toán và khả năng mở rộng cao hơn trên các tập dữ liệu lớn. Do đó, UMAP thường được ưu tiên trong các bài toán giảm chiều và trực quan hóa dữ liệu quy mô lớn, trong khi t-SNE vẫn là lựa chọn hiệu

quả khi cần biểu diễn rõ cấu trúc cục bộ của các cụm dữ liệu. Trong khảo sát so sánh toàn diện của [182] đánh giá 8 phương pháp ước lượng chiều nội tại tiêu biểu (Local PCA, MLE, DANCo, TwoNN, CA-PCA...) trên cả dữ liệu tổng hợp và dữ liệu thực tế, cho thấy hiệu suất các phương pháp này phụ thuộc mạnh vào kích thước mẫu, độ cong và mức nhiễu của đa tạp, đồng thời các phương pháp dựa trên PCA cục bộ (Local PCA, CA-PCA) thể hiện độ ổn định cao nhất trên dữ liệu nhiễu.

1.5.4. Cơ sở lựa chọn hướng tiếp cận học quần thể

Một câu hỏi thực tiễn quan trọng khi thiết kế hệ thống khai phá quan điểm là lựa chọn giữa các phương pháp học máy truyền thống (Machine Learning-ML), trong đó học quần thể (Ensemble Learning) là đại diện tiêu biểu và các mô hình học sâu (Deep Learning - DL), đặc biệt là kiến trúc Transformer và các mô hình ngôn ngữ lớn (LLM). Tổng hợp từ các khảo sát gần đây [141][142][143], luận án đề xuất bốn tiêu chí lựa chọn như sau:

- *Dạng biểu diễn dữ liệu đầu vào.* Khi dữ liệu đã được trích xuất và chuẩn hóa thành đặc trưng có cấu trúc dạng bảng (tabular data) như bộ năm quan điểm *o = (e, a, s, h, t)* trong luận án này, các phương pháp học quần thể dựa trên cây quyết định (Random Forest, Gradient Boosting, Stacking) thường đạt độ chính xác cạnh tranh hoặc vượt trội so với mô hình học sâu, đặc biệt trên tập dữ liệu vừa và nhỏ [141]. Ngược lại, khi đầu vào là văn bản thô chưa qua trích xuất đặc trưng, các mô hình Transformer (BERT, RoBERTa) và mô hình LLM phát huy ưu thế nhờ khả năng học biểu diễn ngữ nghĩa end-to-end [142][143].

- *Yêu cầu về khả năng diễn giải (interpretability).* Trong các lĩnh vực đòi hỏi tính minh bạch như phân tích quan điểm khách hàng phục vụ ra quyết định kinh doanh, mô hình học quần thể kết hợp với kỹ thuật diễn giải SHAP cung cấp giải thích định lượng rõ ràng cho từng dự đoán [144][145], trong khi mô hình học sâu, dù đạt độ chính xác cao, vẫn hoạt động chủ yếu như hộp đen và đòi hỏi thêm các kỹ thuật diễn giải hậu kiểm (post-hoc) phức tạp hơn, độ tin cậy thấp hơn [145].

- *Tài nguyên tính toán và quy mô dữ liệu huấn luyện*. Học quần thể dựa trên cây quyết định có chi phí huấn luyện thấp, không yêu cầu máy tính có cấu hình GPU, phù hợp với dữ liệu vừa và nhỏ (hàng nghìn đến hàng trăm nghìn mẫu) như các tập dữ liệu thực nghiệm trong luận án. Mô hình học sâu và LLM đòi hỏi tài nguyên tính

toán lớn và thường chỉ phát huy ưu thế khi có tập dữ liệu huấn luyện rất lớn (hàng triệu mẫu có nhãn) [143].

- *Khả năng xử lý hiện tượng ngôn ngữ phức tạp.* Mô hình học sâu, đặc biệt là mô hình ngôn ngữ lớn (LLM) có ưu thế vượt trội trong xử lý mỉa mai, phủ định, quan điểm ẩn và ngữ cảnh dài, những hiện tượng mà phương pháp dựa trên đặc trưng trích xuất sẵn khó nắm bắt đầy đủ [142].

- *Hướng mở rộng sang phân tích tình cảm đa phương thức*. Luận án có thể tiếp tục mở rộng sang phân tích tình cảm đa phương thức bao gồm văn bản kết hợp hình ảnh, âm thanh, video đã được khảo sát toàn diện trong [190], trong đó xu hướng chuyển từ phân tích đơn phương thức sang đa phương thức đang ngày càng chiếm ưu thế nhờ sự phong phú thông tin cảm xúc từ nhiều kênh dữ liệu, mở ra hướng tích hợp bổ sung cho mô hình bộ năm quan điểm của luận án trong các nghiên cứu tiếp theo. Khảo sát của [190] trình bày quá trình phát triển từ phân tích đơn phương thức đến kết hợp văn bản, âm thanh và hình ảnh.

Trên cơ sở các tiêu chí trên, luận án lựa chọn hướng tiếp cận học quần thể cho bài toán khai phá quan điểm theo bộ năm quan điểm vì: (1) dữ liệu đầu vào đã được cấu trúc hóa thành 5 thành phần *(e, a, s, h, t)*; (2) yêu cầu cao về khả năng diễn giải kết quả phục vụ ứng dụng thực tế; (3) chi phí tính toán cần duy trì ở mức hợp lý, không phụ thuộc hạ tầng GPU chuyên dụng. Đây cũng là khoảng trống nghiên cứu mà luận án hướng đến giải quyết, kết hợp ưu điểm về độ chính xác và khả năng diễn giải của học máy; kết quả của các mô hình học sâu được trình bày như mốc tham chiếu (xem §2.7, §3.7, §4.7).

1.5.5. Đề xuất nghiên cứu

Xuất phát từ các phân tích trong mục §1.5.2, §1.5.3, §1.5.4, luận án đề xuất hướng tiếp cận mới dựa trên học quần thể (Ensemble Learning) để khai phá quan điểm từ dữ liệu bộ năm có cấu trúc. Lựa chọn này được xây dựng trên bốn căn cứ khoa học sau: (1) dữ liệu bộ năm quan điểm *(e, a, s, h, t)* có bản chất là dữ liệu bảng (tabular) với các thành phần rời rạc, đặc trưng phù hợp hơn với cây quyết định và rừng ngẫu nhiên so với kiến trúc mạng nơ-ron vốn được thiết kế cho dữ liệu chuỗi liên tục [124][154]; (2) yêu cầu về khả năng diễn giải (interpretability) ngày càng trở thành tiêu chí bắt buộc trong các ứng dụng khai phá tri thức thực tiễn [125][76], học

quần thể đáp ứng yêu cầu này thông qua tầm quan trọng của đặc trưng (feature importance) và giá trị SHAP, gán giá trị tầm quan trọng theo lý thuyết trò chơi Shapley cho từng đặc trưng đối với mỗi dự đoán cụ thể để cung cấp cả giải thích cục bộ lẫn toàn cục cho mô hình phức tạp [126], trong khi các mô hình học sâu vẫn được xem là "hộp đen" khó diễn giải; (3) quy mô dữ liệu quan điểm thực tế thường ở mức trung bình (dưới 100K mẫu) phù hợp với học quần thể [124] hơn là các mô hình ngôn ngữ lớn (LLM) vốn đòi hỏi hàng triệu mẫu huấn luyện [127][128]; (4) các nghiên cứu gần đây cho thấy mô hình rừng kết hợp với phân cụm mật độ và học đa tạp mang lại lợi thế đặc biệt trên dữ liệu quan điểm đa chiều [118][129][59], một hướng tiếp cận còn nhiều khoảng trống nghiên cứu cần được khai thác.

Dựa trên các căn cứ trên, luận án đề xuất ba mô hình học quần thể có tính kế thừa và bổ trợ lẫn nhau, tương ứng với ba cấp độ tiếp cận ngày càng tinh tế hơn đối với cấu trúc dữ liệu quan điểm. Cụ thể, ba đóng góp về mô hình được trình bày như sau (đóng góp thứ tư về dữ liệu được nêu tại phần Mở đầu):

Thứ nhất, mô hình rừng phân loại quan điểm dựa trên ma trận bộ năm được đề xuất, trong đó cấu trúc ngữ nghĩa được bảo toàn khi quá trình phân loại được thực hiện. Mô hình được kiểm chứng trên các bộ dữ liệu Amazon, TripAdvisor và VLSP-2018 (Phụ lục F); trên VLSP-2018, kết quả được đặt cạnh các mô hình cơ sở Logistic Regression, SVM, Naive Bayes và mô hình tham chiếu PhoBERT (dùng văn bản gốc làm đầu vào).

Thứ hai, mô hình rừng mật độ quan điểm tích hợp DBSCAN được đề xuất để xử lý dữ liệu mất cân bằng, phương pháp SHAP cùng chỉ số tương tác Shapley được sử dụng để diễn giải kết quả. Mô hình được kiểm chứng trên tập dữ liệu yelpHotelData và yelpResData kèm theo phân tích nghiên cứu cắt giảm đối với các tham số DBSCAN.

Thứ ba, mô hình rừng đa tạp quan điểm được đề xuất để khai thác cấu trúc đa tạp thấp chiều của không gian đặc trưng. Thay vì sử dụng các phương pháp chiếu đa tạp truyền thống như ISOMAP hay UMAP, luận án đề xuất học cấu trúc đa tạp trực tiếp từ rừng ngẫu nhiên không giám sát thông qua tính chất đồng xuất hiện tại nút lá, không cần giải bài toán trị riêng toàn cục. Tính phù hợp của giả thuyết đa tạp được kiểm chứng thông qua kết quả thực nghiệm so sánh với các mô hình khác tại §4.8.

1.6. Trích xuất bộ năm quan điểm

1.6.1. Mô hình trích xuất

Cho $D = \{d_1, d_2, \dots, d_n\}$ biểu diễn một tập hợp các tài liệu quan điểm, trong đó mỗi tài liệu d_i bao gồm một hoặc nhiều câu thể hiện quan điểm của người dùng. Đối với mỗi d_i, mục tiêu là trích xuất một tập hợp các bộ năm quan điểm $O_i = \{o_1, o_2, \dots, o_m\}$, trong đó mỗi bộ năm quan điểm $o_j = (e, a, s, h, t)$ có năm thành tố bao gồm: thực thể, khía cạnh, tình cảm, người giữ quan điểm và thời gian đưa ra quan điểm. Nhiệm vụ trích xuất được thực hiện kết hợp trên các thành phần của bộ năm được trích xuất từ câu đầu vào x và đồ thị phụ thuộc $G = (V, E)$ của câu. Trong đó, V là tập hợp các từ trong câu và E là tập hợp các mối quan hệ phụ thuộc cú pháp kết nối các từ trong câu.

Cho O biểu diễn tập hợp bộ năm quan điểm $o = (e, a, s, h, t)$ cho câu x, được suy luận thông qua mối quan hệ phụ thuộc của các thành phần trong bộ năm quan điểm. Tập hợp tối ưu các bộ năm quan điểm được tạo ra bằng cách tối đa hóa xác suất kết hợp tại công thức (1.32).

$$\hat{O} = \arg\max_{O} \prod_{o \in O} P(e \mid x)\, P(a \mid e, x)\, P(s \mid e, a, x) \times P(h \mid s, G)\, P(t \mid x), \tag{1.32}$$

Trong đó $P(e \mid x)$ biểu diễn xác suất trích xuất thực thể, $P(a \mid e, x)$ là xác suất trích xuất khía cạnh có điều kiện của thực thể, $P(s \mid e, a, x)$ là xác suất phân loại tình cảm, $P(h \mid s, G)$ là xác suất trích xuất người giữ quan điểm, và $P(t \mid x)$ là xác suất trích xuất biểu thức thời gian. Mỗi thành phần được ước lượng thông qua các mô đun có thể diễn giải bằng các mô hình như sau:

Mô hình trích xuất thực thể. Các thực thể được trích xuất bằng cách sử dụng nhận dạng thực thể đặt tên (NER) và thông tin tương hỗ theo điểm (PMI) với từ vựng lĩnh vực. Xác suất thực thể được tính theo công thức (1.33).

$$P(e \mid x) = \frac{\text{freq}(e, x) + \lambda_e \cdot \text{PMI}(e, Dom)}{Z_e}, \tag{1.33}$$

Trong đó $\text{freq}(e, x)$ là tần suất xuất hiện của thực thể e trong câu đầu vào, $\text{PMI}(\text{e}, \text{Dom})$ đo lường thông tin tương hỗ theo điểm giữa thực thể e và từ điển chuyên ngành Dom, λ_e là tham số trọng số cân bằng giữa số hạng tần suất và số hạng liên kết lĩnh vực, và Z_e là hằng số chuẩn hóa đảm bảo $\sum_e P(e \mid x) = 1$. Sự kết hợp

này đảm bảo các mẫu xuất hiện quan sát được và tri thức lĩnh vực có trước đều đóng góp vào độ tin cậy của thực thể.

Mô hình trích xuất khía cạnh. Các khía cạnh được trích xuất từ các liên kết phụ thuộc cú pháp đến các thực thể và gán điểm một khía cạnh a liên quan đến thực thể e thông qua tổng hợp dựa trên mối quan hệ trong công thức (1.34).

$$P(a \mid e, x) = \frac{1}{Z_a} \sum_{r \in R} \omega_r \, \mathbb{1}[(e, a) \in r(G)], \tag{1.34}$$

Trong đó $R = \{\text{amod, nsubj, dobj, compound, ...}\}$ là tập hợp các loại quan hệ phụ thuộc cú pháp được trích xuất từ phân tích phụ thuộc, $\omega_r \in [0,1]$ là trọng số độ tin cậy được xác định theo kinh nghiệm cho loại quan hệ r. Ví dụ: $\omega_{\text{amod}} = 0{,}9$ cho bổ ngữ tính từ, $\omega_{\text{nsubj}} = 0{,}7$ cho chủ ngữ danh từ, $1[\cdot]$ là hàm chỉ thị trả về 1 nếu cặp thực thể - khía cạnh (e, a) được kết nối bởi quan hệ r trong đồ thị G và 0 nếu không, Z_a là hằng số chuẩn hóa. Cơ chế dựa trên quy tắc này tận dụng các dấu hiệu phụ thuộc cú pháp rõ ràng để kết nối các mục tiêu quan điểm với các khía cạnh.

Mô hình phân tích tình cảm. Phân cực tình cảm được ước lượng bởi một tổng hợp các mẫu dựa trên từ vựng và ngữ cảnh. Đối với một khía cạnh a trong câu x, điểm tình cảm được tính theo công thức (1.35).

$$P(s \mid e, a, x) = \text{softmax}\big(\alpha_v \, v(x) + \alpha_s \, s_{sw}(a)\big), \tag{1.35}$$

Trong đó $v(x) \in [-1,1]$ là điểm phân cực dựa trên từ điển VADER, được tính trên toàn bộ câu x, nắm bắt tình cảm ngữ cảnh bao gồm phủ định và từ tăng cường, $s_{sw}(a) \in [-1,1]$ là điểm phân cực dựa trên từ điển SentiWordNet cho thuật ngữ khía cạnh a, cung cấp định hướng tình cảm dựa trên từ vựng, $\alpha_v, \alpha_s \geq 0$ là trọng số tổng hợp được điều chỉnh trên tập kiểm định (validation) để cân bằng đóng góp của tình cảm ở cấp độ câu và cấp độ khía cạnh, $\text{softmax}(\cdot)$ chuyển đổi điểm kết hợp thành phân phối xác suất ba lớp là positive, neutral, negative. Tổng hợp này cân bằng thông tin từ vựng và ngữ cảnh, đảm bảo chấm điểm tình cảm có thể diễn giải.

Mô hình người giữ quan điểm. Người giữ quan điểm h được phát hiện thông qua các quan hệ phụ thuộc trên đồ thị phụ thuộc G hoặc nếu không tìm thấy người giữ quan điểm rõ ràng, mặc định là tác giả tài liệu quan điểm là người giữ quan điểm ngầm định, xác định người giữ quan điểm được mô hình hóa bởi công thức (1.36)

$$P(h \mid s, G) = \frac{1}{Z_h} \sum_{r \in \{\text{nsubj,agent}\}} \omega_r \, \mathbb{1}[h = \text{head}(r)] \tag{1.36}$$

Trong đó {nsubj, agent} là các quan hệ ngữ pháp chỉ ra chủ ngữ hoặc tác nhân của một động từ mang quan điểm. Ví dụ: "customer complained", "I think"; head(r) là biểu diễn từ của quan hệ cú pháp r trong cây phụ thuộc, tương ứng với người giữ quan điểm, $\boldsymbol{\omega_r}$ là trọng số riêng của từng quan hệ phản ánh mức độ tin cậy quan hệ r chỉ ra người giữ quan điểm và Z_h là hằng số chuẩn hóa. Công thức này nắm bắt ngữ pháp rõ ràng cho nguồn quan điểm trong khi xử lý cả người giữ quan điểm rõ ràng là các thực thể được đề cập và người giữ quan điểm ngầm định là tác giả.

Mô hình trích xuất thời gian. Biểu thức thời gian được trích xuất bằng cách sử dụng phương pháp kết hợp khớp mẫu dựa trên quy tắc và kết hợp các siêu dữ liệu được biểu diễn trong công thức (1.37).

$$P(t \mid x) = \lambda_t \, P_{\text{rule}}(t \mid x) + (1 - \lambda_t) \, P_{\text{meta}}(t \mid \text{timestamp}) \tag{1.37}$$

Trong đó $P_{\text{rule}}(t \mid x)$ là xác suất trích xuất biểu thức thời gian t từ câu x bằng cách sử dụng các mẫu biểu thức chính quy khớp với các cụm từ thời gian phổ biến, ví dụ: "yesterday", "last week", "2023-02-15". $P_{\text{meta}}(t \mid \text{timestamp})$ là xác suất được trích xuất từ siêu dữ liệu như ghi nhận thời gian đăng quan điểm, cung cấp ngữ cảnh thời gian ngầm định khi biểu thức thời gian không có trong văn bản và $\lambda_t \in [0,1]$ là tham số cân bằng đóng góp của các dấu hiệu ngôn ngữ rõ ràng so với siêu dữ liệu ngầm định. Phương pháp kết hợp này đảm bảo bao phủ cả tham chiếu thời gian rõ ràng và thời gian ngầm định, đặc biệt quan trọng đối với các tài liệu quan điểm trên mạng xã hội có ngữ cảnh thời gian thường ngầm định.

Lan truyền tình cảm có trọng số quan hệ. Để điều chỉnh tình cảm chi tiết, cơ chế lan truyền tình cảm qua đồ thị phụ thuộc $G = (V, E)$. Xác suất tình cảm của mỗi nút v (tương ứng với một từ) được định nghĩa trong công thức (1.38).

$$P(s \mid v) = \sigma\left(\sum_{u \in N(v)} w_{u,v} \, s_u + b_v \right), \tag{1.38}$$

Trong đó $N(v)$ là các nút lân cận của v, s_u là điểm tình cảm tại nút u, $w_{u,v}$ là trọng số mối quan hệ để nắm bắt sự tương đồng về ngữ nghĩa bởi các liên từ như từ "and" hay đối lập về ngữ nghĩa như từ "but", b_v là độ lệch cụ thể của nút và $\sigma(\cdot)$ là hàm sigmoid cho chuẩn hóa. Cơ chế này tổng hợp các biểu diễn quan hệ từ các lân

cận cú pháp và áp dụng chuyển đổi phi tuyến để đảm bảo tình cảm chi tiết mạch lạc về mặt ngữ cảnh.

*Tính toán độ tin cậy***.** Để định lượng độ tin cậy của mỗi bộ năm quan điểm được dự đoán $o = (e, a, s, h, t)$, điểm tin cậy kết hợp C được tính toán bằng cách kết hợp sự đồng thuận của mô hình và độ tin cậy trung bình của các mô đun được biểu diễn bởi các công thức (1.39), (1.40), (1.41).

$$A = 1 - \frac{Var[P_e, P_a, P_s, P_h, P_t]}{4} \tag{1.39}$$

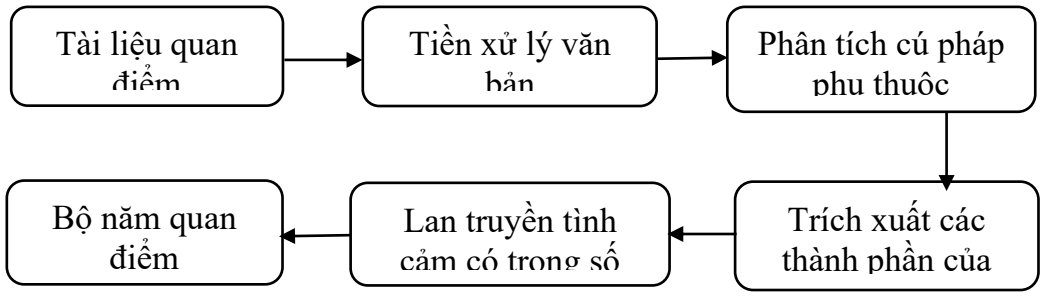
$$\bar{C} = \frac{1}{5} \sum_{m \in \{e,a,s,h,t\}} P_m \tag{1.40}$$

$$C = \frac{A + \bar{C}}{2} \tag{1.41}$$

Trong đó P_m biểu thị xác suất chuẩn hóa được gán bởi mỗi mô đun gồm thực thể, khía cạnh, tình cảm, người giữ quan điểm và thời gian. Số hạng A đo lường tính nhất quán giữa các mô đun, trong khi $\bar{C}$ phản ánh sự chắc chắn tổng thể của chúng. Quyết định cuối cùng chỉ giữ lại các bộ năm quan điểm với $C \geq \tau$, trong đó τ là ngưỡng được hiệu chỉnh trên tập kiểm định.

1.6.2. Các giai đoạn trích xuất bộ năm quan điểm

Trích xuất bộ năm quan điểm là quá trình chuyển đổi văn bản đánh giá thành bộ năm quan điểm có cấu trúc bao gồm 06 giai đoạn: (1) Tài liệu quan điểm; (2) Tiền xử lý văn bản; (3) Phân tích cú pháp phụ thuộc; (4) Trích xuất thành phần ban đầu; (5) Lan truyền tình cảm có trọng số quan hệ; (6) Bộ năm quan điểm.



Hình 1.8: Trích xuất bộ năm quan điểm

Hình 1.8 minh họa sơ đồ của hệ thống trích xuất bộ năm quan điểm bao gồm:

- *Tài liệu quan điểm*: Dưới dạng các đánh giá đầu vào x, mỗi đánh giá là tài liệu quan điểm độc lập.

• *Tiền xử lý văn bản*: Thực hiện tách từ, phân tích thành các thành phần, loại bỏ từ dừng và gắn thẻ POS. URL, hashtag và biểu tượng cảm xúc được lọc để tạo ra các chuỗi ngôn ngữ sạch.

• *Phân tích cú pháp phụ thuộc*: Các đánh giá được phân tích thành đồ thị phụ thuộc $G = (V, E)$ nắm bắt các quan hệ cú pháp và trọng số ω_r.

• *Trích xuất thành phần ban đầu*: Các bộ năm quan điểm (e, a, s, h, t) được tạo ra bằng cách sử dụng các mô hình xác suất được định nghĩa trong công thức (1.33) – (1.37), với mỗi thành phần được ước lượng độc lập.

• *Lan truyền tình cảm có trọng số quan hệ:* Dự đoán tình cảm được điều chỉnh thông qua đồ thị phụ thuộc sử dụng công thức (1.38), đảm bảo tính mạch lạc ngữ cảnh.

• *Bộ năm quan điểm*: Các thành phần được kết hợp thành các bộ năm quan điểm hoàn chỉnh với độ tin cậy được tính toán thông qua các công thức (1.39)–(1.41). Chỉ các bộ năm quan điểm có độ tin cậy $\text{Conf}(o) \geq \tau$ được giữ lại làm đầu ra cuối cùng.

Quá trình trích xuất bộ năm quan điểm được trình bày trong *Thuật toán 1*, tích hợp mô hình trích xuất với phân tích cú pháp phụ thuộc dựa trên quan hệ và lọc độ tin cậy. Thuật toán xây dựng các quan hệ phụ thuộc, xác định các cặp thực thể - khía cạnh, và tích hợp các thành phần tình cảm, người giữ quan điểm và thời gian để tạo thành bộ năm quan điểm.

Thuật toán 1: Trích xuất bộ năm quan điểm

Đầu vào: Tập tài liệu $D = \{d_1, d_2, \ldots d_n\}$, từ vựng lĩnh vực Dom, ngưỡng τ

Đầu ra: Tập bộ năm quan điểm $O = \{o_1, o_2, \ldots o_m\}$

```
O ← ∅
for each d_i ∈ D do
    G ← ParseDependencies(d_i)        // Xây đồ thị phụ thuộc G=(V,E)
    E ← ExtractEntities(G, Dom)       // Công thức (1.33)
    A ← ExtractAspects(G, E)          // Công thức (1.34)
    S ← EnsembleSentiment(d_i, A)     // Công thức (1.35)
    H ← ExtractHolders(G, S)          // Công thức (1.36)
```

```
        T ← ExtractTime(d_i)              // Công thức (1.37)
        S ← PropagateWithGraph(G, S)      // Lan truyền tình cảm, CT (1.38)
        for each nút s mang tình cảm ∈ S do
            R ← FindRelations(G, s)
            for each quan hệ r ∈ R do
                o ← (e ∈ E, a ∈ A, s ∈ S, h ∈ H, t ∈ T) theo r
                Conf(o) ← ComputeConfidence(o)   // Công thức (1.39)–(1.41)
                if Conf(o) ≥ τ then
                    O ← O ∪ {o}
                end if
            end for
        end for
    end for
    return O
```

1.6.3. Đánh giá

1.6.3.1. Các độ đo đánh giá

Hiệu suất của mô hình đề xuất được đánh giá ở cả cấp độ thành phần và cấp độ bộ năm quan điểm, sử dụng độ chính xác (P) công thức (1.21), độ thu hồi (R) công thức (1.22) và độ đo F_1 công thức (1.23) theo tiêu chí khớp chính xác bộ năm quan điểm.

Mỗi bộ năm quan điểm dự đoán $(\hat{e}, \hat{a}, \hat{s}, \hat{h}, \hat{t})$ được coi là đúng khi tất cả các thành phần trong bộ năm quan điểm đều khớp chính xác.

1.6.3.2. Độ đo đồng thuận chú thích

Để đảm bảo độ tin cậy của các chú thích bộ năm quan điểm, hệ số Kappa của Cohen [79][198][199] được sử dụng để đo lường sự đồng thuận giữa các người đánh giá. Kappa của Cohen được định nghĩa theo công thức (1.42). Với ba người chú thích độc lập, κ được tính cho từng cặp người chú thích và lấy trung bình (Hình 1.9); hệ số Fleiss' κ [199] là phương án tương đương cho trường hợp nhiều người chú thích.

$$\kappa = \frac{P_o - P_e}{1 - P_e}, \tag{1.42}$$

Trong đó P_o biểu thị sự đồng thuận quan sát được, đại diện cho tỷ lệ các trường hợp mà người đánh giá gán cùng một nhãn, và P_e đại diện cho sự đồng thuận dự kiến theo ngẫu nhiên.

Sự đồng thuận dự kiến được tính theo công thức (1.43).

$$P_e = \sum_i P_{A,i} \times P_{B,i} \,, \tag{1.43}$$

Trong đó $P_{A,i}$ và $P_{B,i}$ là xác suất cận biên mà người đánh giá A và B gán danh mục i, tương ứng.

1.6.4. Thực nghiệm

1.6.4.1. Tập dữ liệu

Tập dữ liệu Twitter US Airline Sentiment [80] được sử dụng cho khai phá quan điểm trên các văn bản ngắn do người dùng tạo ra. Tập hợp chứa các tweet hướng đến các hãng hàng không lớn của Hoa Kỳ (United, Delta, American, Southwest, Virgin America và US Airways), mỗi tweet được gắn nhãn với phân cực cảm xúc (positive, neutral hoặc negative) và đi kèm với siêu dữ liệu như tác giả, dấu thời gian và hãng hàng không đích. Bảng 1.5 trình bày thống kê tóm tắt của tập dữ liệu Twitter.

Bảng 1.5: Thống kê tóm tắt của tập dữ liệu Twitter US Airline Sentiment

Danh mục	**Số lượng**	**Phần trăm (*%*)**
Tổng số tweet	14.640	100,0
Tweet được chú thích bộ năm	4.500	30,7
Tweet tích cực	2.364	16,1
Tweet trung lập	3.090	21,1
Tweet tiêu cực	9.186	62,7
Thực thể hãng hàng không riêng biệt	6	-
Số token trung bình mỗi tweet	18,7	-

1.6.4.2. Công cụ

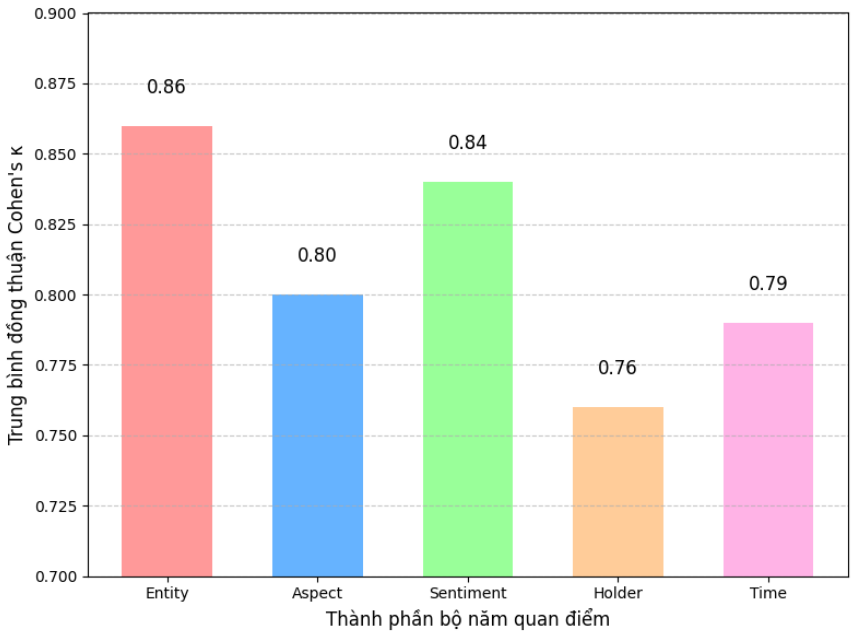
Riêng mô hình trích xuất bộ năm tại mục 1.6 (kết quả của Bài báo số 8) được triển khai bằng ngôn ngữ lập trình Python, sử dụng các thư viện chính bao gồm spaCy [82] cho tách từ, POS và phân tích cú pháp phụ thuộc; NLTK và SentiWordNet cho truy cập từ vựng; VADER cho cảm xúc tweet, và Label Studio cho chú thích [130]. Toàn bộ quá trình được thực hiện trong môi trường Google Colab, với cả kết quả trung gian và cuối cùng được lưu trữ ở các định dạng có cấu trúc (tệp CSV bộ năm)

để tạo điều kiện cho phân tích tiếp theo. Bước trích xuất này là bước tiền xử lý dữ liệu; toàn bộ các mô hình rừng đề xuất, các mô hình so sánh và các thực nghiệm bổ sung tại Chương 2, Chương 3, Chương 4 và các phụ lục đều được triển khai thống nhất bằng ngôn ngữ R trên các tệp bộ năm đầu ra của bước này (mục 2.8, 3.8, 4.9).

1.6.4.3. Các kịch bản

Kịch bản 1: Chú thích và chuẩn bị dữ liệu

Kịch bản đã thiết lập nền tảng cho khai phá quan điểm chi tiết bằng cách xây dựng một tập dữ liệu được chú thích. Công cụ Label Studio được sử dụng để chú thích thủ công 4.500 tweet từ tập dữ liệu Twitter US Airline Sentiment. Ba người chú thích độc lập gắn nhãn mỗi đánh giá theo các hướng dẫn chi tiết xác định phạm vi thực thể, khía cạnh, phân cực tình cảm, người giữ quan điểm là rõ ràng hoặc ngầm định và thời gian lúc đưa ra quan điểm.



Hình 1.9: Trung bình đồng thuận chú thích

Phân tích sự đồng thuận giữa những người chú thích trong Hình 1.9 cho thấy độ tin cậy cao nhất quán trên hầu hết các thành phần, với Cohen's $\kappa > 0{,}79$ cho thực thể, khía cạnh và tình cảm, cho thấy người chú thích đạt được sự đồng thuận mạnh về việc xác định các thành phần thực thể, khía cạnh, tình cảm trong bộ năm quan điểm. Thành phần Người giữ quan điểm có sự đồng thuận tốt ($\kappa = 0{,}76$) do nguồn quan điểm ngầm định hoặc được suy diễn trong khi biểu thức thời gian cũng đạt mức đồng thuận

tốt ($\kappa = 0{,}79$). Tập dữ liệu được chia bằng cách sử dụng phân chia tỉ lệ là 70–15–15 thành các tập huấn luyện, kiểm định và kiểm tra, có sự cân bằng các nhãn tình cảm.

Kịch bản 2: Trích xuất bộ năm quan điểm từ đánh giá

Trong kịch bản này, hệ thống trích xuất các bộ năm quan điểm hoàn chỉnh từ các tweet riêng lẻ bằng cách sử dụng mô hình xác suất mối quan hệ. Mỗi tweet được phân tích độc lập để đánh giá khả năng suy luận các mối quan hệ trong câu giữa các thành phần quan điểm của mô hình. Hiệu suất được đo lường thông qua độ chính xác, độ thu hồi và F1 theo tiêu chí khớp chính xác cho mỗi thành phần, được minh họa trong Bảng 1.6.

Bảng 1.6: Hiệu suất theo từng thành phần

Thành phần	**Precision**	**Recall**	**F1**
Entity (Thực thể)	0,88	0,85	0,86
Aspect (Khía cạnh)	0,81	0,78	0,79
Sentiment (Tình cảm)	0,82	0,80	0,81
Holder (Người giữ quan điểm)	0,69	0,67	0,68
Time (Thời gian)	0,74	0,72	0,73
Overall (Tổng thể)	0,81	0,75	0,78

Kết quả cho thấy độ chính xác cao về thực thể, khía cạnh và tình cảm với hiệu suất thấp hơn một chút đối với người giữ quan điểm ngầm định và biểu thức thời gian. Điểm F1 tổng thể là 0,78 được tính là trung bình điều hòa của độ chính xác (0,81) và độ thu hồi (0,75) tổng hợp theo thành phần tại Bảng 1.6 (Bảng 1.7 ghi 0,80/0,76 theo bảng so sánh của Bài báo 8; F1 ở cả hai cách tổng hợp đều bằng 0,78), điều đó cho thấy mô hình đề xuất duy trì hiệu suất cân bằng trên tất cả các thành phần của bộ năm quan điểm.

Kịch bản 3: Dự đoán và so sánh

Kịch bản này đánh giá khả năng dự đoán đầy đủ của mô hình so với các phương pháp cơ sở tiêu chuẩn như trích xuất quan điểm dựa trên quy tắc, trích xuất quan điểm

phụ thuộc dựa trên thư viện spaCy, và các mô hình học chuyển giao cơ sở như BERT và RoBERTa. Tất cả các mô hình được huấn luyện và kiểm tra trên tập con được chú thích và đánh giá trên cùng một quy trình tiền xử lý.

Bảng 1.7: So sánh với các mô hình cơ sở

Mô hình	Precision	Recall	F1	Time/tweet (ms)	Memory (MB)
Dựa trên quy tắc	0,68	0,65	0,66	5,2	120
spaCy	0,71	0,69	0,70	7,4	160
BERT (baseline)	0,76	0,75	0,75	21,3	850
RoBERTa (baseline)	0,77	0,76	0,76	23,5	910
Mô hình đề xuất	0,80	0,76	0,78	8,6	210

Như được trình bày trong Bảng 1.7, mô hình đề xuất đạt được sự cân bằng giữa độ chính xác và hiệu quả tính toán. Mô hình đề xuất đạt điểm F1 là 0,78; trên cùng tập dữ liệu, các mô hình học chuyển giao RoBERTa và BERT đạt lần lượt 0,76 và 0,75. Mô hình đề xuất sử dụng 210 MB bộ nhớ và 8,6 ms mỗi tweet; RoBERTa sử dụng 910 MB và 23,5 ms mỗi tweet. So với các phương pháp dựa trên quy tắc truyền thống và spaCy, F1 của phương pháp đề xuất là 0,78 so với 0,66 và 0,70, đồng thời duy trì khả năng diễn giải và xử lý thời gian thực.

Dựa vào ba kịch bản, mô hình đề xuất cho thấy tính ổn định, khả năng diễn giải và hiệu quả tính toán. Kịch bản 1 xác thực độ tin cậy của dữ liệu được chú thích. Kịch bản 2 xác nhận trích xuất chính xác ở cấp độ thành phần, đặc biệt là đối với các tweet có nhiều khía cạnh và kịch bản 3 cho thấy khung đề xuất đạt độ chính xác cao với chi phí tài nguyên thấp (Bảng 1.7). Mô hình đề xuất đạt F1=0,78, xác nhận tính phù hợp cho các nhiệm vụ khai phá quan điểm trong thế giới thực với lý thuyết mối quan hệ có thể diễn giải được.

1.7. Kết luận

Chương 1 đã xây dựng cơ sở lý luận cho toàn bộ luận án thông qua việc xác lập mô hình bộ năm thành phần quan điểm *o = (e, a, s, h, t)* làm đơn vị biểu diễn thống nhất của dữ liệu quan điểm, hệ thống hóa các phương pháp học quần thể dựa trên cây

quyết định và làm rõ sự khác biệt giữa ba tiêu chí entropy Shannon, entropy Gaussian và entropy Gaussian trên đa tạp được sử dụng trong ba mô hình đề xuất ở các chương tiếp theo. Đồng thời, chương đã tổng quan các công trình nghiên cứu liên quan, xác định ba khoảng trống nghiên cứu về xử lý dữ liệu mất cân bằng, khai thác cấu trúc đa tạp của dữ liệu quan điểm và nâng cao khả năng giải thích của mô hình. Bên cạnh đó, quy trình trích xuất bộ năm thành phần quan điểm đạt F1-score = 0,78, là nền tảng cho việc xây dựng bộ năm từ văn bản trên các miền dữ liệu khác; trong các chương tiếp theo, bộ năm của các tập Amazon, TripAdvisor và Yelp được xây dựng theo quy trình chuyển đổi trình bày tại mục 2.7.2, 3.7.2 và Phụ lục C. Trên nền tảng đó, Chương 2 trình bày mô hình đề xuất đầu tiên là rừng phân loại quan điểm F_{CF}, xây dựng trực tiếp trên ma trận bộ năm.

CHƯƠNG 2. KHAI PHÁ QUAN ĐIỂM VỚI RỪNG PHÂN LOẠI

2.1. Giới thiệu

Chương này trình bày kết quả nghiên cứu, phát triển và xây dựng mô hình, thuật toán khai phá quan điểm với rừng phân loại (rừng ngẫu nhiên), để giải quyết bài toán khai phá quan điểm trên dữ liệu đánh giá. Mô hình này có khả năng kết hợp nhiều cây quyết định được xây dựng trên các tập con thuộc tính ngẫu nhiên, từ đó tận dụng hiệu quả các đặc trưng đa chiều của dữ liệu đánh giá bao gồm: tên sản phẩm, khía cạnh sản phẩm, người đánh giá, thời gian đánh giá và tình cảm. Kết quả cho thấy mô hình đề xuất có khả năng khám phá các đặc trưng quan trọng của các đánh giá và giải quyết vấn đề ảnh hưởng kết hợp theo tầm quan trọng của các lớp quan điểm khác nhau. Ngoài ra, kết quả đã phân tích và diễn giải ảnh hưởng của các tham số quan trọng liên quan đến quá trình ra quyết định của mô hình rừng phân loại. Mô hình được thực nghiệm trên cả bộ dữ liệu đánh giá thực phẩm và bộ dữ liệu đánh giá khách sạn.

Nội dung chính của chương bao hàm kết quả nghiên cứu của Bài báo số 2, 3.

Phát biểu bài toán. Cho tập dữ liệu quan điểm $O = \{o_1, o_2, o_3, \ldots, o_N\}$, mỗi $o_i = (e_i, a_i, s_i, h_i, t_i)$ gồm thực thể e_i, khía cạnh a_i, giá trị tình cảm s_i, người giữ quan điểm h_i và thời điểm t_i. Sau bước mã hóa số §2.2, dữ liệu được biểu diễn thành ma trận $O_{mat} \in \mathbb{R}^{\mathrm{Nx5}}$ theo công thức (2.2), trong đó s_i là cột nhãn và (e_i, a_i, h_i, t_i) là bốn đặc trưng đầu vào.

Đầu vào: Ma trận $O_{mat} \in \mathbb{R}^{\mathrm{Nx5}}$.

Đầu ra: Nhãn phân loại $\hat{y} \in C = \{\text{positive, negative}\}$, học từ $\{(e_i, a_i, s_i, h_i, t_i)\}_{i=1}^{N}$, sao cho với quan điểm o, mô hình dự đoán $\hat{y} = f_{F_{CF}}(o)$ xấp xỉ đúng nhãn thực tế $y = s$.

Mục tiêu: Tối thiểu hóa sai số phân loại $\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} 1(\hat{y}_i \neq s_i)$ trên tập kiểm tra, đảm bảo khả năng diễn giải của mô hình thông qua cấu trúc cây quyết định.

2.2. Ma trận bộ năm quan điểm

Một bộ năm quan điểm bao gồm năm thành phần, được ký hiệu là $o = (e, a, s, h, t)$ được trình bày tại Mục 1.2.1.

Cho tập hợp N bộ năm quan điểm:

$$O = \{o_1, o_2, o_3, \ldots, o_N\}, \quad (2.1)$$

Trong đó, $o_i = (e_i, a_i, s_i, h_i, t_i)$, i = 1, 2, ..., N là mỗi bộ năm quan điểm.

Ma trận bộ năm quan điểm O_{mat} có kích thước $N \times 5$ được định nghĩa trong công thức (2.2)

$$O_{mat} = \begin{bmatrix} e_1 & a_1 & s_1 & h_1 & t_1 \\ e_2 & a_2 & s_2 & h_2 & t_2 \\ e_3 & a_3 & s_3 & h_3 & t_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ e_N & a_N & s_N & h_N & t_N \end{bmatrix}, \quad (2.2)$$

Trong đó, N là số hàng tương ứng với N bộ năm quan điểm, 5 là số cột tương ứng với năm thành phần của bộ năm quan điểm, cột 1 là thực thể có các giá trị e_i, cột 2 là khía cạnh có các giá trị a_i, cột 3 là cột nhãn về tình cảm có các giá trị s_i, cột 4 là người giữ quan điểm có các giá trị h_i, cột 5 là thời gian có các giá trị t_i.

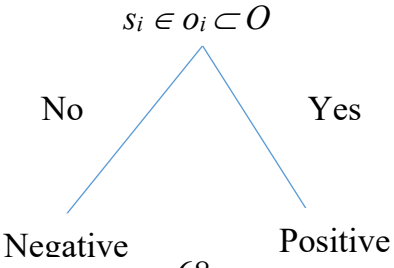
Ma trận bộ năm quan điểm trong công thức (2.2) được gọi là ma trận đặc trưng quan điểm làm đầu vào cho quá trình xây dựng mô hình rừng phân loại quan điểm để khai phá quan điểm.

Ma trận O_{mat} (N×5) đóng vai trò là cấu trúc lưu trữ thống nhất cho cả năm thành phần của bộ năm quan điểm, được tái sử dụng xuyên suốt Chương 2, 3 và 4. Khi bước vào giai đoạn huấn luyện mô hình, cột nhãn s được tách ra làm biến mục tiêu y ngay tại bước tiền xử lý, trước khi đưa vào xây dựng cây quyết định tại mục 2.3; phần còn lại (e, a, h, t) mới là ma trận đặc trưng đầu vào N×4 thực sự cho mô hình. Như vậy, O_{mat} (N×5) khác với một bảng bốn đặc trưng thông thường ở chỗ nó là cấu trúc dữ liệu trung gian lưu đồng thời cả đặc trưng và nhãn, thuận tiện cho việc chia tập Train/Validation/Test một lần và tái sử dụng nhất quán qua ba mô hình rừng đề xuất.

2.3. Mô hình rừng phân loại quan điểm

2.3.1. Cây phân loại quan điểm

Cho tập hợp các bộ năm quan điểm $O = (e_1, a_1, s_1, h_1, t_1), (e_2, a_2, s_2, h_2, t_2), \ldots, (e_N, a_N, s_N, h_N, t_N)$, xây dựng mô hình cây quyết định [61][60] bằng cách phân chia gửi một tập hợp con của các mẫu $o_i \in O$ sang bên trái và các mẫu còn lại bên phải trong Hình 2.1.



Hình 2.1: Phân tách biến dự báo

Cây phân loại quan điểm T_i được xây dựng bằng cách phân chia một nút thành hai nút bằng cách xem xét sự phân chia trên mỗi biến dự đoán và chọn sự phân chia tốt nhất làm tiêu chí phân chia (Hình 2.1).

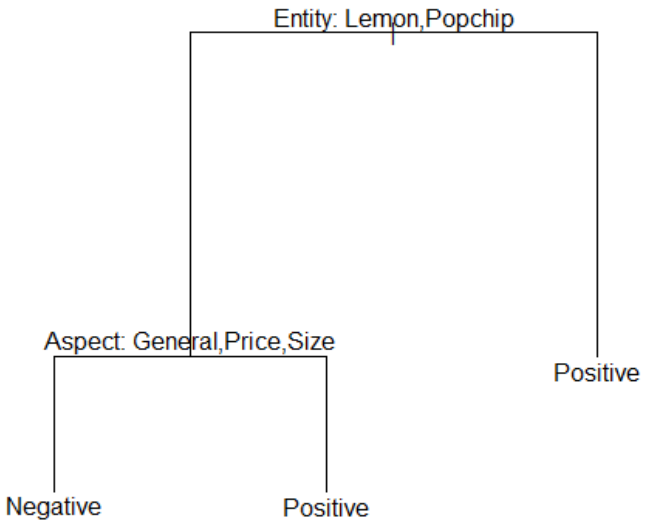
Trong trường hợp phân loại trong đó có $c \in 1, 2, \dots, C$ một tiêu chí phân tách điển hình bằng cách sử dụng Entropy Shannon [61] trong công thức (1.4) để được công thức (2.3).

$$H(O) = -\sum_{c \in C} p(c)\, log\big(p(c)\big) \tag{2.3}$$

Trong đó O là tập các bộ năm quan điểm và c là một lớp (nhãn tình cảm). Tập hợp của tất cả các lớp được ký hiệu là C. Một độ đo tinh khiết của nút bằng cách đo mức tăng thông tin theo công thức (2.4).

$$IG = H(O) - \sum_{i \in \{L,R\}} \frac{|O^i|}{|O|} H\left(O^i\right) \tag{2.4}$$

Cây phân loại quan điểm bao gồm các luật phân loại quan điểm. Đường dẫn từ nút gốc đến nút lá được biểu diễn dưới dạng gốc → lá. Theo quy tắc, bên phải ký hiệu "→" có một giá trị và bên trái ký hiệu "→" có ít nhất một giá trị. Kí hiệu "∩" là dấu giao nhau. Ví dụ, trong Hình 2.2 có quy tắc quyết định quan điểm về thực phẩm là (Entity = *Lemon*) ∩ (Aspect = *Price*) → Opinion = *Negative.*



Hình 2.2: Cây quyết định quan điểm về thực phẩm

2.3.2. Dựng rừng phân loại quan điểm

Rừng [65][66] phân loại quan điểm được xây dựng bằng cách kết hợp nhiều cây quyết định quan điểm trên. Mỗi cây được huấn luyện độc lập trên một tập mẫu bootstrap khác nhau, tạo ra sự đa dạng giữa các cây quan điểm và giảm thiểu trường hợp quá phù hợp với mô hình, các bước xây dựng rừng:

Bước 1: Lấy mẫu Bootstrap

Cho tập dữ liệu quan điểm $O = \{o_1, o_2, \dots, o_N\}$, trong đó $o_i = (e_i, a_i, s_i, h_i, t_i)$ là bộ năm quan điểm thứ i.

Với mỗi cây quyết định thứ i ($i = 1,2, \dots, n$), tạo tập mẫu Bootstrap O_i^* theo các bước sau:

Bước 1.1: Lấy ngẫu nhiên N mẫu từ tập dữ liệu O theo phương pháp lấy mẫu có hoàn lại.

Bước 1.2: Trong mỗi lần lấy mẫu, mỗi phần tử $o_i \in O$ có xác suất $1/N$ được chọn.

Bước 1.3: Kết quả là tập $O_i^* = \{o_{k_1}, o_{k_2}, \dots, o_{k_N}\}$ với $k_i \in \{1,2, \dots, N\}$.

Dựa vào tính chất thống kê của phương pháp Bootstrap, xác suất một mẫu o_i xuất hiện ít nhất một lần trong O_i^* được xác định theo công thức (2.5).

$$P\left(o_i \in O_i^*\right) = 1 - \left(1 \quad \frac{1}{N}\right)^N \to 1 - e^{-1} \approx 0{,}632 \tag{2.5}$$

Ngược lại, xác suất một mẫu không được chọn trong O_i^* được xác định theo công thức (2.6).

$$P(o_i \notin O_i^*) = \left(1 \quad \frac{1}{N}\right)^N \to e^{-1} \approx 0{,}368 \tag{2.6}$$

Các mẫu không xuất hiện trong O_i^* được gọi là tập Out-of-Bag của cây thứ i được biểu diễn trong công thức (2.7).

$$\text{OOB}_i = \{o_i \in O \mid o_i \notin O_i^*\}\ , \tag{2.7}$$

Tập OOB có thể được sử dụng để ước lượng lỗi tổng quát hóa mà không cần sử dụng tập xác thực riêng biệt.

Bước 2: Lựa chọn đặc trưng ngẫu nhiên

Tại mỗi nút trong quá trình xây dựng cây thứ i:

Bước 2.1: Chọn ngẫu nhiên m_{try} đặc trưng từ tập đặc trưng $\chi = \{e, a, h, t\}$.

Bước 2.2: Tìm phép phân tách tối ưu trên tập m_{try} đặc trưng đã chọn.

Bước 2.3: Thực hiện phân tách và tiếp tục đệ quy cho các nút con.

Dựa vào [65] giá trị m_{try} thường được chọn là $m_{try} \approx \sqrt{|\chi|}$. Trong bài toán phân loại quan điểm tập đặc trưng gồm $|\chi| = 4$, do đó chọn $m_{try} = 2$. Tập đặc trưng được chọn tại mỗi nút được biểu diễn như công thức (2.8).

$$\chi_{m_{try}} \subset \chi = \{e, a, h, t\}, |\ \chi_{m_{try}}\ | = m_{try}, \tag{2.8}$$

Bước 3: Huấn luyện các cây quyết định

Với mỗi $i = 1, 2, \dots, n$

Bước 3.1: Tạo tập Bootstrap: $O_i^* \leftarrow \text{BootstrapSample}(O, N)$

Trong đó:

- O_i^*: là tập dữ liệu bootstrap thứ *i* được lấy mẫu ngẫu nhiên có hoàn lại từ tập dữ liệu huấn luyện O.
- $N = |O|$: số lượng mẫu dữ liệu quan điểm.

Bước 3.2: Xây dựng cây quyết định:

$$T_i \leftarrow \text{BuildTree}(O_i^*, m_{try})$$

Tại mỗi nút của cây:

- Chọn ngẫu nhiên m_{try} đặc trưng từ tập đặc trưng:

$$\chi = \{e, a, h, t\}$$

- Tính Entropy ở công thức (2.3) và độ lợi thông tin (Information Gain) ở công thức (2.4).
- Tìm phép phân tách tốt nhất và thực hiện tách dữ liệu.
- Tiếp tục thực hiện đệ quy cho các nút con đến khi đạt điều kiện dừng.

Bước 3.3: Lưu cây quyết định:

$$F_{CF} \leftarrow F_{CF} \cup \{T_i\}$$

Bước 4: Tạo rừng phân loại quan điểm

Rừng phân loại quan điểm bao gồm n cây quyết định trong công thức (2.9) có cơ chế Bootstrap Aggregating (Bagging) [185], nền tảng của rừng phân loại ngẫu nhiên đã được phân tích lý thuyết và ứng dụng toàn diện, trong đó Bagging giảm phương sai dự đoán bằng cách huấn luyện nhiều cây trên các tập con bootstrap khác

nhau và tổng hợp qua bầu chọn đa số, là cơ sở lý thuyết trực tiếp cho thuật toán xây dựng rừng phân loại quan điểm được mô tả trên.

$$F_{CF} = \{T_1, T_2, \dots, T_n\}, \quad (2.9)$$

Trong đó

- $T_i: \chi \to C$: là một hàm ánh xạ từ không gian đặc trưng đến không gian lớp tình cảm;
- $\chi = \{e, a, h, t\}$: là không gian đặc trưng;
- $C = \{\text{positive, negative}\}$: là tập lớp tình cảm.

2.3.3. Dự đoán rừng phân loại quan điểm

Sau khi xây dựng được rừng phân loại:

$$F_{CF} = \{T_1, T_2, \dots, T_n\},$$

Việc dự đoán cho một mẫu quan điểm mới được thực hiện bằng phương pháp biểu quyết đa số (Majority Voting).

Cho mẫu đầu vào:

$$x = (e, a, h, t),$$

Mỗi cây quyết định T_i trong rừng đưa ra một dự đoán lớp tình cảm cho mẫu x.

$$T_1(x), T_2(x), \dots, T_n(x)\ , \quad (2.10)$$

Trong đó:

- $T_i(x)$ là dự đoán của cây thứ i đối với mẫu x
- n là tổng số cây trong rừng.

Dự đoán cuối cùng của mô hình rừng phân loại được xác định bằng lớp có số phiếu biểu quyết lớn nhất (2.11).

$$\hat{y} = f(x) = argmax_{c \in C} \sum_{n=1}^{n} I(T_i(x) = c), \quad (2.11)$$

Trong đó:

- $\hat{y} = f(x)$: là nhãn tình cảm dự đoán cuối cùng của rừng phân loại đối với mẫu x.
- C là tập lớp tình cảm, C ={positive, negative},

- $I(.)$ là hàm chỉ thị, $I(T_i(x) = c) = 1$ nếu cây T_i dự đoán lớp c, $I(T_i(x) = c) = 0$ nếu ngược lại.
- $argmax$: toán tử chọn lớp có tổng số phiếu lớn nhất.

2.4. Mô hình tổng quan khai phá quan điểm với rừng phân loại

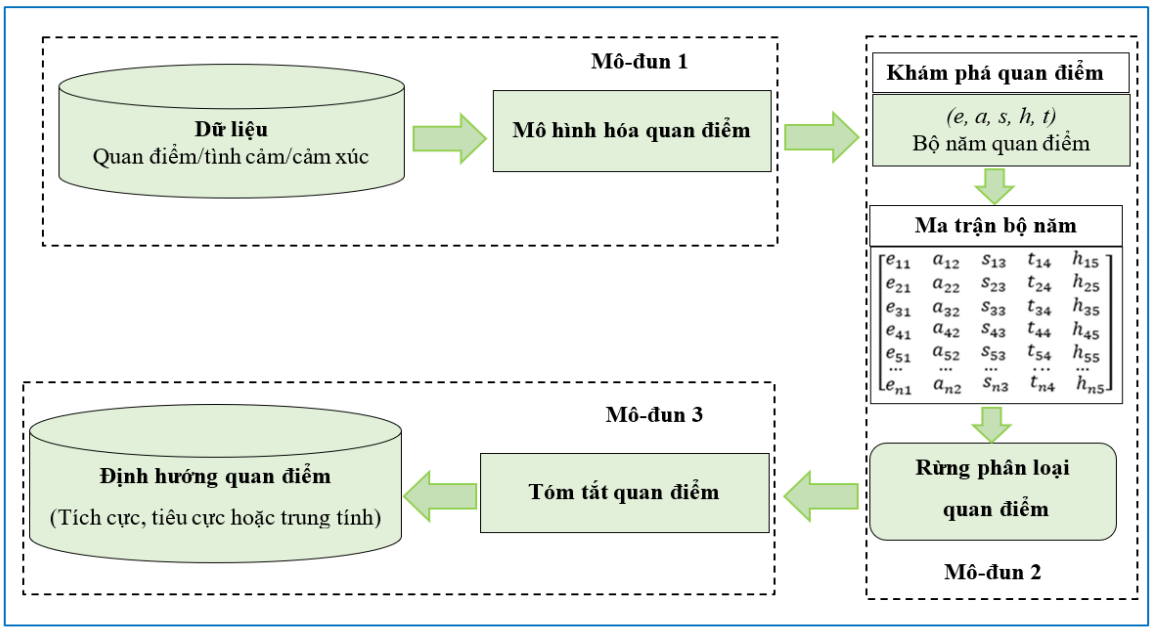
2.4.1. Tổng quan mô hình

Mô hình khai phá quan điểm với rừng phân loại được đề xuất thực hiện trên ba mô-đun chính, tạo thành một quy trình hoàn chỉnh từ dữ liệu đầu vào đến kết quả phân loại cuối cùng. Kiến trúc ba mô-đun này được thiết kế nhằm tách biệt các giai đoạn xử lý, giúp mô hình dễ dàng mở rộng, bảo trì và tối ưu hóa từng thành phần độc lập. Sơ đồ tổng quát được minh họa trong Hình 2.3, trong đó:

Mô-đun 1 (Mô hình hóa quan điểm): Chịu trách nhiệm tiếp nhận và chuẩn hóa quan điểm đầu vào từ các nguồn đánh giá khác nhau như đánh giá sản phẩm, đánh giá dịch vụ. Văn bản thô bao gồm quan điểm, tình cảm và cảm xúc của người dùng được chuyển đổi thành định dạng chuẩn để phục vụ cho các bước xử lý tiếp theo.

Mô-đun 2 (Khám phá và biểu diễn quan điểm): Thực hiện việc khám phá, trích xuất các mối quan hệ giữa các đặc trưng và chuyển đổi dữ liệu thành dạng bộ năm quan điểm *(e, a, s, h, t)*. Sau đó, các bộ năm này được tổ chức thành ma trận đặc trưng, tạo đầu vào chuẩn cho việc xây dựng mô hình rừng phân loại quan điểm.

Mô-đun 3 (Tóm tắt và định hướng quan điểm): Sử dụng kết quả từ mô hình rừng phân loại để tóm tắt quan điểm và đưa ra định hướng cuối cùng về quan điểm là tích cực, tiêu cực hoặc trung tính.



Hình 2.3**:** Khai phá quan điểm với rừng phân loại

2.4.2. Giai đoạn mô hình hóa quan điểm

Mô-đun 1 đóng vai trò là đầu vào của toàn bộ hệ thống, thực hiện việc mô hình hóa quan điểm từ dạng văn bản thô ban đầu sang dạng có cấu trúc. Quá trình này bao gồm các bước chính:

Bước 1. Thu thập dữ liệu: Dữ liệu đầu vào được thu thập từ các nguồn đánh giá trực tuyến hoặc các nền tảng thương mại điện tử khác. Dữ liệu thô thường ở dạng văn bản tự do, kèm theo các thông tin meta như người đánh giá, thời gian đánh giá, điểm số đánh giá.

Bước 2. Tiền xử lý dữ liệu: Các đánh giá văn bản được làm sạch bằng cách loại bỏ các ký tự đặc biệt, số, và các từ dừng (stop words) không mang ý nghĩa. Quá trình phân tách được thực hiện để tách các từ trong câu thành các từ riêng biệt.

Bước 3. Chuẩn hóa dữ liệu: Các thông tin được chuẩn hóa về định dạng thống nhất, bao gồm: tên sản phẩm/dịch vụ là các thực thể, các khía cạnh được đánh giá, người đánh giá, thời gian đánh giá và định hướng quan điểm ban đầu.

Bước 4. Xử lý mất cân bằng dữ liệu: Dữ liệu đánh giá thực tế thường có xu hướng lệch về các đánh giá tích cực, mô-đun này thực hiện việc cân bằng dữ liệu bằng cách phân loại lại các xếp hạng.

2.4.3. Giai đoạn khám phá và biểu diễn quan điểm

Mô-đun 2 là thành phần cốt lõi của hệ thống, thực hiện ba nhiệm vụ là khám phá quan điểm, biểu diễn dưới dạng ma trận và xây dựng mô hình rừng phân loại.

Bước 1. Khám phá quan điểm: Quá trình này trích xuất các thành phần quan trọng từ mỗi đánh giá để tạo thành bộ năm quan điểm *(e, a, s, h, t).*

Bước 2. Biểu diễn dưới dạng ma trận: Các bộ năm quan điểm được tổ chức thành ma trận đặc trưng N×5, trong đó N là số lượng bộ năm quan điểm và 5 cột tương ứng với năm thành phần của bộ năm. Ma trận này được gọi là ma trận đặc trưng quan điểm và là đầu vào trực tiếp cho thuật toán rừng phân loại.

Bước 3. Xây dựng mô hình rừng phân loại: Sử dụng ma trận đặc trưng quan điểm, mô-đun này xây dựng mô hình rừng phân loại thông qua các bước:

Bước 3.1: Tạo các mẫu Bootstrap từ tập dữ liệu gốc

Bước 3.2: Với mỗi mẫu Bootstrap, xây dựng một cây quyết định với việc lựa chọn ngẫu nhiên tập con các đặc trưng tại mỗi nút phân chia

Bước 3.3: Kết hợp các cây quyết định thành rừng phân loại

Bước 3.4: Sử dụng phương pháp biểu quyết đa số để đưa ra dự đoán cuối cùng

2.4.4. Giai đoạn định hướng quan điểm

Mô-đun 3 thực hiện giai đoạn cuối cùng của quy trình, tập trung vào việc tóm tắt kết quả và đưa ra định hướng quan điểm rõ ràng, bao gồm các bước:

Bước 1. Tóm tắt quan điểm: Hệ thống tạo ra các báo cáo tóm tắt về tỷ lệ phần trăm các quan điểm tích cực, tiêu cực, trung tính; Các khía cạnh được đánh giá tích cực nhất và tiêu cực nhất; Xu hướng thay đổi quan điểm theo thời gian; So sánh giữa các sản phẩm/dịch vụ khác nhau.

Bước 2. Định hướng quan điểm: Dựa trên kết quả tóm tắt, mô-đun đưa ra các định hướng "tích cực" khi đa số các đánh giá và các khía cạnh quan trọng nhận được phản hồi tích cực, định hướng "tiêu cực" khi có nhiều đánh giá tiêu cực hoặc các khía cạnh quan trọng bị đánh giá kém và định hướng "trung tính" khi các đánh giá tích cực và tiêu cực cân bằng nhau.

2.5. Thuật toán và đồ thị tính toán mô hình rừng phân loại quan điểm

2.5.1. Thuật toán và độ phức tạp

Thuật toán xây dựng và dự đoán rừng phân loại quan điểm trình bày như sau:

Thuật toán 2.1. BuildClassificationForest

Đầu vào:

$O = \{(x_i, y_i)\}i=1…N$: tập dữ liệu huấn luyện,

với:

$x_i = (e_i, a_i, h_i, t_i)$: vector đặc trưng (biến dự báo).

$y_i = s_i$: nhãn tình cảm (biến phụ thuộc).

n: số cây trong rừng.

max_depth : độ sâu tối đa của cây.

M: số đặc trưng chọn ngẫu nhiên tại mỗi nút.

min_split: số mẫu tối thiểu để tiếp tục phân tách.

Đầu ra:

$F_{CF} = \{T_1, T_2, …, T_n\}$: rừng phân loại quan điểm

1: FCF ← ∅

2: for i = 1 to n do
3: $O_i^* \leftarrow$ Bootstrap(O, N) // *lấy mẫu có hoàn lại*
4: $T_i \leftarrow$ BuildTree(O_i^*, max_depth, M, min_split, depth=0)
5: $F_{CF} \leftarrow F_{CF} \cup \{T_i\}$
6: end for
7: return F_{CF}

Hàm phụ BuildTree(O, max_depth, M, min_split, depth)

// Bước 1 - Kiểm tra điều kiện dừng đệ quy

1: if depth = max_depth or |O| < min_split or pure(O) then
2: return Leaf(majority_class(O)) // trả về nút lá với nhãn đa số
3: end if

// Bước 2 - Chọn ngẫu nhiên tập đặc trưng ứng viên

4: Features $\leftarrow$ RandomSubset(χ, M) // χ = {e, a, h, t}, $M \leq 4$

// Bước 3 - Tìm phân tách tối ưu theo Information Gain

5: best_gain $\leftarrow -\infty$ // khởi tạo giá trị độ lợi thông tin lớn nhất.
6: for each f $\in$ Features do
7: for each $\theta \in$ thresholds(O, f) do
8: $O_L \leftarrow \{ o \in O : o[f] \leq \theta \}$
9: $O_R \leftarrow \{ o \in O : o[f] > \theta \}$
10: gain $\leftarrow$ ComputeIG(O, O_L, O_R) // theo công thức (2.3) và (2.4)
11: if gain > best_gain then
12: best_gain $\leftarrow$ gain
13: $f^* \leftarrow$ f // đặc trưng phân tách tốt nhất
14: $\theta^* \leftarrow \theta$ // ngưỡng phân tách tốt nhất
15: end if
16: end for
17: end for

// Bước 4 — Phân tách dữ liệu và gọi đệ quy

18: $O_L \leftarrow \{ o \in O : o[f^*] \leq \theta^* \}$
19: $O_R \leftarrow \{ o \in O : o[f^*] > \theta^* \}$
20: $T_L \leftarrow$ BuildTree(O_L, max_depth, M, min_split, depth+1)
21: $T_R \leftarrow$ BuildTree(O_R, max_depth, M, min_split, depth+1)
22: return Node(f^*, θ^*, T_L, T_R) //nút trong với tối ưu (f^*, θ^*)

Độ phức tạp tính toán. Giả sử tập dữ liệu *O* gồm *N mẫu d = 4 chiều (e, a, h, t)*, rừng gồm *n* cây, mỗi cây có chiều sâu tối đa *h* và trung bình *L* nút lá.

Độ phức tạp huấn luyện. Mỗi cây được xây dựng trên tập Bootstrap N' ≈ 0,63N mẫu, tại mỗi nút chọn ngẫu nhiên m_{try} đặc trưng và tìm ngưỡng tách tốt nhất, dẫn đến độ phức tạp trên dữ liệu cân bằng được biểu diễn bởi công thức (2.12).

$$\Theta(\mathrm{n} \times \mathrm{N}' \times m_{try} \times \mathrm{h}) = \mathrm{O}(\mathrm{n} \times \mathrm{N} \times m_{try} \times \log \mathrm{N}) \quad (2.12)$$

Trong đó:

- *n:* là số lượng cây quyết định trong rừng;
- *N*: là số lượng mẫu huấn luyện;
- m_{try} : số đặc trưng được chọn ngẫu nhiên tại mỗi nút phân tách.
- logN: độ sâu trung bình của mỗi cây quyết định trong rừng phân loại quan điểm

Thuật toán 2.2. PredictClassificationForest

Đầu vào:

$F_{CF} = \{T_1, \ldots, T_n\}$: rừng đã huấn luyện.

x = (e, a, h, t): mẫu quan điểm cần phân loại.

Đầu ra:

$\hat{y} \in C = \{positive, negative\}$: nhãn phân loại.

votes ← [ ]
for i = 1 to n do
    votes ← votes ∪ {Traverse(T_i, x)}
end for
ŷ ← MajorityVote(votes)
return ŷ

Độ phức tạp dự đoán. Mỗi mẫu mới được phân loại qua *n* cây, mỗi cây duyệt theo chiều sâu *O(log N)*, tổng cộng độ phức tạp của thuật toán đối với mỗi mẫu theo công thức (2.13).

$$\Theta(n \times log\ N) \quad (2.13)$$

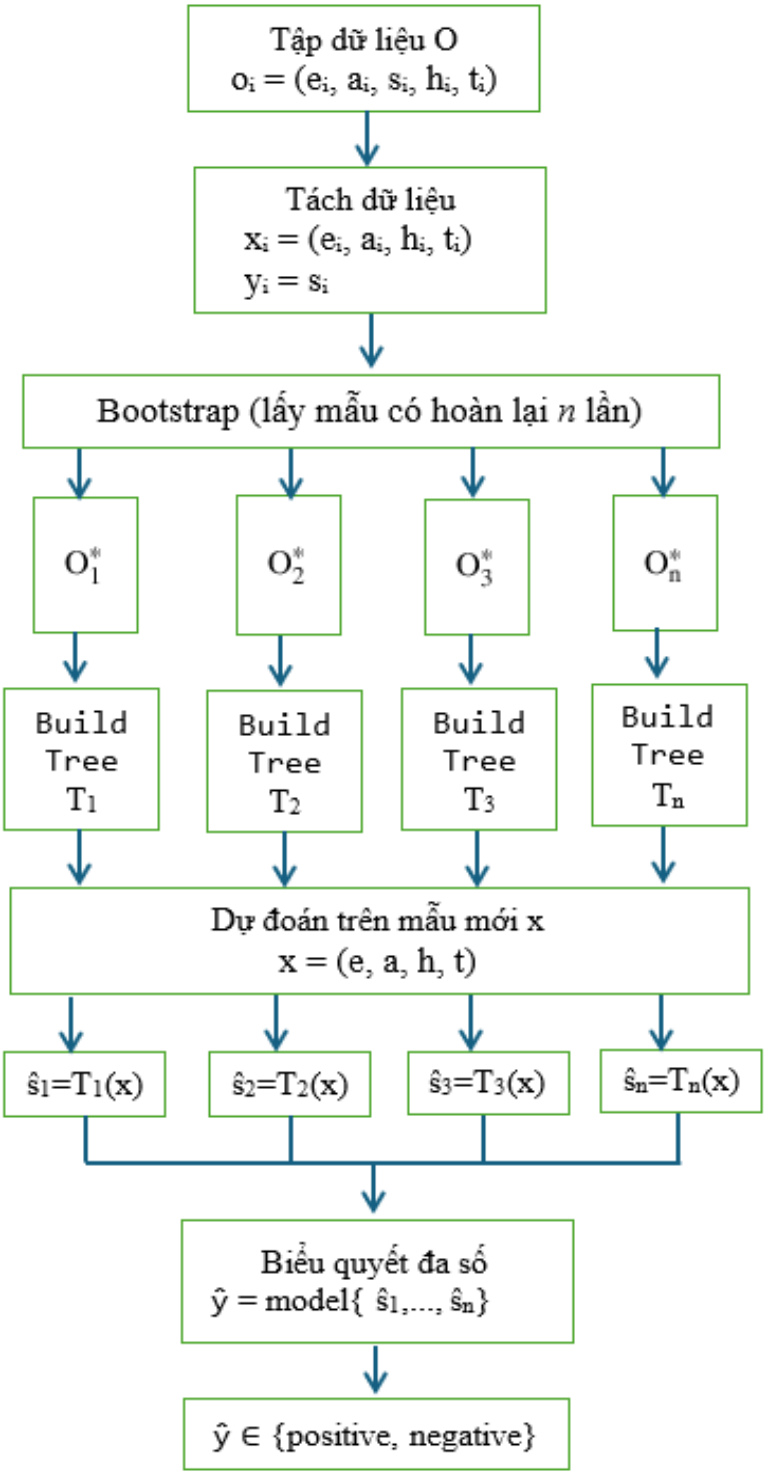
Trong đó:

- *n:* là số lượng cây quyết định trong rừng;
- *N*: là số lượng mẫu dữ liệu huấn luyện được sử dụng để xây dựng cây.

So sánh: độ phức tạp này tương đương Random Forest [65] và nhỏ hơn Gradient Boosting [84] do các cây được xây song song độc lập thay vì tuần tự.

2.5.2. Đồ thị tính toán mô hình rừng phân loại quan điểm

Đồ thị tính toán để biểu diễn trực quan luồng dữ liệu và các tính toán trong mô hình rừng phân loại quan điểm được sơ đồ hóa tại Hình 2.4:



Hình 2.4: Đồ thị tính toán của mô hình rừng phân loại quan điểm

Đồ thị tính toán cho thấy luồng dữ liệu được tổ chức theo ba giai đoạn chính: (i) giai đoạn lấy mẫu Bootstrap, trong đó n tập con O_1^*, O_2^*, ..., O_n^* được tạo ra từ tập gốc O bằng phương pháp lấy mẫu có hoàn lại; (ii) giai đoạn xây dựng cây, trong đó mỗi tập con được sử dụng để cây phân loại T_i tương ứng được xây dựng độc lập; (iii) giai đoạn tổng hợp, trong đó dự đoán của n cây quyết định được kết hợp bằng biểu quyết đa số để nhãn cuối cùng $\hat{y}$ được xác định.

2.6. Diễn giải mô hình rừng phân loại quan điểm

2.6.1. Diễn giải cây phân loại quan điểm

2.6.1.1 Quy tắc quyết định (IF-THEN)

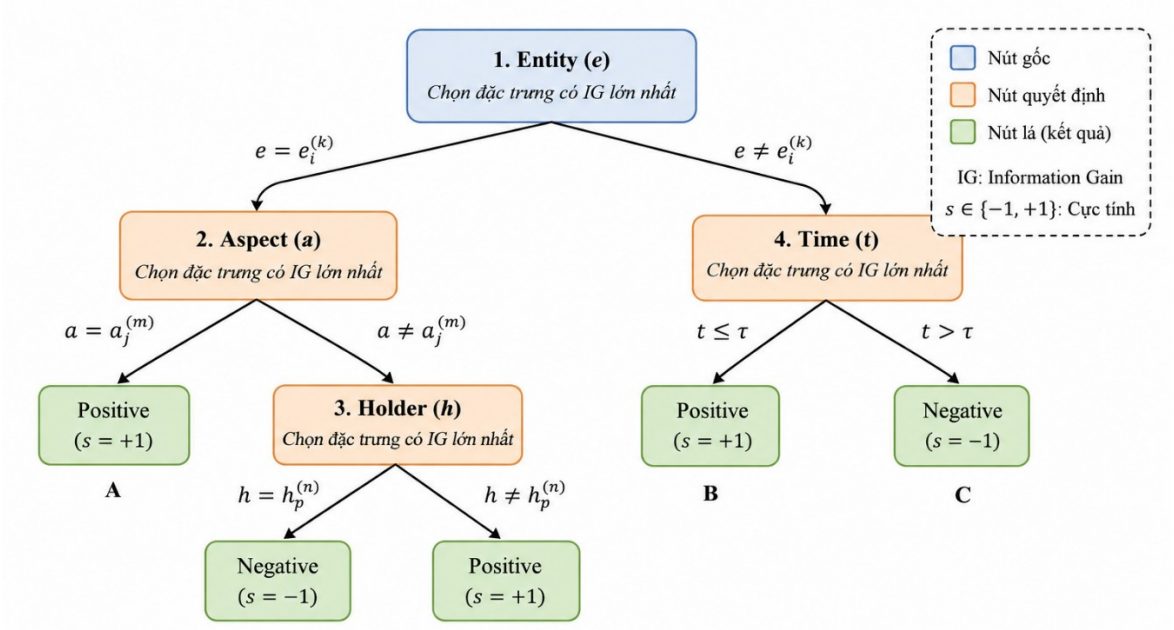
Một quy tắc quyết định là một chuỗi các điều kiện kết hợp bởi toán tử logic AND (∩), dẫn đến một kết luận phân loại, được biểu diễn trong công thức (2.14)

$$\text{IF } (f_1 = v_1) \cap (f_2 = v_2) \cap \ldots \cap (f_k = v_k) \text{ THEN class} = c \qquad (2.14)$$

Trong đó:

- f_i: đặc trưng thứ i;
- v_i: giá trị hoặc ngưỡng của đặc trưng f_i;
- c: lớp phân loại đầu ra với $c \in C$;
- C: tập lớp tình cảm, $C = \{positive, negative\}$.

Ví dụ: Cây quyết định quan điểm như sau:



Hình 2.5: Ví dụ minh họa cây quyết định cho phân loại quan điểm

Cây được xây dựng theo nguyên tắc chọn đặc trưng cho độ lợi thông tin (Information Gain) lớn nhất tại mỗi nút. Đặc trưng sử dụng {e (Entity), a (Aspect), h (Holder), t (Time)}. Kết quả tại lá: Positive (s = +1) hoặc Negative (s = -1).

Dựa trên các đường đi từ nút gốc đến nút lá trong Hình 2.5, có thể rút ra các quy tắc quyết định như sau:

- Quy tắc 1 (Nhánh A):

$$\text{IF (e} = e_i^{(k)}) \wedge (\text{a} = a_j^{(m)}) \text{ THEN } s = +1$$

Nếu quan điểm thuộc thực thể $e_i^{(k)}$ và khía cạnh $a_j^{(m)}$ thì mô hình dự đoán tình cảm tích cực (Positive).

- Quy tắc 2:

$$\text{IF (e} = e_i^{(k)}) \wedge (\text{a} \neq a_j^{(m)}) \wedge (\text{h} = h_p^{(n)}) \text{ THEN } s = -1$$

Nếu quan điểm thuộc thực thể $e_i^{(k)}$ và khía cạnh khác $a_j^{(m)}$ và người đánh giá $h_p^{(n)}$ thì mô hình dự đoán tình cảm tiêu cực (Negative).

- Quy tắc 3:

$$\text{IF (e} = e_i^{(k)}) \wedge (\text{a} \neq a_j^{(m)}) \wedge (\text{h} \neq h_p^{(n)}) \text{ THEN } s = +1$$

Nếu quan điểm thuộc thực thể $e_i^{(k)}$ và khía cạnh khác $a_j^{(m)}$ và người đánh giá khác $h_p^{(n)}$ thì mô hình dự đoán tình cảm tích cực (Positive).

- Quy tắc 4 (Nhánh B):

$$\text{IF (e} \neq e_i^{(k)}) \wedge (\text{t} \leq \tau) \text{ THEN } s = +1$$

Nếu quan điểm không thuộc thực thể $e_i^{(k)}$ và thời gian đánh giá nhỏ hơn hoặc bằng ngưỡng τ thì mô hình dự đoán tình cảm tích cực (Positive).

- Quy tắc 5 (Nhánh C):

$$\text{IF (e} \neq e_i^{(k)}) \wedge (\text{t} > \tau) \text{ THEN } s = -1$$

Nếu quan điểm không thuộc thực thể $e_i^{(k)}$ và thời gian đánh giá lớn hơn ngưỡng τ thì mô hình dự đoán tình cảm tiêu cực (Negative).

2.6.1.2 Độ sâu và tầm quan trọng của đặc trưng

Độ sâu của nút: Độ sâu của một nút v trong cây T được định nghĩa trong công thức (2.15):

$$Depth_T(v) = \begin{cases} 0, & v \text{ là nút gốc}, \\ \text{Depth}\big(\text{parent}(v)\big) + 1, & v \text{ không phải nút gốc}. \end{cases} \quad (2.15)$$

Độ sâu tối thiểu của đặc trưng: Độ sâu tối thiểu của đặc trưng f trong cây T là độ sâu nhỏ nhất của các nút chứa đặc trưng đó trong công thức (2.16):

$$MinDepth_T(f) = \min\{\text{Depth}(v) \mid v \in T, v.\text{feature} = f\} \quad (2.16)$$

Nếu đặc trưng f không xuất hiện trong cây: $MinDepth_T(f) = \infty$

- Nếu đặc trưng có Depth nhỏ, có nghĩa là gần gốc thì quan trọng, ảnh hưởng đến nhiều dự đoán.

- Nếu đặc trưng có Depth lớn, có nghĩa là gần lá thì ít quan trọng hơn, chỉ tinh chỉnh phân loại.

2.6.1.3 Độ đo mức độ hỗn loạn tại nút

Tại mỗi nút v, Entropy được tính theo công thức (2.3) với tập O thay bằng tập O_v là tập mẫu tại nút v. Xác suất thực nghiệm $p_v(c)$ được xác định theo công thức (2.17).

$$p_v(c) = \frac{|\{o_i \in O_v \mid s_i = c\}|}{|O_v|} \quad (2.17)$$

Trong đó:

là xác suất thực nghiệm của lớp c tại nút v với:

- O_v là tập mẫu dữ liệu tại nút v,
- s_i là nhãn tình cảm của mẫu o_i,
- C là tập lớp phân loại.
- Nếu $H(v) = 0$, nút v hoàn toàn tinh khiết, nghĩa là tất cả các mẫu thuộc cùng một lớp. Ngược lại, giá trị $H(v)$ càng lớn cho thấy dữ liệu tại nút càng hỗn loạn và chứa nhiều lớp khác nhau.

- Khi phân tách nút v thành hai nút con v_L nút bên trái và v_R nút bên phải thì độ lợi thông tin được xác định theo công thức (2.18):

$$IG(v) = H(v) - \left(\frac{|O_{v_L}|}{|O_v|} H(v_L) + \frac{|O_{v_R}|}{|O_v|} H(v_R)\right) \quad (2.18)$$

Giá trị $IG(v)$ càng lớn cho thấy phép phân tách càng hiệu quả, do làm giảm đáng kể mức độ hỗn loạn và tăng độ tinh khiết của các nút con.

2.6.2. Diễn giải rừng phân loại ngẫu nhiên

2.6.2.1. Độ sâu tối thiểu trung bình

Độ sâu tối thiểu trung bình [74] của đặc trưng f trong rừng phân loại

$$F = \{T_1, T_2, \dots, T_n\}$$

Được xác định là giá trị trung bình của độ sâu tối thiểu của đặc trưng f trên tất cả các cây quyết định ở công thức (2.19).

$$\overline{Depth}(f) = \frac{1}{J}\sum_{i=1}^{n} Depth_{T_i}(f) \tag{2.19}$$

Trong đó $\text{Depth}_{T_i}(f)$ là độ sâu tối thiểu của f trong cây thứ i

Trường hợp đặc trưng f không xuất hiện trong cây T_i, có hai cách xử lý:

Cách 1: Loại bỏ khỏi tính trung bình:

$$\overline{Depth}(f) = \frac{i:f\in T_j \text{Depth}_\text{i}(f)}{|\{i \mid f \in T_i\}|}$$

Cách 2: Gán giá trị mặc định:

$$\text{Depth}_{T_i}(f) = \max_{\text{depth}}(T_i) + 1 \text{ nếu } f \notin T_i$$

Nếu $\overline{Depth}(f) < \overline{Depth}_{\max}$ thì đặc trưng được coi là quan trọng, trong đó:

$$\overline{Depth}_{\max} = \frac{1}{F}\sum_{f'\in\chi} \overline{Depth}(f')$$

là độ sâu tối thiểu trung bình trên tất cả đặc trưng.

- Nếu $Depth$ nhỏ thì đặc trưng xuất hiện gần nút gốc và quan trọng.
- Nếu $\overline{Depth}$ lớn thì đặc trưng xuất hiện gần nút lá và ít quan trọng.

2.6.2.2. Tầm quan trọng của đặc trưng dựa trên Information Gain

Tầm quan trọng của đặc trưng f trong cây quyết định T_i được xác định bằng tổng độ lợi thông tin tại tất cả các nút sử dụng đặc trưng theo công thức (2.20).

$$I_{T_i}(f) = \sum_{v\in T_i: v.\text{feature}=f} IG(v) \tag{2.20}$$

Trong đó $IG(v)$ là độ lợi thông tin tại nút v được xác định theo công thức (2.18).

Tầm quan trọng trung bình của đặc trưng f trên toàn bộ rừng phân loại F_{CF} được tính theo công thức (2.21).

$$\bar{I}(f) = \frac{1}{n}\sum_{i=1}^{n} I_{T_i}(f) \tag{2.21}$$

Để so sánh tương đối giữa các đặc trưng, giá trị tầm quan trọng được chuẩn hóa sao cho tổng bằng 1 theo công thức (2.22).

$$\bar{I}_{\text{norm}}(f) = \frac{\bar{I}(f)}{\sum_{f' \in \chi} \bar{I}(f')} \tag{2.22}$$

- Nếu $\bar{I}(f)$ có giá trị lớn, đặc trưng f đóng góp đáng kể vào việc giảm mức độ hỗn loạn của dữ liệu và có vai trò quan trọng trong quá trình phân loại.

Ví dụ: Nếu $\bar{I}_{\text{norm}}(f) = 0{,}4$ cho thấy đặc trưng f đóng góp 40% vào tổng độ lợi thông tin của toàn bộ rừng.

2.6.2.3. Tầm quan trọng của đặc trưng dựa trên hoán vị

Nếu đặc trưng f thực sự quan trọng, việc hoán vị ngẫu nhiên các giá trị của f trên tập Out-of-Bag sẽ làm giảm đáng kể độ chính xác của mô hình. Với rừng phân loại $F = \{T_1, \ldots, T_n\}$, tầm quan trọng hoán vị của đặc trưng f được xác định theo công thức (2.23).

$$I^{\text{perm}}(f) = \frac{1}{n}\sum_{i=1}^{n} \text{Acc}_{T_i}^{\text{OOB}} - \text{Acc}_{T_i}^{\text{OOB, perm}(f)} \tag{2.23}$$

Trong đó:

- OOB là tập mẫu Out-of-Bag của cây T_i xác định theo công thức (2.7),
- $\text{Acc}_{T_i}^{\text{OOB}}$: Độ chính xác của cây T_i trên tập OOB.
- $\text{Acc}_{i}^{\text{OOB, perm}(f)}$: Độ chính xác của cây T_i sau khi hoán vị ngẫu nhiên các giá trị của đặc trưng f trên tập OOB.

Nếu $I^{\text{perm}}(f)$ có giá trị lớn, đặc trưng f ảnh hưởng đến khả năng dự đoán của mô hình. Ngược lại, nếu $I^{\text{perm}}(f) \approx 0$, đặc trưng f đóng góp rất ít vào hiệu năng phân loại và có thể được loại bỏ mà không ảnh hưởng đáng kể đến kết quả.

2.6.2.4. Tương tác giữa các đặc trưng

Hai đặc trưng $f_1$và f_2 được xem là có tương tác nếu ảnh hưởng kết hợp của chúng khác với tổng ảnh hưởng riêng lẻ. Trong cây quyết định, tương tác giữa các đặc trưng được thể hiện qua sự xuất hiện cùng nhau trên một đường đi quyết định và thứ tự xuất hiện từ nút gốc đến nút lá.

Độ sâu tối thiểu có điều kiện của đặc trưng f_2 khi f_1 xuất hiện tại nút gốc được xác định theo công thức (2.24)–(2.26).

$$\text{CDepth}(f_2 \mid f_1) = \mathbb{E}[Depth_{T_i}(f_2) \mid \text{root}(T_i).\text{feature} = f_1] \quad (2.24)$$

Cụ thể theo công thức (2.25)

$$\text{CDepth}(f_2 \mid f_1) = \frac{\sum_{i:\text{root}(T_i)=f_1} Depth_{T_i}(f_2)}{\mid \{i:\text{root}(T_i).\text{feature} = f_1\} \mid} \quad (2.25)$$

Trong đó:

• $Depth_{T_i}(f_2)$ là độ sâu tối thiểu của đặc trưng f_2 trong cây T_i. Độ đo tương tác giữa hai đặc trưng được xác định theo công thức (2.26).

$$\text{INT}(f_1, f_2) = \overline{Depth}(f_2) - \text{CDepth}(f_2 \mid f_1) \quad (2.26)$$

Kết quả được diễn giải như sau:

- Nếu $\text{INT}(f_1, f_2) > 0$: khi f_1 xuất hiện tại nút gốc, f_2 có xu hướng xuất hiện sớm hơn so với bình thường, cho thấy tương tác dương, hai đặc trưng bổ trợ lẫn nhau trong quá trình phân tách.

- Nếu $\text{INT}(f_1, f_2) < 0$: khi f_2 xuất hiện muộn hơn khi f_1 đã được chọn tại nút gốc, cho thấy tương tác âm hoặc cạnh tranh giữa hai đặc trưng.

- Nếu $\text{INT}(f_1, f_2) \approx 0$: hai đặc trưng f_1 và f_2 gần như độc lập với nhau trong cấu trúc rừng.

2.6.2.5. Biểu đồ phụ thuộc từng phần

Biểu đồ phụ thuộc từng phần (Partial Dependence Plot- PDP) mô tả ảnh hưởng biên của một hoặc nhiều đặc trưng đối với dự đoán của mô hình, bằng cách lấy trung bình dự đoán trên phân phối thực nghiệm của các đặc trưng còn lại. PDP được sử dụng để phân tích xu hướng tăng giảm, quan sát quan hệ tuyến tính hoặc phi tuyến và phát hiện tương tác giữa các đặc trưng.

Với đặc trưng f nhận giá trị v, hàm phụ thuộc từng phần được xác định theo công thức (2.27)

$$\text{PD}_f(v) = \frac{1}{N}\sum_{i=1}^{N} y(x_i^{f \leftarrow v}) \quad (2.27)$$

Trong đó:

- $x_i^{f \leftarrow v}$: là mẫu dữ liệu x_i sau khi thay thế giá trị của đặc trưng f bằng giá trị v;
- $y(.)$: là giá trị dự đoán của mô hình rừng phân loại;
- N: là tổng số mẫu trong tập dữ liệu.

Trường hợp xét đồng thời hai đặc trưng f_1 *và* f_2 với các giá trị tương ứng là v_1 và v_2, hàm PDP hai chiều được xác định theo công thức (2.28):

$$\mathrm{PD}_{f_1,f_2}(v_1, v_2) = \frac{1}{N}\sum_{i=1}^{N} y(x_i^{f_1 \leftarrow v_1, f_2 \leftarrow v_2}) \quad (2.28)$$

Biểu đồ phụ thuộc từng phần (PDP) hai chiều cho phép phát hiện tương tác phi tuyến giữa hai đặc trưng bộ năm mà phân tích đơn chiều không thể nắm bắt được.

2.7. Thực nghiệm

Nội dung trong chương trình bày về tập dữ liệu đánh giá thực phẩm, tập dữ liệu đánh giá khách sạn và thực nghiệm trên các tập dữ liệu này, mô tả kịch bản và kết quả thực nghiệm mô hình. Kết quả thực nghiệm được thể hiện trong các kịch bản đã nêu và cho kết quả tương ứng với từng trường hợp.

2.7.1. Tập dữ liệu sử dụng

2.7.1.1. Tập dữ liệu Amazon Food Reviews

Tập dữ liệu Amazon Food Reviews là tập con thuộc cơ sở dữ liệu đánh giá sản phẩm được công bố [67][183] gồm hơn 500.000 đánh giá thực phẩm. Mỗi đánh giá chứa thông tin người dùng, xếp hạng sao và nội dung văn bản đánh giá, là cơ sở dữ liệu chuẩn được cộng đồng NLP sử dụng rộng rãi cho phân loại tình cảm và truy xuất thông tin. Tập dữ liệu này thuộc nhóm thương mại điện tử, miền phổ biến nhất cho nghiên cứu ABSA với các nhãn đánh giá sản phẩm theo khía cạnh cụ thể, phù hợp với bài toán trích xuất bộ năm quan điểm của luận án [191].

Các chi tiết của bộ dữ liệu trong Bảng 2.1 mô tả các đánh giá được viết bởi hơn 200.000 người dùng cho 74.258 sản phẩm, trong đó có 260 người dùng đã cung cấp hơn 50 đánh giá. Độ dài trung bình của mỗi đánh giá là khoảng 56 từ.

Bảng 2.1: Chi tiết dữ liệu Amazon

Thống kê dữ liệu	Số lượng mẫu
Đánh giá	568.454

Người dùng	256.059
Sản phẩm	74.258
Người dùng với >50 reviews	260
Trung bình số từ trên mỗi đánh giá	56

Bảng 2.2 có 18.532 đánh giá (14.825 đánh giá huấn luyện, 3.707 đánh giá thử nghiệm) là tập dữ liệu dùng cho thực nghiệm khảo sát ban đầu về biểu diễn đặc trưng văn bản bằng Doc2vec (Bài báo 1), chưa sử dụng cấu trúc bộ năm. Thực nghiệm mô hình rừng phân loại quan điểm (F_{CF}) sử dụng một tập riêng gồm 8.020 mẫu bộ năm quan điểm được trích xuất từ 5.411 đánh giá lấy mẫu ngẫu nhiên (Bài báo 2), chia thành 6.416 mẫu huấn luyện và 1.604 mẫu kiểm tra như trình bày tại Bảng 2.5 (§2.7.2.1).

Bảng 2.2: Dữ liệu huấn luyện và thử nghiệm

Thống kê dữ liệu	**Số lượng mẫu**
Đánh giá	18.532
Huấn luyện	14.825
Thử nghiệm	3.707

2.7.1.2. Tập dữ liệu TripAdvisor Hotel Reviews

Tập dữ liệu TripAdvisor Hotel Reviews [2] gồm 235.793 đánh giá khách sạn từ TripAdvisor. Mỗi đánh giá bao gồm văn bản đánh giá, định danh khách sạn, xếp hạng tổng thể và bảy xếp hạng khía cạnh tùy chọn: Value, Rooms, Location, Cleanliness, Check in/front desk, Service và Business service, thang xếp hạng từ 1★ đến 5★.

Theo khảo sát hệ thống của [192] phân tích tình cảm cho đánh giá khách sạn bao gồm các giai đoạn: tiền xử lý, biểu diễn đặc trưng, phân loại tình cảm và giải thích mô hình. Bộ năm quan điểm (e, a, s, h, t) được Bing Liu [1] đề xuất như khung biểu diễn chuẩn cho khai phá quan điểm, trong đó tập dữ liệu TripAdvisor là tập dữ liệu đánh giá khách sạn được cộng đồng nghiên cứu sử dụng nhiều nhất. Trong luận án, thực nghiệm được thực hiện trên tập con gồm 1.850 khách sạn với 108.891 đánh giá có đầy đủ thông tin cần thiết cho việc tạo bộ năm quan điểm, như trình bày tại Bảng 2.3.

Bảng 2.3: Thống kê tập dữ liệu

Thống kê dữ liệu	Số lượng
Số lượng khách sạn	1.850
Số lượng đánh giá	108.891

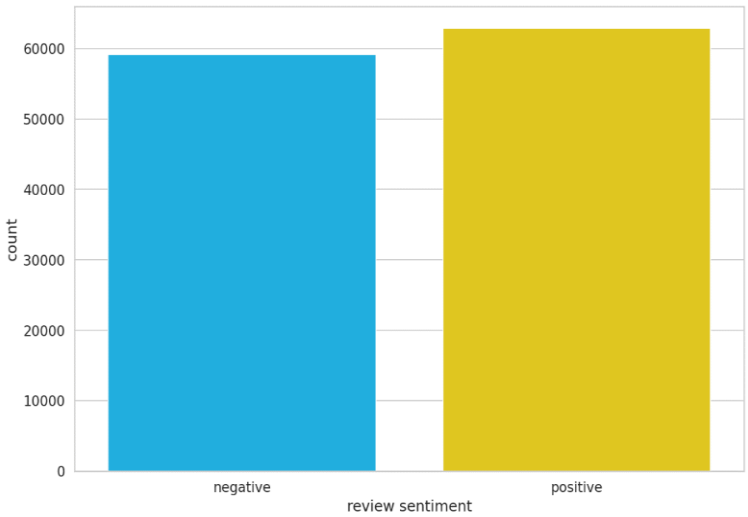
2.7.2. Tiền xử lý dữ liệu

2.7.2.1. Tiền xử lý dữ liệu Amazon Food Reviews

Trong tập dữ liệu Amazon Food Reviews có các đánh giá chứa các chữ số, ký tự đặc biệt,... Một số từ có thể xuất hiện thường xuyên trong câu nhưng chúng không đóng góp nghĩa của câu (ví dụ: giới từ như on, the, at,...). Thực hiện tách các từ trong đánh giá thành token và xóa những từ không cần thiết bằng cách sử dụng công cụ NLTK. Các từ còn lại có thể mang ý nghĩa của các câu bao gồm danh từ, cụm danh từ, động từ, tính từ và trạng từ. Sau đó loại bỏ các giá trị trống trong tập dữ liệu. Dữ liệu còn lại là 568.411 đánh giá. Tỷ lệ phần trăm theo xếp hạng của các đánh giá trong Bảng 2.4. Sau đó, xử lý phân bố dữ liệu cân bằng bằng cách chia tập dữ liệu thành xếp hạng tiêu cực với hạng 1★, 2★ là âm và xếp hạng tích cực với hạng 4★, 5★ là dương và loại bỏ hạng 3★. Trong Hình 2.6 đã minh họa một tập con lấy mẫu cân bằng riêng số lượng đánh giá tích cực và tiêu cực sau khi nhị phân hóa, gồm 59.167 đánh giá tiêu cực và 62.856 đánh giá tích cực (tổng cộng 122.023 đánh giá đã được gán nhãn phân cực rõ ràng).

Bảng 2.4: Chi tiết tỷ lệ phần trăm của hạng dựa trên các đánh giá

Hạng	Tổng số đánh giá	Tỷ lệ (%)
5	363.111	63,88
4	80.655	14,19
3	42.638	7,5
2	29.743	5,23
1	52.264	9,19



Hình 2.6: Phân bố số lượng đánh giá có quan điểm là tích cực và tiêu cực

Tập dữ liệu gốc Amazon Food Reviews gồm 568.454 đánh giá, tuy nhiên do phân phối nhãn nghiêng mạnh về phía tích cực nên áp dụng chiến lược lấy mẫu phân tầng ngẫu nhiên (stratified random sampling) với tỷ lệ 1:1 giữa nhãn tích cực và tiêu cực để tạo tập dữ liệu thực nghiệm cân bằng, cụ thể: lấy ngẫu nhiên 5.411 đánh giá (tỷ lệ positive:negative xấp xỉ 50:50) từ bộ dữ liệu 568.454 đánh giá để thực nghiệm trích xuất bộ năm quan điểm. Kết quả thu được tập dữ liệu bộ năm quan điểm có 8.020 mẫu, sau đó chia ngẫu nhiên tập bộ năm quan điểm này thành tập dữ liệu huấn luyện là 6.416 mẫu (80%) và tập dữ liệu kiểm tra là 1.604 mẫu (20%) chi tiết tại Bảng 2.5 để huấn luyện và đánh giá mô hình rừng phân loại quan điểm (F_{CF}).

Bảng 2.5: Dữ liệu huấn luyện và kiểm tra

Tổng số mẫu bộ năm quan điểm	**Số lượng mẫu bộ năm quan điểm để huấn luyện**	**Số lượng mẫu bộ năm quan điểm để kiểm tra**
8.020	6.416	1.604

2.7.2.2. Tiền xử lý dữ liệu TripAdvisor Hotel Reviews

Từ tập dữ liệu 108.891 đánh giá được chuyển thành 136.528 mẫu bộ năm quan điểm. Sau đó loại bỏ các mẫu không có giá trị phân cực còn lại 45.706 mẫu để thực nghiệm. Tập dữ liệu 45.706 mẫu bộ năm quan điểm có phân phối hạng tình cảm nghiêng về phía tích cực dẫn đến tỷ lệ mất cân bằng nhãn khoảng 4:1 sau khi nhị phân hóa positive/negative. Sau đó, giá trị của các đặc trưng được mã hóa bằng cách đánh số thứ tự như trong Bảng 2.6.

Bảng 2.6: Thống kê các giá trị của đặc trưng

Các đặc trưng	Số các giá trị khác nhau	Đánh số thứ tự
Entity	14	1 đến 14
Aspect	7	1 đến 7
Holder	8.642	1 đến 8.642
Time	2.520	1 đến 2.520

Ghi chú: Bảng 2.6 thống kê trên tập con dùng để thực nghiệm. Trong tập con này, luận án chỉ lấy 14 khách sạn (thành phần thực thể Entity, là tên khách sạn) trong tổng số 1.850 khách sạn của tập dữ liệu TripAdvisor Hotel Reviews [2] đã mô tả tại mục 2.7.1.2, do đó đặc trưng Entity chỉ có 14 giá trị khác nhau. Đặc trưng Aspect có 7 giá trị tương ứng với bảy khía cạnh được chấm điểm của tập dữ liệu (Value, Rooms, Location, Cleanliness, Check in/front desk, Service, Business service).

2.7.3. Các kịch bản thực nghiệm

2.7.3.1. Khai phá quan điểm với rừng phân loại dựa trên bộ năm quan điểm trên tập dữ liệu đánh giá thực phẩm Amazon Food Reviews

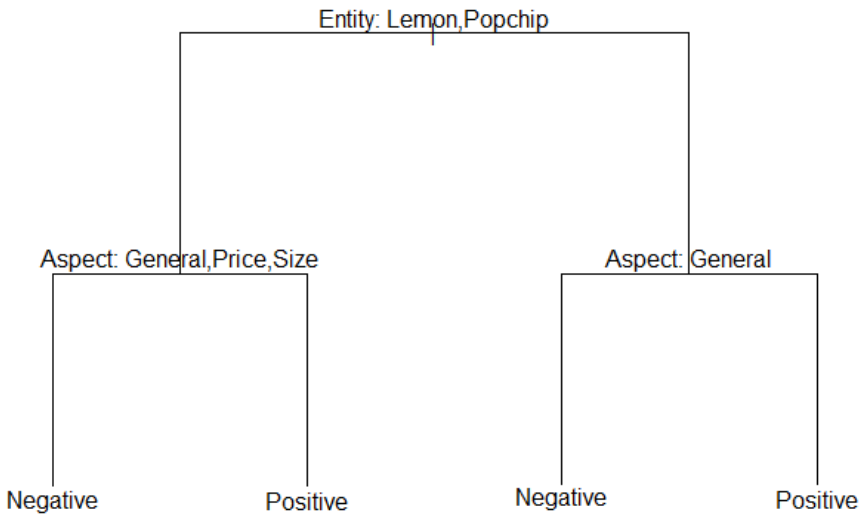
Kịch bản 1. Rừng phân loại quan điểm dựa trên bộ năm quan điểm trên tập dữ liệu đánh giá thực phẩm Amazon Food Reviews

Từ bộ dữ liệu đánh giá thực phẩm, xây dựng mô hình rừng phân loại để khai thác quan điểm về các khía cạnh sản phẩm thực phẩm đã được khách hàng đánh giá. Trước tiên, dữ liệu được chia thành n mẫu bootstrap, tiếp theo xây dựng rừng phân loại với các thiết lập tham số mô hình tại Bảng 2.7 và sau đó tổng hợp phân loại bằng cách lấy giá trị xảy ra thường xuyên nhất cho mỗi phân loại, được minh họa Hình 2.7.

Bảng 2.7: Thiết lập tham số mô hình Rừng phân loại quan điểm trong thực nghiệm

Tham số	Ý nghĩa	Giá trị thử nghiệm	Giá trị được chọn	Cơ sở lựa chọn
n	Số cây	50, 70, 100	100	Giá trị kinh nghiệm cân bằng giữa độ

				chính xác và thời gian huấn luyện
m_{try}	Số đặc trưng ngẫu nhiên tại mỗi nút	2, 3, 4	$\lvert\sqrt{\chi}\rvert = \lvert\sqrt{4}\rvert = 2$	Theo quy tắc chuẩn Random Forest [65]; χ=4 đặc trưng đầu vào
max_depth	Chiều sâu tối đa	-	Không giới hạn	Cây mọc đến khi nút lá thuần nhất hoặc *< min_samples_split*
min_samples_split	Số mẫu tối thiểu để tách	2, 3	2	Giá trị mặc định
criterion	Tiêu chí phân tách	-	Entropy $H(O)$	Sử dụng Entropy Shannon $H(O)$, công thức (2.3)
random_state	Số khởi tạo bộ sinh số ngẫu nhiên	-	42	Đảm bảo tái lập kết quả



Hình 2.7: Phân loại quan điểm theo khía cạnh thực phẩm.

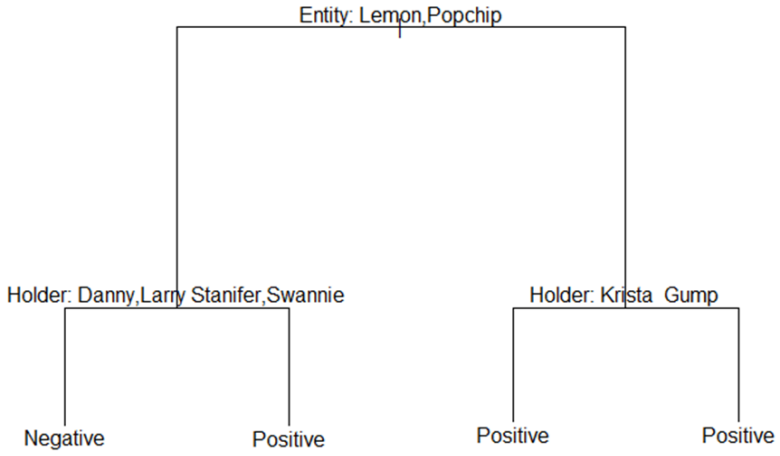
Đánh giá mô hình rừng phân loại trên bộ dữ liệu đánh giá thực phẩm theo khía cạnh trong Bảng 2.8.

Bảng 2.8: Đánh giá mô hình F_{CF} theo khía cạnh thực phẩm

Thuật toán	Độ chính xác	Độ chính xác dương	Độ bao phủ	F1
Rừng phân loại (F_{CF})	0,805	0,80	0,76	0,78

Kịch bản 2. Rừng phân loại quan điểm dựa trên đặc trưng thời gian và người đưa ra quan điểm trên tập dữ liệu đánh giá thực phẩm Amazon Food Reviews

Kịch bản 2 xem xét đến yếu tố đặc trưng về thời gian nêu quan điểm và người đưa ra quan điểm là tích cực hay tiêu cực. Kết quả mô hình rừng phân loại quan điểm theo người đưa ra đánh giá được minh họa tại Hình 2.8 và thời gian đánh giá được minh họa tại Hình 2.9.

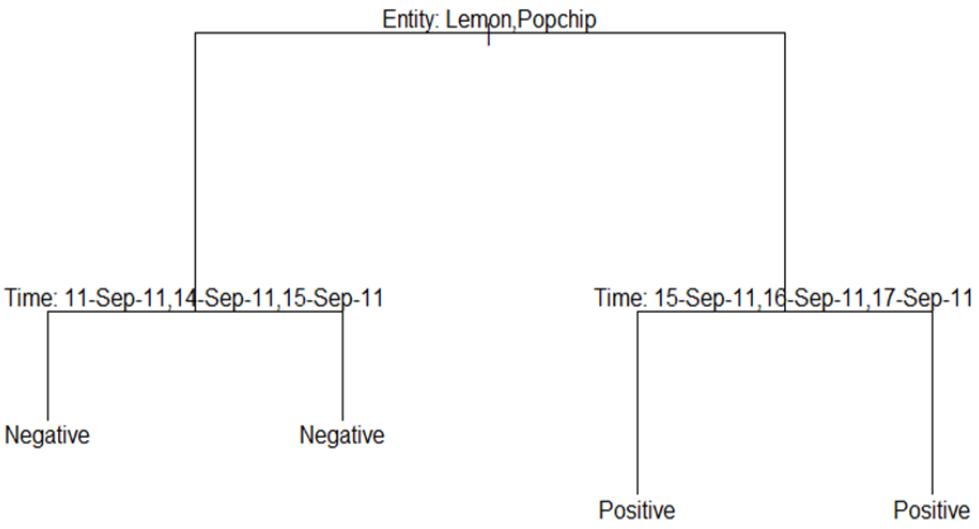


Hình 2.8: Phân loại quan điểm theo người đánh giá

Đánh giá mô hình rừng phân loại trên bộ dữ liệu đánh giá thực phẩm theo thời gian và người đánh giá trong Bảng 2.9.

Bảng 2.9: Đánh giá mô hình F_{CF} theo thời gian và người đánh giá

Thuật toán	Độ chính xác	Độ chính xác dương	Độ bao phủ	F1
Rừng phân loại (FCF)	0,82	0,85	0,67	0,75



Hình 2.9: Phân loại quan điểm theo thời gian đánh giá

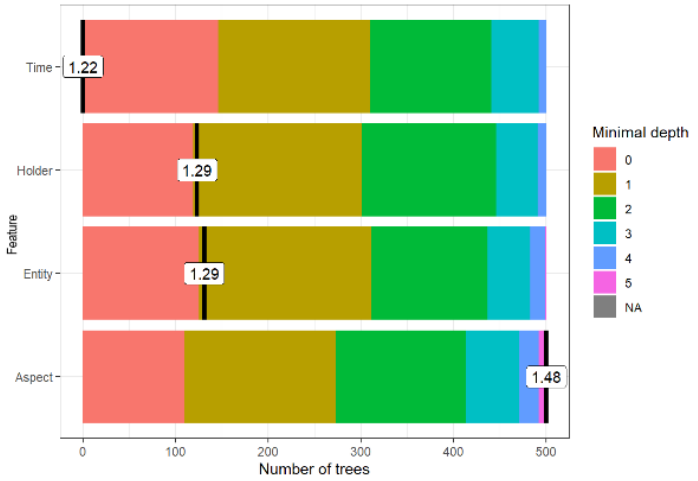
2.7.3.2. Khai phá quan điểm với rừng phân loại dựa trên bộ năm quan điểm trên tập dữ liệu đánh giá khách sạn TripAdvisor Hotel Reviews

Kịch bản 1. Diễn giải rừng phân loại quan điểm dựa trên bộ năm quan điểm trên tập dữ liệu đánh giá khách sạn TripAdvisor Hotel Reviews

Từ tập dữ liệu đánh giá khách sạn, mô hình rừng ngẫu nhiên được xây dựng để khai phá quan điểm về các đặc trưng quan trọng bao gồm thực thể, khía cạnh, người đưa ra quan điểm và thời gian đánh giá về khách sạn. Kết quả của thuật toán được trình bày trong Bảng 2.10.

Bảng 2.10: Đánh giá mô hình rừng ngẫu nhiên trên bộ dữ liệu khách sạn

Độ chính xác	Độ chính xác dương (P)	Độ bao phủ (R)	Trung bình điều hòa F1
0,91	0,96	0,50	0,66



Hình 2.10**:** Phân bố độ sâu tối thiểu của các đặc trưng trong rừng

Sự phân bố độ sâu tối thiểu của các đặc trưng trong rừng có 500 cây (thực nghiệm trên TripAdvisor Hotel Reviews): mức độ sâu tối thiểu từ 0 đến 5 được thể hiện trong Hình 2.10. Độ sâu tối thiểu trung bình của đặc trưng Time có giá trị nhỏ nhất là 1,22, xuất hiện gần gốc trong hầu hết các cây, do đó là đặc trưng quan trọng nhất theo tiêu chí độ sâu tối thiểu (theo tầm quan trọng hoán vị, Entity có tác động lớn nhất, Bảng 2.11). Ngoài ra, đặc trưng Time có trong phần lớn các cây có độ sâu tối thiểu là 0 hoặc 1 vì màu đỏ và màu cam chiếm ưu thế. Trong khi đó, độ sâu tối thiểu trung bình của đặc trưng Holder và Entity phân bố tương tự nhau, có giá trị xấp xỉ nhau (lần lượt 1,286 và 1,290). Còn độ sâu tối thiểu trung bình của đặc trưng Aspect có giá trị cao nhất là 1,48, ít quan trọng nhất. Kết quả cho thấy đặc trưng có độ sâu tối thiểu nhỏ thì xuất hiện sớm, ảnh hưởng đến nhiều dự đoán và quan trọng.

Tầm quan trọng của các đặc trưng trong rừng: Trong Bảng 2.11, độ sâu tối thiểu trung bình của đặc trưng Time có giá trị nhỏ nhất là 1,2220. Đối với tầm quan trọng của độ lợi thông tin (IG) thì đặc trưng Holder có giá trị cao nhất là 92,1901, đóng góp nhiều nhất vào giảm độ hỗn tạp của các đặc trưng và thứ hai là đặc trưng Time có giá trị 89,3295, đặc trưng Aspect có giá trị nhỏ nhất 28,6544 (giá trị của Entity xem ghi chú dưới Bảng 2.11). Trong khi giá trị tầm quan trọng của hoán vị của đặc trưng Entity có giá trị là 0,0122, tác động cao nhất khi bị hoán vị. Đối với số lần làm nút gốc thì đặc trưng Time xuất hiện ở gốc nhiều nhất với giá trị là 146 lần, tiếp theo là Entity có giá trị 125 lần, Holder có giá trị 119 lần và Aspect có giá trị 110 lần. Giá trị của các đặc trưng được liệt kê trong Bảng 2.11.

Sự tương tác của các đặc trưng: Trong Bảng 2.12, giá trị cường độ tương tác dương lớn nhất (và là cặp dương duy nhất) là Time:Holder (0,321). Do đặc trưng Time ở nút gốc và đặc trưng Holder có giá trị độ sâu tối thiểu có điều kiện là 0,9650 so với giá trị độ sâu tối thiểu không điều kiện là 1,2860 và số lần xuất hiện cùng nhau là 475 lần. Điều này có nghĩa là khi Time xuất hiện ở nút gốc, Holder có xu hướng xuất hiện sớm hơn vì độ sâu giảm từ 1,2860 xuống 0,9650, cho thấy hai đặc trưng này có hiệu ứng tương tác mạnh. Điều này phản ánh thực tế là tình cảm phụ thuộc đồng thời vào thời gian đánh giá (Time) và người đánh giá (Holder). Ngược lại, các cặp tương tác có giá trị âm cho thấy quan hệ cạnh tranh. Đối với tương tác Entity:Aspect = -1,7279, tương tác âm cực mạnh, cho thấy khi Entity ở gốc, Aspect

hầu như không xuất hiện (độ sâu tăng từ 1,4820 lên 3,2099). Điều này do Entity đã chứa thông tin về Aspect, dẫn đến dư thừa giữa hai đặc trưng. Đối với tương tác Entity:Time = -0,8981 và Entity:Holder = -0,6712, cũng cho thấy Entity có vai trò cao, Entity được chọn làm phân tách chính, các đặc trưng khác trở nên ít quan trọng hơn. Điều này cho thấy một số thực thể có tình cảm ổn định không phụ thuộc thời gian hay người đánh giá. Đối với tương tác âm trung bình Holder:Time = -0,3969, cho thấy mặc dù Time:Holder có tương tác dương (+0,321), nhưng quan hệ ngược lại là cạnh tranh. Điều này phản ánh sự không đối xứng trong vai trò của các đặc trưng. Đặc trưng Time là yếu tố quan trọng cần kết hợp với Holder, nhưng nếu biết được Holder là người đánh giá có xu hướng rõ ràng thì Time trở nên ít quan trọng hơn.

Bảng 2.11: Thống kê tầm quan trọng của các đặc trưng trong rừng

Đặc trưng	**Độ sâu tối thiểu trung bình**	**Số nút sử dụng đặc trưng *f***	**Độ chính xác giảm (Tầm quan trọng của hoán vị)**	**Độ giảm IG (Tầm quan trọng của IG)**	**Số lần làm nút gốc**
Aspect	1,4820	9366	0,0043	28,6544	110
Entity	1,2900	13458	0,0122	–	125
Holder	1,2860	25390	0,0021	92,1901	119
Time	1,2220	24937	0,0000	89,3295	146

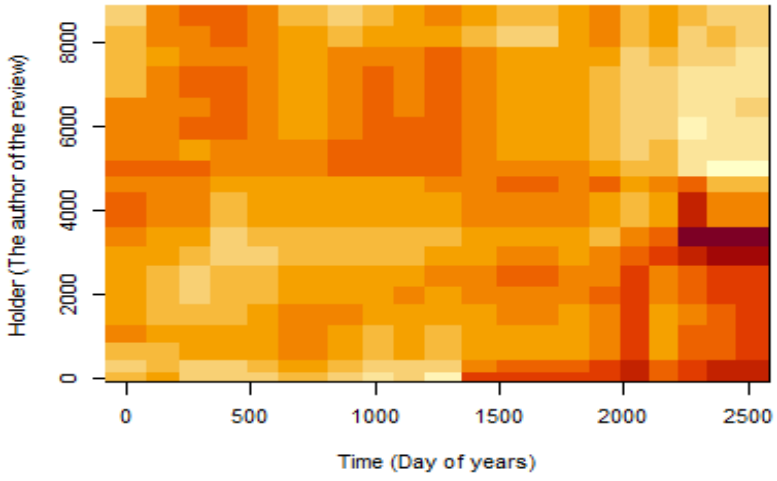
Bảng 2.12: Thống kê sự tương tác của các đặc trưng trong rừng

Đặc trưng được phân tích	**Đặc trưng ở gốc**	**Độ sâu tối thiểu có điều kiện**	**Số lần xuất hiện cùng nhau**	**Tên tương tác**	**Độ sâu tối thiểu không điều kiện**	**Cường độ tương tác**
Holder	Time	0,9650	475	Time:Holder	1,2860	0,321
Entity	Time	1,5192	457	Time:Entity	1,2900	-0,2292
Time	Holder	1,6189	448	Holder:Time	1,2220	-0,3969
Holder	Entity	1,9572	429	Entity:Holder	1,2860	-0,6712
Time	Entity	2,1201	423	Entity:Time	1,2220	-0,8981
Aspect	Entity	3,2099	394	Entity:Aspect	1,4820	-1,7279

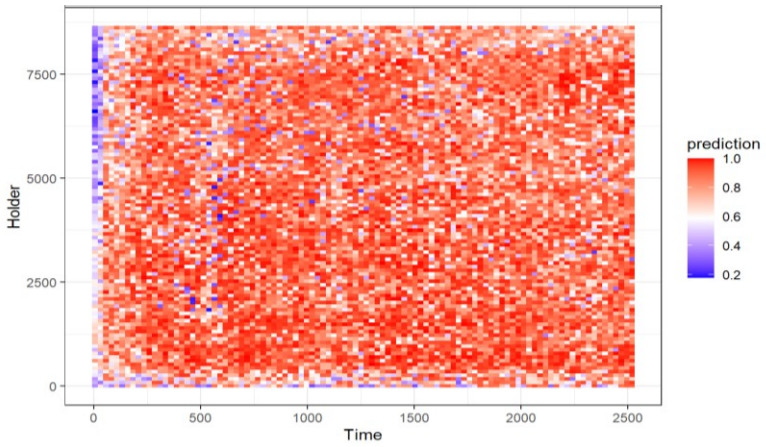
Kịch bản 2. Diễn giải rừng phân loại quan điểm dựa trên thời gian và người đưa ra quan điểm trên tập dữ liệu đánh giá khách sạn TripAdvisor Hotel Reviews

Hình 2.11 hiển thị biểu đồ nhiệt phân vùng không gian 2 chiều {Time, Holder} theo công thức (2.28). Quan sát cho thấy, sự xuất hiện các dải ngang với màu sắc khác nhau, chứng tỏ Holder có ảnh hưởng mạnh về tình cảm thay đổi theo Time. Một số Holder có dải màu đỏ đậm có xu hướng đánh giá tích cực cao, ổn định theo Time, chứng tỏ một số người dùng có xu hướng đánh giá ổn định theo thời gian, phụ thuộc vào đặc điểm cá nhân quyết định. Trong khi các Holder khác có dải màu cam và vàng có xu hướng tăng mạnh theo Time, chứng tỏ một số người dùng khác bị ảnh hưởng mạnh bởi thời gian, có thể do kinh nghiệm tích lũy. Kết quả cho thấy, mô hình rừng phân loại đã tự động phát hiện và khai thác tương tác qua cấu trúc cây nắm bắt được mối quan hệ phi tuyến tính của Time và Holder, hiệu ứng của Time phụ thuộc vào Holder cụ thể và xác nhận kết quả phân tích tương tác của (Time, Holder) = 0,321 được thể hiện trong Bảng 2.12.

Hình 2.12, hiển thị các giá trị dự đoán thực tế của rừng phân loại trên không gian {Time, Holder}. Mỗi điểm đại diện cho một mẫu dữ liệu, màu sắc biểu diễn xác suất dự đoán tích cực. Quan sát cho thấy, đa số các mẫu có giá trị dự đoán từ 0,6 đến 1 nhiều hơn các mẫu có giá trị dự đoán từ 0,2 đến dưới 0,6, điều này cho thấy dự đoán tích cực chiếm đa số, khoảng 80-90% điểm có màu đỏ và cam, đặc trưng Holder có ảnh hưởng rõ rệt vì trên trục Y các đặc trưng Holder khác nhau có màu sắc khác biệt, đặc trưng Time có xu hướng nhẹ vì trên trục X từ ngày đầu đến ngày sau có xu hướng màu đỏ hơn nhưng không mạnh, cho thấy người đánh giá có xu hướng thời gian sau có tình cảm tích cực hơn thời gian trước.



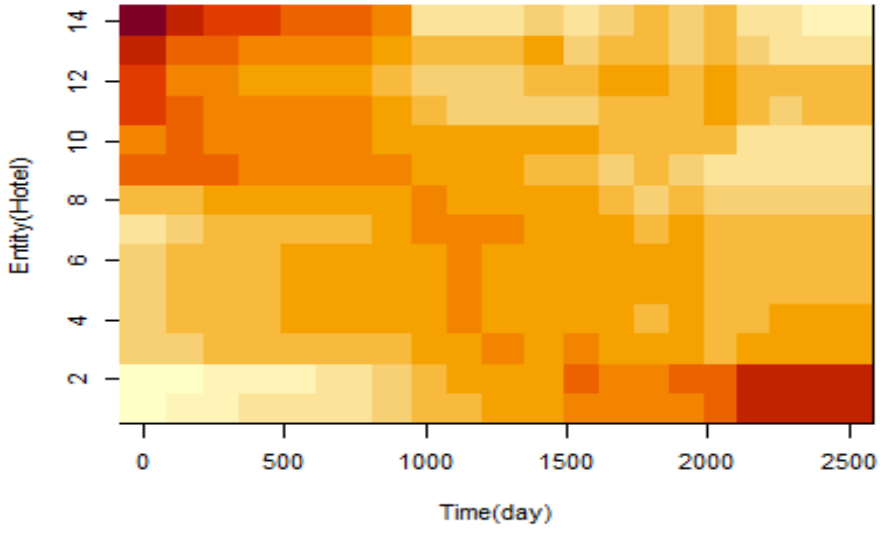
Hình 2.11: Minh họa phân vùng không gian hai chiều{Time, Holder}



Hình 2.12: Hiển thị các giá trị dự đoán của rừng ngẫu nhiên đối với {Time, Holder}

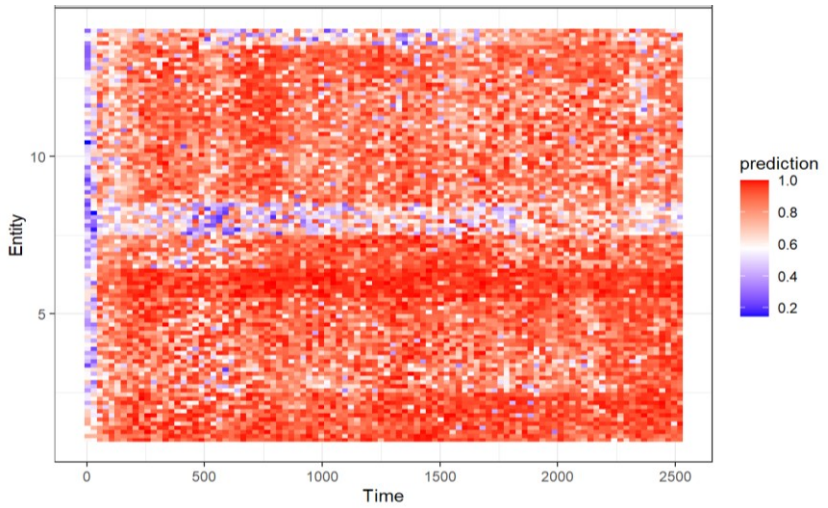
Kịch bản 3. Diễn giải rừng phân loại quan điểm dựa trên thực thể và thời gian trên tập dữ liệu đánh giá khách sạn TripAdvisor Hotel Reviews

Hình 2.13, hiển thị xu hướng trung bình của xác suất dự đoán tích cực trên không gian {Entity, Time}. Qua quan sát cho thấy sự phân hóa rõ rệt, Entity có chỉ số ID từ 2-3 có dải màu đỏ đậm đồng nhất cao bất kể thời gian, cho thấy các thực thể có được lựa chọn mạnh không ảnh hưởng bởi yếu tố thời gian. Entity có chỉ số ID từ 8-11 có màu sắc biến động mạnh theo trục Time, đặc biệt xuất hiện "vùng trũng" màu vàng nhạt tại khoảng Time từ 1000-1500, phản ánh thời kỳ khó khăn có thể do biến động thị trường hoặc vấn đề nội bộ. Một số Entity có chỉ số ID từ 12-14 cho thấy xu hướng cải thiện theo thời gian từ màu vàng sang màu cam đến màu đỏ, cho thấy quá trình nâng cấp chất lượng sản phẩm theo thời gian.



Hình 2.13: Minh họa phân vùng không gian hai chiều{Entity, Time}

Hình 2.14, hiển thị dự đoán thực tế cho từng mẫu dữ liệu, bổ sung thông tin về tính không đồng nhất mà biểu đồ Hình 2.13 không thể hiện. Đặc trưng Entity có chỉ số ID ≈ 9, xuất hiện dải ngang màu xanh tím xuyên suốt trục Time, cho thấy đây là thực thể có phần lớn các mẫu đều tiêu cực. Quan sát tổng thể biểu đồ Hình 2.14 phản ánh tập dữ liệu thiên về lớp tích cực nhiều hơn lớp tiêu cực. Điều này phù hợp với thực tế, người dùng có xu hướng đánh giá tích cực nhiều hơn tiêu cực. Xuất hiện các điểm ngoại lệ màu xanh tím rải rác trong vùng đỏ và ngược lại, cho thấy tình cảm phụ thuộc vào nhiều yếu tố ngoài Entity và Time còn có Aspect, Holder.



Hình 2.14**:** Hiển thị các giá trị dự đoán của rừng ngẫu nhiên đối với {Entity, Time}

Kết quả trực quan hóa cho thấy sự không đồng nhất đã giải thích tại sao tương tác âm mạnh giữa Entity và Time (Entity:Time = -0,8981). Khi Entity được xác định trước ở nút gốc cây, tình cảm của phần lớn trường hợp chiếm ≈ 70% đã được quyết định, Time chỉ cần thiết để tinh chỉnh cho ≈ 30% Entity còn lại. Chứng tỏ rằng Entity quan trọng hơn Time trong việc xác định tình cảm.

Từ kết quả ba kịch bản thực nghiệm cho thấy mô hình rừng phân loại quan điểm F_{CF} đạt Accuracy = 0,805 trên Amazon Food Reviews (kịch bản theo khía cạnh) và Accuracy = 0,82 (kịch bản theo thời gian và người đánh giá) trên 8.020 mẫu bộ năm quan điểm, nhờ cơ chế Bootstrap và lựa chọn đặc trưng ngẫu nhiên giúp giảm phương sai và hạn chế quá khớp (overfitting). Để tham chiếu, thực nghiệm khảo sát ban đầu tại Bài báo 1 (biểu diễn Doc2vec, không dùng bộ năm, 18.532 đánh giá) cho Accuracy của Cây quyết định là 0,752, Random Forest là 0,821 và Gradient Boosting là 0,845.

Tuy nhiên, kết quả trên TripAdvisor Hotel Reviews (Bảng 2.10) cho thấy hiện tượng nghịch lý về độ chính xác, bởi vì Accuracy đạt 0,91 rất cao nhưng Recall chỉ 0,50, dẫn đến F1 = 0,66 thấp hơn đáng kể so với hai kịch bản trên Amazon. Nguyên nhân nằm ở tỷ lệ mất cân bằng nhãn khoảng 4:1 của tập TripAdvisor, khiến mô hình có xu hướng dự đoán thiên về lớp đa số positive để tối đa hóa Accuracy tổng thể, trong khi bỏ sót một nửa số mẫu negative thực sự.

Hạn chế này khẳng định rằng chỉ riêng cơ chế học quần thể dựa trên Bootstrap chưa đủ để xử lý triệt để dữ liệu mất cân bằng nghiêm trọng, mô hình cần một cơ chế bổ sung khai thác cấu trúc mật độ thực tế của dữ liệu thay vì chỉ dựa trên phân phối nhãn. Đây chính là động lực trực tiếp dẫn đến đề xuất mô hình rừng mật độ quan điểm F_{DF}, tích hợp thuật toán phân cụm DBSCAN, được trình bày tại Chương 3. Khảo sát toàn diện và mới nhất về học bất cân bằng của [186] khẳng định vấn đề mất cân bằng phân lớp (class imbalance) tiếp tục là thách thức lớn trong học máy, đặc biệt ở các bài toán thực tiễn như phân tích tình cảm nơi một số lớp quan điểm xuất hiện hiếm hơn nhiều và chỉ ra rằng các phương pháp dựa trên cấu trúc dữ liệu như phân cụm mật độ thường hiệu quả hơn lấy mẫu quá mức (oversampling) đơn thuần trong trường hợp dữ liệu mất cân bằng nhiều.

Kết quả thực nghiệm từ Bảng 2.8 đến Bảng 2.10 cho thấy mô hình F_{CF} đạt Accuracy 0,80 - 0,82 trên tập dữ liệu Amazon Food Reviews và 0,91 trên TripAdvisor Hotel Reviews. Tuy nhiên, con số 0,91 trên TripAdvisor phản ánh hiện tượng nghịch lý độ chính xác do mất cân bằng nhãn 4:1, Recall chỉ đạt 0,50 và F1 = 0,66. Đối chiếu mô hình F_{CF} với các mô hình học máy khác (RF, GradBoost, SVM) cùng các mốc tham chiếu học sâu (LSTM, BiLSTM-Attention) và Transformer (BERT, RoBERTa – dùng văn bản gốc làm đầu vào) được trình bày tại Chương 3 và Chương 4 trên tập dữ liệu Yelp cùng bộ năm quan điểm.

Mô hình rừng phân loại quan điểm (F_{CF}) có độ phức tạp huấn luyện là $\Theta(n \times N \times mtry \times logN)$ và dự đoán là $\Theta(n \times log\ N)$ tương đương với Random Forest tiêu chuẩn [65], độ phức tạp mô hình F_{CF} ở mức thấp và chỉ yêu cầu CPU thông thường. F_{CF} là mô hình đơn giản và hiệu quả tính toán, phù hợp làm nền tảng cho các mô hình phức tạp hơn.

2.8. Công cụ sử dụng

Mô hình rừng phân loại quan điểm F_{CF} được triển khai bằng ngôn ngữ R phiên bản 4.3.2 trong môi trường RStudio 2023.12.1, với các gói chính: tree và rpart (xây dựng cây quyết định), randomForest [65], randomForestExplainer (độ sâu tối thiểu, tầm quan trọng và tương tác của đặc trưng tại Bảng 2.11, Bảng 2.12), pdp và ALEPlot (biểu đồ phụ thuộc từng phần và phân vùng không gian hai chiều tại Hình 2.11–2.14), boot (ước lượng bootstrap), dplyr và tidyr (xử lý dữ liệu), ggplot2 (trực quan hóa). Danh sách chi tiết các thư viện và phiên bản được trình bày trong Bảng D.1 (Phụ lục D).

2.9. Kết luận chương

Chương 2 đã trình bày chi tiết mô hình khai phá quan điểm với rừng phân loại. Mô hình dựa trên sự kết hợp các cây quyết định để phân loại quan điểm là tích cực hay tiêu cực. Mỗi cây được xây dựng trên tập dữ liệu bộ năm quan điểm. Kết quả thử nghiệm trên hai tập dữ liệu đánh giá thực phẩm (Amazon Food Reviews) và đánh giá khách sạn (TripAdvisor Hotel Reviews) dựa trên các kịch bản khác nhau để khai thác các đặc trưng quan trọng của các đánh giá cho thấy hiệu quả của mô hình đề xuất trong giải quyết vấn đề ảnh hưởng theo tầm quan trọng của các lớp quan điểm khác nhau. Ngoài ra, kết quả cũng có thể diễn giải, khái quát hóa ảnh hưởng của các tham số quan trọng có liên quan đến việc ra quyết định của mô hình rừng phân loại quan điểm trên bộ dữ liệu đánh giá khách sạn. Tuy nhiên, F_{CF} học trên toàn bộ không gian đặc trưng như một khối đồng nhất, chỉ dựa vào Bootstrap nên còn hạn chế trên dữ liệu mất cân bằng (trên TripAdvisor Hotel Reviews, Accuracy 0,91 nhưng Recall chỉ 0,50) và chưa khai thác cấu trúc cụm tự nhiên của các quan điểm. Hạn chế này dẫn đến Chương 3: mô hình rừng mật độ quan điểm F_{DF} phân cụm dữ liệu theo mật độ trước khi xây dựng rừng phân loại cho từng cụm.

CHƯƠNG 3. KHAI PHÁ QUAN ĐIỂM VỚI RỪNG MẬT ĐỘ

3.1. Giới thiệu

Chương này trình bày kết quả nghiên cứu, phát triển và xây dựng mô hình, thuật toán khai phá quan điểm với rừng mật độ, nhằm giải quyết bài toán khai phá quan điểm trên dữ liệu đánh giá có cấu trúc phân nhóm tự nhiên, khác với rừng phân loại ở Chương 2 được xây dựng trên dữ liệu phân bố đồng đều. Xuất phát từ quan sát thực tế, dữ liệu quan điểm thường tập trung thành các nhóm khác biệt về đặc điểm và phân bố tình cảm. Để khai thác cấu trúc này, mô hình đề xuất kết hợp phân cụm DBSCAN để phát hiện cụm dựa trên mật độ với học quần thể từ nhiều rừng quyết định, trong đó mỗi rừng được xây dựng và tối ưu riêng cho một cụm mật độ cụ thể. Chương này đồng thời đóng góp một số điểm mới quan trọng, bao gồm việc xây dựng cây mật độ dựa trên entropy không giám sát, kiến trúc rừng mật độ với cơ chế gộp xác suất từ các rừng theo cụm, phương pháp diễn giải mô hình thông qua phân tích tầm quan trọng và tương tác đặc trưng, cũng như khả năng tự động xử lý nhiễu và điểm ngoại lai. Theo số liệu đã công bố (chia ngẫu nhiên theo dòng bộ năm), mô hình rừng mật độ đạt độ chính xác cao hơn mô hình rừng ngẫu nhiên không phân cụm (khi chia theo đánh giá gốc tại Phụ lục J, hai mô hình tương đương), đạt sự cân bằng tốt giữa độ chính xác và độ bao phủ, đồng thời cung cấp các phân tích diễn giải [75] có ý nghĩa về vai trò của từng đặc trưng trong dự đoán tình cảm.

Nội dung chính của chương bao hàm các kết quả nghiên cứu của Bài báo số 4, Bài báo số 5.

Phát biểu bài toán. Cho tập dữ liệu quan điểm $O = \{o_1, o_2, o_3, \ldots, o_N\}$, mỗi $o_i = (e_i, a_i, s_i, h_i, t_i)$ gồm thực thể e_i, khía cạnh a_i, giá trị tình cảm s_i, người giữ quan điểm h_i và thời điểm t_i. Sau bước chuẩn hóa Z-score §3.2.2 theo công thức (3.4), tạo thành $O^{norm} \in \mathbb{R}^{\mathrm{Nx5}}$. Mô hình phân cụm mật độ trên O^{norm} (bốn cột đặc trưng e, a, h, t; nhãn s được tách riêng) để phát hiện K nhóm quan điểm có phân bố tương đồng $\mathcal{C} = \{C_1, \ldots, C_K, Noise\}$, sau đó xây dựng rừng phân loại riêng trong từng cụm.

Đầu vào: Ma trận $O^{norm} \in \mathbb{R}^{\mathrm{Nx5}}$ đã chuẩn hóa.

Đầu ra: Nhãn phân loại $\hat{y} \in C = \{\text{positive, negative}\}$, tổng hợp từ K rừng phân loại cụm. Mô hình đồng thời cung cấp thông tin diễn giải $\hat{c} \in \{1, .., K\}$ - chỉ số cụm của từng quan điểm.

Mục tiêu: Tối thiểu hóa cùng hàm sai số $\mathcal{L}$ như Chương 2. Mô hình rừng mật độ quan điểm (F_{DF}) nhằm cải thiện độ chính xác phân loại trên dữ liệu mất cân bằng nhờ phân cụm mật độ trước khi phân loại.

Luận án sử dụng hai nhãn nhị phân {positive, negative} thay vì thang điểm nhiều mức vì hai lý do: (1) Hầu hết các ứng dụng khai phá quan điểm trong thương mại điện tử (Amazon, Yelp) và hệ thống khuyến nghị quan tâm chính đến chiều phân cực tích cực/tiêu cực để phân loại đánh giá hữu ích; (2) Dữ liệu đánh giá dạng sao (★) có phân phối nghiêng mạnh về đầu 4★-5★ và đánh giá 3★ thường phản ánh cảm xúc trung lập không ổn định, việc đưa neutral vào sẽ tăng nhiễu hơn là tăng thông tin phân biệt lớp. Quy tắc nhị phân hóa: 1★-2★ là negative, 4★-5★ là positive, 3★ được loại bỏ đối với Amazon Food Reviews và TripAdvisor; riêng hai tập Yelp, 3★ được gộp vào negative (xem §3.7.2.2, Bảng 3.3).

3.2. Biểu diễn và chuẩn hóa bộ năm quan điểm

3.2.1. Biểu diễn bộ năm quan điểm

Xét tập dữ liệu gồm n bộ năm quan điểm:

$$O = \{o_1, o_2, \dots, o_n\},$$

trong đó mỗi quan điểm o_i được biểu diễn dưới dạng véc-tơ năm chiều:

$$o_i = (e_i, a_i, s_i, h_i, t_i)^T \in \mathbb{R}^5,$$

Mỗi bộ năm quan điểm o được ánh xạ sang một véc-tơ số trong không gian đặc trưng như sau:

$$\mathrm{o} = [\mathrm{ID}_e, \mathrm{ID}_a, \mathrm{Score}_s, \mathrm{ID}_h, \mathrm{Days}_t]^T \tag{3.1}$$

trong đó:

- ID_e: mã hóa thực thể;
- ID_a: mã hóa khía cạnh;
- Score_s: điểm tình cảm;
- ID_h: mã hóa chủ thể đánh giá;

• Days_t: đặc trưng thời gian, số ngày tính từ mốc chuẩn.

3.2.2. Mã hóa và chuẩn hóa quan điểm

Các thành phần rời rạc e_i, a_i, h_i được ánh xạ sang không gian số thông qua hàm mã hóa ở công thức (3.2).

$$\text{ID}: \mathcal{X} \to \mathbb{N}, \tag{3.2}$$

trong đó $\mathcal{X}$ là tập các giá trị rời rạc tương ứng.

Thành phần thời gian t_i được chuyển đổi về số ngày tính từ một mốc chuẩn ở công thức (3.3).

$$\text{Days}(t_i) = t_i - t_0, \tag{3.3}$$

Giá trị tình cảm s_i được chuẩn hóa về khoảng liên tục $[-1,1]$ hoặc $[0, 1]$ tùy theo thang đo ban đầu.

Do các đặc trưng trong véc-tơ o_i có thang đo khác nhau, cần tiến hành chuẩn hóa để tránh hiện tượng một chiều chi phối khoảng cách.

Áp dụng chuẩn hóa Z-score cho từng chiều k.

$$o_{\text{norm}}^{(k)} = \frac{o^{(k)} - \mu_k}{\sigma_k}, \tag{3.4}$$

trong đó:

• μ_k: giá trị trung bình của chiều k,

• σ_k: độ lệch chuẩn của chiều k.

Sau bước mã hóa, mỗi quan điểm được ánh xạ vào không gian Euclid năm chiều. Để tránh sự chi phối của các chiều có thang đo lớn, toàn bộ dữ liệu được chuẩn hóa về cùng miền giá trị. Không gian đặc trưng sau chuẩn hóa ở công thức (3.5).

$$\mathbf{O}_{\text{norm}} \subset \mathbb{R}^5. \tag{3.5}$$

Đây là không gian đầu vào cho các bước phân cụm và mô hình hóa mật độ tiếp theo. Lưu ý về ký hiệu: véc-tơ năm chiều ở công thức (3.1) là biểu diễn đầy đủ của bộ năm; trong bài toán phân loại của luận án, thành phần tình cảm s chính là nhãn cần dự đoán nên được tách khỏi đầu vào. Vì vậy, phân cụm DBSCAN, các rừng theo cụm và giá trị SHAP trong thực nghiệm được tính trên véc-tơ bốn chiều (e, a, h, t) (quy tắc MinPts = d + 1 với d = 4 tại mục 3.7.3); ký hiệu năm chiều được giữ để mô tả cấu trúc bộ năm.

3.3. Mô hình hóa rừng mật độ quan điểm

3.3.1. Phân cụm mật độ quan điểm bằng DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [72][69] để phân cụm không gian dựa trên mật độ, phát hiện cấu trúc phân bố tự nhiên. Thuật toán DBSCAN được xác định bởi hai tham số chính: ε là bán kính vùng lân cận và MinPts là số lượng điểm tối thiểu để hình thành một vùng mật độ cao.

Đánh giá thực nghiệm quy mô lớn gần đây của [184] so sánh các phương pháp phát hiện ngoại lệ dựa trên phân cụm (bao gồm DBSCAN) với các phương pháp cổ điển, cho thấy nhóm phương pháp dựa trên phân cụm có lợi thế về hiệu quả tính toán, khả năng sử dụng các độ đo đánh giá và diễn giải tốt hơn.

Vùng lân cận ε của một điểm $o \in O$ được định nghĩa tại công thức (3.6).

$$N_\varepsilon(o) = \{o' \in O \mid d(o, o') \leq \varepsilon\}. \tag{3.6}$$

Khoảng cách giữa hai quan điểm o_i, o_j được tính bằng khoảng cách Euclid tại công thức (3.7).

$$d(o_i, o_j) = \sqrt{\sum_{k=1}^{5} \left(o_i^{(k)} \quad o_j^{(k)}\right)^2}, \tag{3.7}$$

Một điểm o được gọi là điểm lõi (core point) nếu:

$$| N_\varepsilon(o) | \geq \text{MinPts}.$$

Một điểm o_2 được gọi là có thể đạt được theo mật độ từ o_1 nếu tồn tại một chuỗi các điểm:

$$o_1 \to p_1 \to p_2 \to \cdots \to o_2,$$

trong đó mỗi điểm trung gian p_i đều là điểm lõi và $p_{i+1} \in N_\varepsilon(p_i)$.

Một cụm mật độ C_k được định nghĩa là tập hợp các điểm liên thông theo mật độ từ một điểm khởi tạo o_{seed}:

$$C_k = \{o \in O \mid o \text{ liên thông theo mật độ với } o_{\text{seed}} \in C_k\}.$$

Các điểm không thuộc bất kỳ cụm nào được xác định là nhiễu theo công thức (3.8).

$$\text{Noise} = \{o \in O \mid o \notin C_k, \forall k\}. \tag{3.8}$$

Các cụm C_k thu được là cơ sở để xây dựng các rừng mật độ cục bộ.

3.3.2. Cây mật độ quan điểm

Khảo sát toàn diện về phát triển của thuật toán phân cụm theo sự tập trung các đỉnh mật độ (DPC) từ năm 2014 đến năm 2023 của [194] đã tổng kết hơn 110 bài báo cho thấy việc xác định đỉnh mật độ như trung tâm cụm là hướng nghiên cứu chủ đạo và mạnh mẽ trong phân cụm không giám sát, đặc biệt khi dữ liệu có cấu trúc hình học đa dạng. Mô hình cây mật độ quan điểm trong luận án kế thừa ý tưởng này, sử dụng đỉnh mật độ để xác định nút phân chia trong cây.

Cây mật độ [65] được huấn luyện độc lập và sử dụng tối ưu hóa nút ngẫu nhiên. Do đó, việc tối ưu hóa nút tách thứ j được thực hiện ở công thức (3.9).

$$\Theta_j^* = \underset{\Theta \in \tau_j}{\operatorname{argmax}}\, IG_D\left(O_j, O_{jL}, O_{jR}\right) \tag{3.9}$$

chọn Θ sao cho độ lợi thông tin ΔH_D lớn nhất được xác định ở công thức (3.10).

$$IG_D\left(O_j, O_{jL}, O_{jR}\right) = H_D\left(O_j\right) - \frac{\left|O_{jL}\right|}{\left|O_j\right|} H_D\left(O_{jL}\right) - \frac{\left|O_{jR}\right|}{\left|O_j\right|} H_D\left(O_{jR}\right) \tag{3.10}$$

trong đó O_{jL}, O_{jR} là hai tập con sau khi tách.

Để xác định đầy đủ mô hình cây mật độ, cần xác định dạng chính xác Entropy H_D của một tập hợp các điểm huấn luyện O. Trong trường hợp cây mật độ, dữ liệu không được dán nhãn nên việc xác định Entropy không giống như cây phân loại. Do đó, cần xác định một Entropy không giám sát, tức là một Entropy H_D áp dụng cho dữ liệu không dán nhãn. Vì vậy, sử dụng các phân phối Gaussian đa biến tại các nút, Entropy $H_D(O)$ được cho bởi công thức (3.11)

$$H_D(O) = \frac{1}{2}\log((2\pi e)^d |\Lambda(O)|) \tag{3.11}$$

Trong đó Λ liên quan đến ma trận hiệp phương sai $d \times d$. Do đó, mức tăng thông tin trong công thức (3.10) được cho bởi công thức (3.12).

$$I\left(O_j, \theta\right) = \log\left(\left|\Lambda\left(O_j\right)\right|\right) - \sum_{i \in \{L,R\}} \frac{\left|O_j^i\right|}{\left|O_j\right|} \log\left(\left|\Lambda\left(O_j^i\right)\right|\right) \tag{3.12}$$

Với |.| là định thức của ma trận.

Khi áp dụng công thức (3.11)–(3.12) cho các đặc trưng định danh (Entity, Aspect, Holder, Time) đã được mã hóa số thứ tự, ma trận hiệp phương sai Λ(O) được

diễn giải như hiệp phương sai thực nghiệm trên không gian số đã mã hóa — tương tự cách các phương pháp phân cụm dựa trên khoảng cách (k-means, DBSCAN) xử lý dữ liệu định danh đã mã hóa số và không giả định các đặc trưng này thực sự tuân theo phân phối Gaussian liên tục; đây là một xấp xỉ nhằm đo độ phân tán/đồng nhất cục bộ tại mỗi nút. Tại các nút lá có số mẫu nhỏ (|O| xấp xỉ hoặc nhỏ hơn số chiều d) hoặc có cột hằng, khiến |Λ(O)| tiến về 0: Thuật toán 3.1 (BuildDensityTree) đã có điều kiện dừng đệ quy khi |O| < min_split hoặc depth = max_depth (mục 3.4.1), giúp tránh việc phân chia tiếp tại các nút suy biến; tuy nhiên luận án chưa áp dụng cơ chế chính quy hóa tường minh (để tăng độ ổn định số học khi |O| gần ngưỡng min_split, đây là hướng cần bổ sung ở các nghiên cứu tiếp theo).

Tập hợp các lá trong cây thứ t trong một khu rừng xác định một phân vùng dữ liệu tại nút lá ở công thức (3.13)

$$l(o): \mathbb{R}^d \rightarrow \ L \subset \mathbb{N} \tag{3.13}$$

Trong đó, $l(o)$ biểu diễn lá chứa điểm đầu vào o và L là tập hợp tất cả các lá trong một cây nhất định. Thống kê của tất cả các điểm huấn luyện đến từng nút lá được tóm tắt bằng một phân phối Gaussian đa biến [73] $\mathcal{N}\left(o, \mu_{l(o)}, \Lambda_{l(o)}\right)$. Đầu ra của cây thứ t được mô hình hóa trong công thức (3.14).

$$p_t(o) = \frac{\pi_{l(o)}}{Z_t} \mathcal{N}\left(o; \mu_{l(o)}, \Lambda_{l(o)}\right) \tag{3.14}$$

Vectơ μ_l biểu diễn giá trị trung bình của tất cả các điểm đạt tới lá l và Λ_l là ma trận hiệp phương sai liên quan. Vô hướng π_l là tỷ lệ của tất cả các điểm huấn luyện đạt tới lá l, nghĩa là $\pi_l = \frac{|O_l|}{|O|}$

Trong công thức (3.14), mỗi Gaussian bị cắt bớt bởi các ranh giới của ô phân vùng được liên kết với lá tương ứng. Vì vậy, để đảm bảo chuẩn hóa xác suất cần kết hợp hàm phân vùng Z_t, được định nghĩa ở công thức (3.15)

$$Z_t = \int_o \left(\sum_l \pi_l \, \mathcal{N}(o; \mu_l, \Lambda_l) p(l|o) \right) do \tag{3.15}$$

Tuy nhiên, trong một cây mật độ, mỗi điểm dữ liệu đạt đến chính xác một nút đầu cuối trên mỗi cây. Như vậy, điều kiện $p(l|o)$ là một hàm delta $p(l|o) = \delta(l = l(o))$ do đó công thức (3.15) trở thành công thức (3.16).

$$Z_t = \int_o \pi_{l(o)} \mathcal{N}(o; \mu_{l(o)}, \Lambda_{l(o)}) do. \quad (3.16)$$

Đối với các bộ học yếu phức tạp hơn, có thể dễ dàng ước tính Z_t bằng tích phân số trên một lưới thông thường ở công thức (3.17)

$$Z_t \approx \Delta . \sum_i \pi_{l(o_i)} \mathcal{N}\left(o_i; \mu_{l(o_i)}, \Lambda_{l(o_i)}\right) \quad (3.17)$$

Trong đó, Δ biểu diễn độ dài, diện tích, thể tích,... tùy thuộc vào chiều của miền; Các điểm o_i được tạo trên mạng lưới thường hữu hạn với khoảng cách Δ.

3.3.3. Rừng mật độ quan điểm

Rừng mật độ quan điểm là một tập hợp các cây mật độ quan điểm được huấn luyện ngẫu nhiên, các lá của rừng chứa các mô hình dự đoán quan điểm. Rừng mật độ quan điểm được tính bằng trung bình cộng của tất cả các cây mật độ ở công thức (3.18).

$$p(o) = \frac{1}{T} \sum_{t=1}^{T} p_t(o) \quad (3.18)$$

Với mật độ mỗi cây là $p_t(o)$

3.4. Mã giả và đồ thị tính toán mô hình rừng mật độ quan điểm

3.4.1. Thuật toán và độ phức tạp

Thuật toán BuildDensityForest (xây dựng rừng mật độ với DBSCAN) và PredictAndOrientation (thuật toán dự đoán kèm định hướng quan điểm và diễn giải bằng SHAP) được trình bày như sau:

Thuật toán 3.1. BuildDensityForest

Đầu vào:

O = $\{o_i\}_{i=1}$…N : tập dữ liệu huấn luyện, xây cây theo cấu trúc mật độ không giám sát với: $o_i = (e_i, a_i, s_i, h_i, t_i)$ là bộ năm quan điểm thứ i. (Thuật toán *dùng đặc trưng (e_i, a_i, h_i, t_i) để phân tách; s_i được bảo lưu cho giai đoạn dự đoán).*

n : số cây trong rừng

ε (eps) : bán kính lân cận DBSCAN

MinPts : số điểm tối thiểu trong lân cận ε

max_depth : độ sâu tối đa của cây

min_split : số mẫu tối thiểu để phân tách

Đầu ra:

$F_{DF} = \{T_1, T_2, \ldots, T_n\}$: rừng mật độ quan điểm

1: $F_{DF} \leftarrow \emptyset$

2: for i = 1 to n do

3: $O_i^* \leftarrow$ Bootstrap(O, N) // lấy mẫu có hoàn lại — Bootstrap sampling

4: $T_i \leftarrow$ BuildDensityTree(O_i^*, max_depth, ε, MinPts, depth=0)

5: $F_{DF} \leftarrow F_{DF} \cup \{T_i\}$

6: end for

7: return F_{DF}

Hàm phụ BuildDensityTree(O, max_depth, ε, MinPts, depth):

// Bước 1- Kiểm tra điều kiện dừng: phân cụm DBSCAN tại nút lá

1: if depth = max_depth or |O| < min_split then

2: $\mathcal{C}_{leaf} \leftarrow$ DBSCAN(O, ε, MinPts) *// phân cụm mật độ tại nút lá*

3: $p_{leaf} \leftarrow$ FitGaussianMixture($\mathcal{C}_{leaf}$) *// ước tính mật độ xác suất Gaussian*

4: return Leaf(p_{leaf}, $\mathcal{C}_{leaf}$)

5: end if

// Bước 2- Chọn ngẫu nhiên tập đặc trưng ứng viên

6: Features ← RandomSubset(χ, M) *// χ = {e, a, h, t}*

// Bước 3-Tìm phân tách tối ưu theo Entropy mật độ

7: best_gain ← $-\infty$

8: for each f ∈ Features, θ ∈ thresholds(O, f) do

9: $O_L \leftarrow \{o \in O : o[f] \leq \theta\}$; $O_R \leftarrow \{o \in O : o[f] > \theta\}$

10: gain ← ComputeIG(O, O_L, O_R) *// theo công thức (3.11) và (3.12)*

11: if gain > best_gain then

12: best_gain ← gain

13: $f^* \leftarrow f$; $\theta^* \leftarrow \theta$

14: end if

15: end for

// Bước 4 - Phân tách dữ liệu và gọi đệ quy

16: $O_L \leftarrow \{ o \in O : o[f^*] \leq \theta^* \}$

17: $O_R \leftarrow \{ o \in O : o[f^*] > \theta^* \}$

18: $T_L \leftarrow$ BuildDensityTree(O_L, max_depth, ε, MinPts, depth+1)

19: $T_R \leftarrow$ BuildDensityTree(O_R, max_depth, ε, MinPts, depth+1)

20: return Node(f*, θ*, T_L, T_R)

Độ phức tạp tính toán.

Độ phức tạp của thuật toán xây dựng rừng mật độ gồm hai thành phần độc lập, thực hiện tại hai giai đoạn khác nhau trong quá trình xây dựng cây.

Thành phần 1 - Chi phí xây dựng cây phân tách tương tự như rừng phân loại quan điểm ở công thức (2.12). Với *n* cây trong rừng, mỗi cây được xây trên tập Bootstrap *N* mẫu, tại mỗi nút chọn ngưỡng tối ưu theo Entropy mật độ qua *logN* tầng phân tách, mỗi tầng xét m_{try} đặc trưng ngẫu nhiên.

Thành phần 2 - Chi phí phân cụm DBSCAN tại các nút lá theo công thức:

$$\Theta(n \times L \times N \times logN) \tag{3.19}$$

Thành phần này phát sinh do DBSCAN được gọi tại mỗi nút lá sau khi cây đã phân tách xuống tầng sâu nhất. Điểm quan trọng là DBSCAN không chạy trên toàn bộ *N* mẫu mà chỉ chạy trên tập con O_l tại nút lá đó.

Trong đó:

- *n*: số lượng cây trong rừng mật độ;
- *N*: số lượng mẫu huấn luyện;
- m_{try}: số đặc trưng được chọn ngẫu nhiên tại mỗi nút phân tách;
- *L:* số nút lá trung bình của một cây, xấp xỉ N/min_split;

Độ phức tạp tổng thể của thuật toán xây dựng rừng mật độ được xác định bởi sự kết hợp giữa chi phí xây dựng cây và chi phí phân cụm mật độ DBSCAN tại các nút lá được biểu diễn theo công thức (3.20).

$$\Theta\left(n \times m_{try} \times N \times logN\right) + \Theta(n \times L \times N \times logN) \tag{3.20}$$

Thuật toán 3.2. PredictAndOrientation

Đầu vào*:*

$F_{DF} = \{T_1, \ldots, T_n\}$: rừng mật độ đã huấn luyện

o = (e, a, s, h, t) : mẫu quan điểm cần phân loại

aspects = $\{a_1, \ldots, a_k\}$: tập khía cạnh cần định hướng

***Đầu ra**:*

ŷ : nhãn phân loại tổng thể

orientation_per_aspect : định hướng tình cảm theo từng khía cạnh

shap_values : $\varphi = (\varphi_e, \varphi_a, \varphi_s, \varphi_h, \varphi_t)$ là giá trị Shapley

// Bước 1 — Tổng hợp xác suất mật độ từ rừng

p(o) ← ComputeDensity(FDF, o) // trung bình mật độ xác suất qua *n* cây

ŷ ← $\text{argmax}_y p(y|o)$

// Bước 2 — Định hướng quan điểm theo từng khía cạnh

orientation_per_aspect ← { }

for each a_k in aspects do

  o_k ← project(o, aspect = a_k) // chiếu bộ năm lên khía cạnh a_k

  $\hat{y}_k \leftarrow \text{argmax}_y p(y|o_k)$

  orientation_per_aspect[a_k] ← $\hat{y}_k$

end for

// Bước 3 — Diễn giải bằng giá trị SHAP

$\varphi \leftarrow$ SHAP_TreeExplainer(F_{DF}, o) // $\varphi = (\varphi_e, \varphi_a, \varphi_s, \varphi_h, \varphi_t)$

shap_values ← φ

return (ŷ, orientation_per_aspect, shap_values)

Độ phức tạp của thuật toán dự đoán và định hướng quan điểm được xác định bởi ba giai đoạn thực thi độc lập. Với *n* cây trong rừng, *K* khía cạnh cần định hướng và *N* mẫu huấn luyện, độ phức tạp tổng thể được biểu diễn theo từng giai đoạn như sau:

Giai đoạn 1-Tổng hợp mật độ xác suất được tính theo công thức:

$$\Theta(n \times logN) \tag{3.21}$$

Mỗi cây duyệt từ gốc đến lá với chiều sâu trung bình *logN*, lấy trung bình mật độ xác suất từ *n* cây. Chi phí này tương đương Thuật toán 2.2 (PredictClassificationForest) của rừng phân loại.

Giai đoạn 2- Định hướng tình cảm theo từng khía cạnh:

$$\Theta(K \times n \times logN) \tag{3.22}$$

Với mỗi khía cạnh a_k trong tập khía cạnh A, thuật toán chiếu bộ năm o lên khía cạnh đó rồi duyệt lại toàn bộ n cây một lần. Chi phí tăng tuyến tính theo K: chẳng hạn khi K = 10 khía cạnh, giai đoạn 2 tốn gấp 10 lần so với giai đoạn 1.

Giai đoạn 3 — Tính giá trị SHAP [75] theo công thức:

$$\Theta\left(n \times L \times d^2\right) \tag{3.23}$$

Trong đó:

- n: số lượng cây trong rừng mật độ;
- N: số lượng mẫu dữ liệu huấn luyện được sử dụng để xây dựng cây;
- $K = |A|$: số lượng khía cạnh cần định hướng tình cảm – tương ứng với tập hợp các giá trị của thành phần a trong bộ năm quan điểm;
- L: số nút lá trung bình của một cây.
- *d: chiều sâu tối đa của cây; theo thuật toán TreeSHAP, chi phí giải thích một mẫu trên mỗi cây là $\Theta(L \times d^2)$ với d là chiều sâu cây (không phải số đặc trưng); với cây cân bằng, $d \approx logN$.*

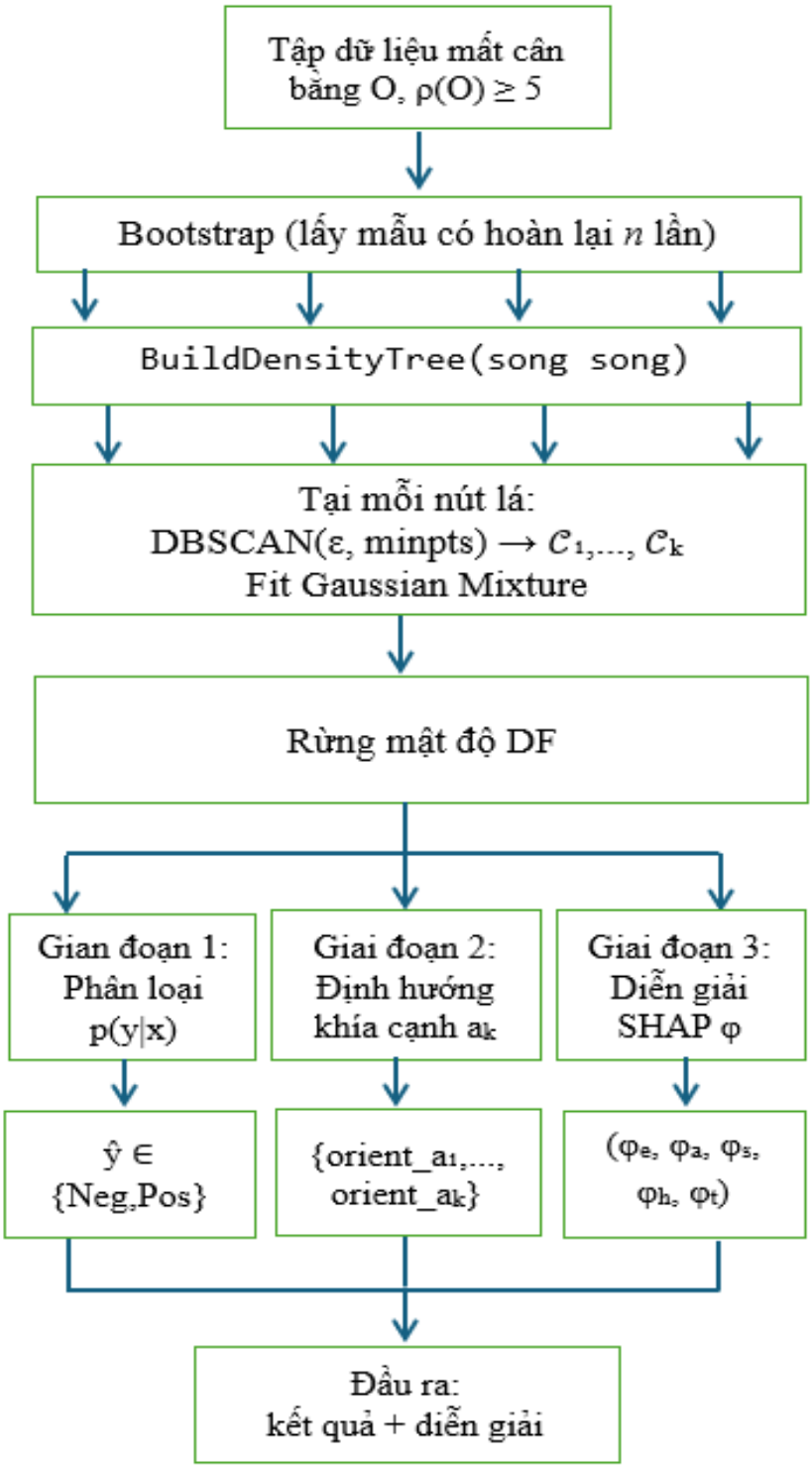
Tổng hợp ba giai đoạn trên, độ phức tạp tổng thể của thuật toán dự đoán và định hướng quan điểm là:

$$\Theta(n \times logN) + \Theta(K \times n \times logN) + \Theta\left(n \times L \times d^2\right) \tag{3.24}$$

3.4.2. Đồ thị tính toán mô hình rừng mật độ quan điểm

Đồ thị tính toán được đưa vào để cấu trúc luồng dữ liệu và phân cấp được biểu diễn trực quan ở Hình 3.1.

Đồ thị tính toán này cho thấy đặc điểm khác biệt của mô hình rừng mật độ so với rừng phân loại truyền thống. Tại mỗi nút lá, thay vì chỉ lưu nhãn đa số, thuật toán DBSCAN được áp dụng để các cụm trong tập con dữ liệu được phát hiện theo mật độ, sau đó hàm mật độ xác suất Gaussian Mixture được khớp để phân phối xác suất p(o) tại nút lá được biểu diễn. Trong giai đoạn dự đoán, ba luồng song song được thực thi: (1) phân loại tổng thể; (2) định hướng theo từng khía cạnh trong bộ năm quan điểm; (3) diễn giải bằng SHAP để tầm quan trọng của từng đặc trưng được phân tích.

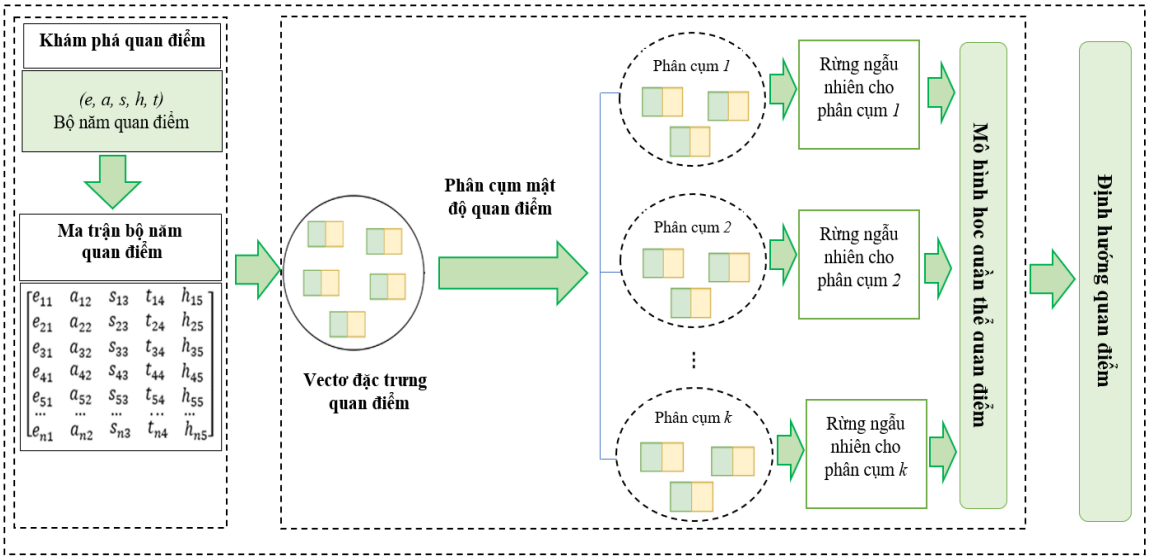


Hình 3.1: Đồ thị tính toán của mô hình rừng mật độ quan điểm

3.5. Mô hình tổng quan khai phá quan điểm với rừng mật độ

Khai phá quan điểm với rừng mật độ được thiết kế dựa trên kiến trúc ba mô-đun như minh họa trong Hình 3.2, trong đó mỗi mô-đun đảm nhận một vai trò cụ thể trong quy trình xử lý. Khác với mô hình rừng phân loại ở Chương 2 xử lý toàn bộ dữ liệu như một khối đồng nhất, kiến trúc mới này phân tách dữ liệu thành các cụm mật

độ trước khi xây dựng mô hình, từ đó tận dụng cấu trúc phân nhóm tự nhiên của dữ liệu quan điểm.



Hình 3.2: Đề xuất mô hình rừng mật độ quan điểm

Sơ đồ tổng quan khai phá quan điểm với rừng mật độ gồm ba bước theo quy trình tuần tự:

Bước 1: chuyển đổi dữ liệu văn bản thô thành ma trận số, trong đó mỗi hàng là một vector đặc trưng năm chiều đại diện cho một quan điểm.

Bước 2: thực hiện hai nhiệm vụ chính: (1) phân tích cấu trúc dữ liệu bằng DBSCAN để phát hiện các cụm mật độ tự nhiên và (2) xây dựng một rừng mật độ riêng biệt cho mỗi cụm được phát hiện.

Bước 3: tổng hợp dự đoán từ tất cả các rừng thông qua tính trung bình của tất cả các cây mật độ.

3.6. Diễn giải rừng mật độ quan điểm

3.6.1. SHAP

SHAP (SHapley Additive exPlanations) [125] là phương pháp để giải thích các dự đoán cá thể. Đặt f là mô hình dự đoán ban đầu được giải thích và g là mô hình giải thích. Mô hình tập trung giải thích dự đoán $f(o)$ dựa trên một đầu vào duy nhất o. Mô hình được diễn giải lấy đầu vào o' để ánh xạ tới các đầu vào ban đầu thông qua hàm ánh xạ $o = h_o(o')$. Các phương pháp cục bộ cố gắng đảm bảo $g(z') \approx f\left(h_o(z')\right)$, với $z' \approx o'$. SHAP giải thích ở công thức (3.25).

$$g(z') = \varphi_0 + \sum_{j=1}^{M} \varphi_j z'_j \tag{3.25}$$

Trong đó, g là mô hình giải thích, $z' \in \{0,1\}^M$ là vectơ nhị phân biểu diễn nếu có sự hiện diện của đặc trưng j thì có giá trị là 1 hoặc vắng mặt đặc trưng j thì có giá trị là 0, M là số đặc trưng đầu vào, φ_0 là giá trị kỳ vọng của mô hình (giá trị cơ sở) được tính trung bình của tất cả dự đoán có thể và $\varphi_j \in \{0,1\}^M$ là giá trị Shapley cho đặc trưng j.

Giá trị Shapley là sự đóng góp của từng đặc trưng quan điểm vào quyết định cuối cùng của mô hình rừng mật độ quan điểm được tính trọng số và tổng trên tất cả các kết hợp có thể có của các giá trị đặc trưng ở công thức (3.26).

$$\varphi_j = \sum_{S \subseteq F \setminus \{j\}} \frac{|S|!\,(|F| - |S| - 1)!}{|F|!} \left[f_{S\cup\{j\}}\left(o_{S\cup\{j\}}\right) - f_S(o_S) \right] \tag{3.26}$$

Trong đó F là tập hợp tất cả các đặc trưng quan điểm. S là tập con của F không chứa đặc trưng j. $|S|$ là số lượng phần tử trong tập S. $|F|$ là số lượng phần tử trong tập F. $f_S(o_S)$ là giá trị dự đoán của mô hình khi chỉ sử dụng các đặc trưng trong tập S. $f_{S\cup\{j\}}\left(o_{S\cup\{j\}}\right)$ là giá trị dự đoán của mô hình khi sử dụng các đặc trưng trong tập S cùng với đặc trưng j.

3.6.2. Tầm quan trọng của đặc trưng SHAP

Trong học máy, nhiệm vụ quan trọng là dự đoán biến phản hồi dựa trên một tập hợp các biến dự đoán, lý thuyết về tầm quan trọng [125][74] của lượng tạp chất giảm trung bình thu được từ các cây ngẫu nhiên. Tầm quan trọng của các đặc trưng được tính toán thông qua giá trị SHAP. Trong trường hợp này, việc tính toán tầm quan trọng tổng quát của mô hình được tính trung bình các giá trị Shapley cơ bản [125] cho từng đặc trưng quan điểm trên tất cả các mẫu quan điểm ở công thức (3.27).

$$I_j = \frac{1}{|O|} \sum_{i=1}^{|O|} \varphi_j^{(k)} \tag{3.27}$$

Với I là tầm quan trọng của đặc trưng thông qua giá trị SHAP và j là chỉ số của đặc trưng. |O| là số lượng mẫu trong tập dữ liệu bộ năm quan điểm. $\varphi_j^{(k)}$ là giá trị SHAP của đặc trưng j cho mẫu k.

3.6.3. Giá trị tương tác SHAP

Các đặc trưng tương tác thông qua tương tác đặc trưng kết hợp bổ sung sau khi tính đến các tương tác đặc trưng riêng lẻ, tức là các phần tách xuất hiện trong cây con tối đa cho một trong các đặc trưng được chọn. Chỉ số tương tác Shapley [133] được định nghĩa ở công thức (3.28).

$$I_{i,j}(o) = \sum_{S \subseteq F \setminus \{i,j\}} \frac{|S|!\,(|F| - |S| - 2)!}{2(|F| - 1)!} \delta_{ij}(o, S) \tag{3.28}$$

khi $i \neq j$ và

$$\delta_{ij}(o, S) = f_o(S \cup \{i, j\}) - f_o(S \cup \{i\}) - f_o(S \cup \{j\}) + f_o(S) \tag{3.29}$$

Trong đó, $I_{i,j}(o)$ là giá trị tương tác SHAP giữa hai đặc trưng i và j cho mẫu quan điểm o. F là tập hợp tất cả các đặc trưng. $|F|$ là số lượng đặc trưng quan điểm. S là tập con của F không chứa đặc trưng quan điểm i và j.

3.7. Thực nghiệm

3.7.1. Tập dữ liệu sử dụng

Các thử nghiệm được thực hiện trên hai bộ dữ liệu *yelpHotelData* và *yelpResData* về các đánh giá khách sạn và nhà hàng. Tập dữ liệu *YelpHotelData* bao gồm 3 bảng dữ liệu có tên là Hotel, Review và Reviewer. Tương tự, tập dữ liệu YelpResData cũng bao gồm 3 bảng dữ liệu có tên là Restaurant, Review và Reviewer, chi tiết của hai tập dữ liệu được trình bày ở phần Phụ lục A, Phụ lục B.

3.7.2. Tiền xử lý dữ liệu

3.7.2.1. Tiền xử lý dữ liệu yelpHotelData

Tập dữ liệu *yelpHotelData* ban đầu bao gồm 3 bảng dữ liệu Hotel, Review và Reviewer có mối quan hệ với nhau. Dựa vào mối quan hệ giữa các cột dữ liệu trong các bảng để gộp các bảng dữ liệu này thành một bảng. Sau đó, các đặc trưng quan trọng trong dữ liệu đã chọn bao gồm tên khách sạn, khía cạnh của khách sạn, người đánh giá và thời gian bằng cách chuyển đổi các cột khía cạnh của khách sạn thành các hàng và giữ nguyên các cột còn lại. Đối với các khía cạnh của khách sạn, đã khám phá ba khía cạnh ảnh hưởng đến đánh giá khách sạn gồm AcceptsCreditCards, PriceRange, WiFi. Kết quả chuyển đổi từ dữ liệu gốc ban đầu có 688.329 bản ghi thành cấu trúc dữ liệu bộ năm quan điểm có 882.474 mẫu, chi tiết các đặc trưng được liệt kê trong Bảng 3.1. Trong đó, các đặc trưng của dữ liệu gồm: AcceptsCreditCard,

PriceRange, WiFi là các thuộc tính tiện ích của cơ sở kinh doanh khách sạn được trích xuất từ dữ liệu cấu trúc sẵn có (tại bảng hotel), không phải khía cạnh được trích xuất từ nội dung văn bản đánh giá hay được người dùng chấm điểm độc lập theo đúng phát biểu bài toán ABSA (Aspect-Based Sentiment Analysis). Về mối quan hệ với phát biểu bài toán của luận án: theo cơ sở lựa chọn hướng tiếp cận học quần thể đã trình bày tại §1.5.4–1.5.5, luận án xác định phạm vi bài toán là khai phá quan điểm trên dữ liệu bộ năm quan điểm o = (e, a, s, h, t) đã được cấu trúc hóa dạng bảng (tabular), phù hợp làm đầu vào cho các mô hình rừng học quần thể có khả năng diễn giải cao, không đặt mục tiêu giải bài toán ABSA ở mức trích xuất khía cạnh trực tiếp từ văn bản tự do trên mọi tập dữ liệu thực nghiệm. Nhiệm vụ trích xuất khía cạnh từ văn bản theo đúng phát biểu ABSA được thực hiện trên tập Amazon Food Reviews; tập TripAdvisor Hotel Reviews cung cấp xếp hạng độc lập theo từng khía cạnh do người dùng tự chấm và hai tập dữ liệu này minh chứng khả năng của mô hình đề xuất trên dữ liệu có tín hiệu khía cạnh thực sự độc lập.

Đối với yelpHotelData, do hai tập này không cung cấp sẵn khía cạnh trích xuất từ văn bản, luận án chủ động thay thế bằng các thuộc tính tiện ích cơ sở kinh doanh như một dạng đặc trưng khía cạnh cấu trúc vẫn thỏa mãn yêu cầu đầu vào dạng bảng của mô hình rừng, nhưng không mang tín hiệu tình cảm độc lập theo từng khía cạnh trong nội bộ một đánh giá, do mọi dòng bộ năm sinh ra từ cùng một đánh giá đều kế thừa chung một điểm sao tổng thể. Đây là một giới hạn được nhận thức rõ của cách tổ chức dữ liệu trên hai tập Yelp khi đối chiếu với phát biểu bài toán ABSA ở mức khía cạnh chi tiết hơn. Tuy nhiên, trong phạm vi Chương 3, vai trò chính của thành phần khía cạnh trên hai tập Yelp là bổ sung một chiều đặc trưng cấu trúc để mô hình rừng mật độ (phân cụm DBSCAN) và rừng đa tạp (ma trận đồng xuất hiện) khai thác cấu trúc không đồng đều của không gian đặc trưng, không nhất thiết phải mang tín hiệu phân biệt tình cảm độc lập để có ích cho các mô hình này.

Bảng 3.1: Thống kê các đặc trưng trong bộ dữ liệu yelpHotelData

Các đặc trưng của đánh giá khách sạn	**Các giá trị khác nhau của đặc trưng**
Tên khách sạn	từ 1 đến 123.461

Khía cạnh khách sạn	từ 1 đến 3
Người đánh giá khách sạn	từ 1 đến 4.596
Thời gian đánh giá	từ 1 đến 4.382
Xếp hạng quan điểm	từ 1 đến 5

3.7.2.2. Tiền xử lý dữ liệu yelpResData

Tập dữ liệu ban đầu *yelpResData* có các đặc trưng quan trọng được chọn bao gồm tên nhà hàng, khía cạnh của nhà hàng, người đánh giá nhà hàng, thời gian đánh giá nhà hàng và quan điểm đánh giá nhà hàng. Về khía cạnh của nhà hàng, đã khám phá 19 khía cạnh ảnh hưởng đến nhà hàng như các tiêu chí về phù hợp với trẻ em (GoodforKids), chấp nhận thẻ tín dụng (AcceptsCreditCards), có chỗ đậu xe (Parking), trang phục của nhân viên (Attire), sự phù hợp với nhóm người (GoodforGroups), mức giá (PriceRange), nhận đặt chỗ (TakesReservations), vận chuyển (Delivery), mang đi (Takeout), dịch vụ bồi bàn (WaiterService), ngồi ngoài trời (OutdoorSeating), WiFi, tốt cho trẻ (GoodFor), rượu (Alcohol), độ ồn (NoiseLevel), có tivi (TV), môi trường xung quanh (Ambience), dịch vụ ăn uống (Caters) và dịch vụ xe lăn cho người khuyết tật (WheelchairAccessible). Kết quả chuyển đổi từ dữ liệu gốc ban đầu có 788.471 bản ghi thành cấu trúc dữ liệu bộ năm quan điểm có 14.791.500 mẫu, chi tiết các đặc trưng được liệt kê trong Bảng 3.2.

Tương tự như trường hợp yelpHotelData (§3.7.2.1), 19 khía cạnh nêu trên là các thuộc tính tiện ích của nhà hàng chứ chưa phải khía cạnh được đề cập trực tiếp trong nội dung đánh giá; nhãn tình cảm của mọi khía cạnh sinh ra từ cùng một đánh giá đều lấy theo điểm sao tổng thể của đánh giá đó. Đây cũng là lý do số mẫu bộ năm tăng mạnh so với số bản ghi đánh giá gốc: mỗi đánh giá gốc được nhân bản thành tối đa 19 dòng (một dòng cho mỗi khía cạnh), nên số mẫu bộ năm báo cáo tại Phụ lục C (~14.791.500) lớn hơn nhiều lần số bản ghi đánh giá gốc thống kê tại Phụ lục A; tương tự, số mẫu bộ năm của yelpHotelData (882.474) tương ứng với số bản ghi đánh giá gốc nhân với tối đa 3 khía cạnh.

Bảng 3.2: Thống kê các đặc trưng trong bộ dữ liệu yelpResData

Các đặc trưng của đánh giá nhà hàng	Các giá trị khác nhau của đặc trưng
Tên nhà hàng	từ 1 đến 184.167
Khía cạnh nhà hàng	từ 1 đến 19
Người đánh giá nhà hàng	từ 1 đến 12.943
Thời gian đánh giá	từ 1 đến 4.541
Xếp hạng quan điểm (5★)	từ 1★ đến 5★

Quy trình chuyển đổi 5 mức sao (★) sang 2 nhãn nhị phân được thực hiện theo quy tắc sau: các đánh giá 1★ và 2★ (tương ứng "very negative" và "negative") được gán nhãn negative; các đánh giá 4★ và 5★ (tương ứng "positive" và "very positive") được gán nhãn positive. Các đánh giá trung lập 3★ ("neutral") bị loại bỏ khỏi tập dữ liệu vì không mang thông tin tình cảm rõ ràng theo hướng nào và việc phân loại sẽ làm tăng nhiễu trong quá trình huấn luyện. Quy tắc này được áp dụng cho tập Amazon Food Reviews và TripAdvisor Hotel Reviews; riêng hai tập yelpHotelData và yelpResData có điều chỉnh ngưỡng nhị phân, chi tiết chuyển đổi và giải thích tại Bảng 3.3.

Bảng 3.3: Chuyển đổi 5 mức sao sang 2 nhãn nhị phân

Mức đánh giá (★)	Nhãn 5 cấp độ	Nhãn nhị phân	Lý do
1★	very negative	negative	Tình cảm tiêu cực rõ ràng
2★	negative	negative	Tình cảm tiêu cực rõ ràng
3★	neutral	Loại bỏ (Amazon, TripAdvisor); negative (yelpHotelData, yelpResData)	Amazon, TripAdvisor: không mang thông tin tình cảm rõ ràng; Yelp: gộp vào negative để giữ trọn dữ liệu (xem đoạn dưới)

4★	positive	positive	Tình cảm tích cực rõ ràng
5★	very positive	positive	Tình cảm tích cực rõ ràng

Riêng đối với hai tập yelpHotelData và yelpResData (Chương 3, Chương 4), luận án áp dụng ngưỡng nhị phân bất đối xứng: "positive" ứng với đánh giá 4★–5★, "negative" ứng với đánh giá 1★–3★ (gộp cả mức trung lập 3★), thay vì loại bỏ 3★ như quy tắc áp dụng cho Amazon Food Reviews và TripAdvisor Hotel Reviews. Sự khác biệt này xuất phát từ đặc thù cấu trúc dữ liệu bộ năm của hai tập Yelp: "khía cạnh" ở đây là các thuộc tính tiện ích của cơ sở kinh doanh (ví dụ AcceptsCreditCard, PriceRange, WiFi), không phải khía cạnh được trích xuất từ nội dung đánh giá hay được người dùng chấm điểm riêng như ở Amazon/TripAdvisor; mọi dòng bộ năm sinh ra từ cùng một đánh giá đều kế thừa chung đúng một điểm sao tổng thể (hạn chế đã nêu tại §3.7.2.1/§3.7.2.2). Do sentiment ở mức khía cạnh không thực sự độc lập, việc loại bỏ trực tiếp các dòng 3★ sẽ làm mất cân đối số khía cạnh quan sát được giữa các thực thể, ảnh hưởng đến chất lượng phân cụm DBSCAN/K-means ở các bước sau; vì vậy luận án giữ nguyên toàn bộ dữ liệu và gộp 3★ vào nhóm negative. Lựa chọn này cũng phù hợp với đặc điểm thiên lệch dương (positive skew) phổ biến trên các nền tảng đánh giá dịch vụ trực tuyến, trong đó cảm xúc tiêu cực của văn bản đánh giá thường có ảnh hưởng mạnh hơn đến điểm sao so với cảm xúc tích cực cùng cường độ, khiến đánh giá 3★ thường phản ánh sự pha trộn hoặc nghiêng nhẹ về phía chưa hài lòng hơn là trung lập thực sự [205], [206]. Do quy tắc này gắn với đặc thù dữ liệu (không phụ thuộc vào mô hình), Chương 3 (rừng mật độ) và Chương 4 (rừng đa tạp) — dù đề xuất hai mô hình khác nhau — đều áp dụng thống nhất cùng một ngưỡng trên hai tập Yelp.

Sau khi áp dụng ngưỡng nhị phân bất đối xứng nêu trên, tập dữ liệu còn lại có tỷ lệ mất cân bằng khoảng 1,7:1 đến 1,9:1 (positive:negative). Bootstrap trong mô hình rừng mật độ được áp dụng kết hợp với chiến lược lấy mẫu phân tầng (stratified sampling) để đảm bảo mỗi cây trong rừng nhận được tỷ lệ nhãn đại diện, tránh tình trạng cây chỉ học từ nhãn đa số.

Xử lý mất cân bằng nhãn: Sau khi nhị phân hóa theo ngưỡng bất đối xứng nêu trên (4★–5★ → positive; 1★–3★ → negative), phân phối nhãn trên hai tập Yelp nghiêng về phía tích cực với tỷ lệ khoảng 1,7:1 đối với yelpHotelData (560.769 positive / 321.705 negative, positive chiếm khoảng 63,5%) và 1,9:1 đối với yelpResData (9.650.537 / 5.140.963, positive chiếm khoảng 65,2%). Trong mô hình rừng mật độ, vấn đề này được xử lý theo ba hướng: (1) Phân cụm DBSCAN phân vùng không gian đặc trưng, các vùng mật độ thưa thường chứa các mẫu tiêu cực (thiểu số), giúp rừng chuyên biệt theo cụm học tốt hơn, theo chiến lược kết hợp phân cụm mật độ với rừng ngẫu nhiên đã được chứng minh hiệu quả trên dữ liệu mất cân bằng [70]; (2) Lấy mẫu phân tầng (stratified sampling) khi chia dữ liệu đảm bảo tỷ lệ nhãn nhất quán giữa các tập; (3) Các kỹ thuật cân bằng lớp tại tầng rừng cơ sở (trọng số lớp, Bootstrap cân bằng lớp, SMOTE, hiệu chỉnh ngưỡng quyết định) được so sánh trực tiếp tại Phụ lục G.

3.7.3. Các kịch bản thực nghiệm

3.7.3.1. Mô hình khai phá quan điểm với rừng mật độ dựa trên bộ năm quan điểm trên tập dữ liệu khách sạn yelpHotelData

Kịch bản 1. Rừng mật độ quan điểm dựa trên bộ năm quan điểm trên tập dữ liệu khách sạn yelpHotelData

Từ bộ dữ liệu đánh giá khách sạn *yelpHotelData*, phân cụm DBSCAN trên không gian vectơ đặc trưng đa chiều được thực hiện. Kết quả của mỗi phân cụm mật độ được sử dụng để xây dựng rừng phân loại để đưa ra quan điểm về các đặc trưng của khách sạn như tên khách sạn, khía cạnh của khách sạn, người đánh giá khách sạn, thời gian đánh giá. Các khu rừng mật độ được xây dựng bằng cách chia dữ liệu trong một cụm thành *n* mẫu bootstrap. Mỗi cây trong rừng được xây dựng trên một tập hợp con các đặc trưng ngẫu nhiên bao gồm tên khách sạn, khía cạnh của khách sạn, người đánh giá khách sạn, thời gian đánh giá. Đặc trưng quyết định có giá trị là "positive" hay "negative". Các khu rừng ngẫu nhiên được xây dựng trên các phân cụm mật độ và kết hợp thành một bộ phân loại duy nhất được gọi là rừng mật độ để dự đoán kết quả phân loại quan điểm là "positive" hay "negative". Cụ thể, sau khi tiền xử lý bộ

dữ liệu đánh giá khách sạn đã thu được 882.474 bộ năm quan điểm. Thử nghiệm được thực hiện trên 112.275 mẫu.

Thiết lập tham số DBSCAN. Hai tham số cốt lõi của DBSCAN được lựa chọn thông qua thực nghiệm trên tập xác thực, cụ thể:

Bảng 3.4: Thiết lập tham số mô hình Rừng mật độ quan điểm trong thực nghiệm

Tham số	**Ý nghĩa**	**Giá trị thử nghiệm**	**Giá trị được chọn**	**Cơ sở lựa chọn**
ε (eps)	Bán kính vùng lân cận: điểm o' là lân cận của *o* nếu dist(o, o') ≤ ε	0,1; 0,3; 0,5; 0,7; 1,0	0,5	Căn cứ vào biểu đồ k-distance; ε = 0,5 tạo ra số cụm K phù hợp với phân phối dữ liệu
MinPts	Số điểm tối thiểu trong vùng lân cận để hình thành một cụm lõi	3, 5, 7, 10	5	Quy tắc kinh nghiệm MinPts ≥ d+1 với d=4 (số chiều đặc trưng đầu vào e, a, h, t sau khi tách nhãn tình cảm s); MinPts=5=d+1 cho kết quả ổn định

Quá trình huấn luyện được thực hiện như sau:

Bước 1: Chia dữ liệu thành 2 tập dữ liệu là tập dữ liệu huấn luyện (80%), tập dữ liệu kiểm tra (20%).

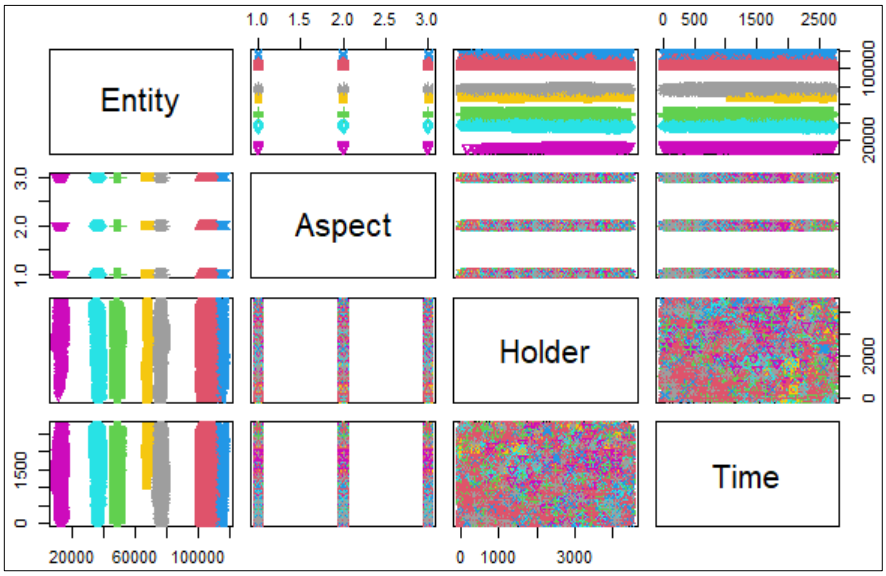
Bước 2: Phân cụm dữ liệu trên tập dữ liệu huấn luyện không dán nhãn bằng thuật toán phân cụm DBSCAN với các tham số được thiết lập ở Bảng 3.4. Kết quả được 07 phân cụm mật độ trong Hình 3.3.

Bước 3: Lấy 07 phân cụm để huấn luyện các thuật toán rừng ngẫu nhiên, mỗi phân cụm huấn luyện một thuật toán rừng ngẫu nhiên.

Bước 4: Tổng hợp kết quả của từng thuật toán rừng ngẫu nhiên để cho kết quả dự đoán quan điểm cuối cùng về khách sạn là “positive” hay “negative”.

Hình 3.3 hiển thị biểu đồ phân tán của bốn đặc trưng bao gồm Entity, Aspect, Holder, Time. Trong đó, Aspect có sự phân tách rõ ràng với ba danh mục, Entity và Holder có số lượng cao với phân phối bị lệch và Time có độ lan truyền theo thời gian ổn định. Các biểu đồ phân tán cho thấy không có cụm tự nhiên rõ ràng, phù hợp với

dữ liệu quan điểm dựa trên khía cạnh và chứng minh việc phân cụm dựa trên mật độ là hợp lý.



Hình 3.3: Kết quả phân cụm quan điểm trên tập dữ liệu yelpHotelData

Lưu ý về mối quan hệ giữa Thuật toán 3.1 và Kịch bản 1 (thực nghiệm ở trên): Thuật toán 3.1 (BuildDensityForest/BuildDensityTree, mục 3.4.1) trình bày khung tổng quát của mô hình rừng mật độ quan điểm, trong đó phép phân cụm DBSCAN và ước lượng mật độ hỗn hợp Gaussian (FitGaussianMixture) chỉ được thực hiện tại các nút lá, sau khi cây đã được phân tách qua nhiều tầng theo tiêu chí Entropy mật độ ở các công thức (3.11) và (3.12), với điều kiện dừng đệ quy khi depth = max_depth hoặc |O| < min_split. Kịch bản 1 trình bày ở trên, để đảm bảo khả năng thực thi trên tập dữ liệu lớn và tận dụng nhãn cực tính sẵn có trong tập huấn luyện, sử dụng một biến thể đơn tầng của Thuật toán 3.1 (tương ứng max_depth = 0, đồng thời thay ước lượng mật độ tại nút lá bằng rừng phân loại có giám sát như trình bày dưới đây): điều kiện dừng ở dòng 1 của hàm BuildDensityTree được thỏa mãn ngay tại nút gốc, khiến nút gốc đồng thời là nút lá duy nhất của cây, do đó DBSCAN (Bước 2 của thực nghiệm ở trên) chỉ cần thực hiện một lần trên toàn bộ tập huấn luyện thay vì đệ quy qua nhiều tầng phân tách. Tại mỗi cụm mật độ thu được, thay vì ước lượng hàm mật độ Gaussian không giám sát pleaf, luận án huấn luyện một mô hình rừng phân loại quan điểm (Chương 2) có giám sát ngay trên dữ liệu của cụm đó để ước lượng trực tiếp xác suất phân cực p(y|o) (Bước 3 của thực nghiệm ở trên), qua đó khai thác nhãn cực tính đã có thay vì bỏ qua như trong ước lượng mật độ thuần túy. Xác suất dự đoán

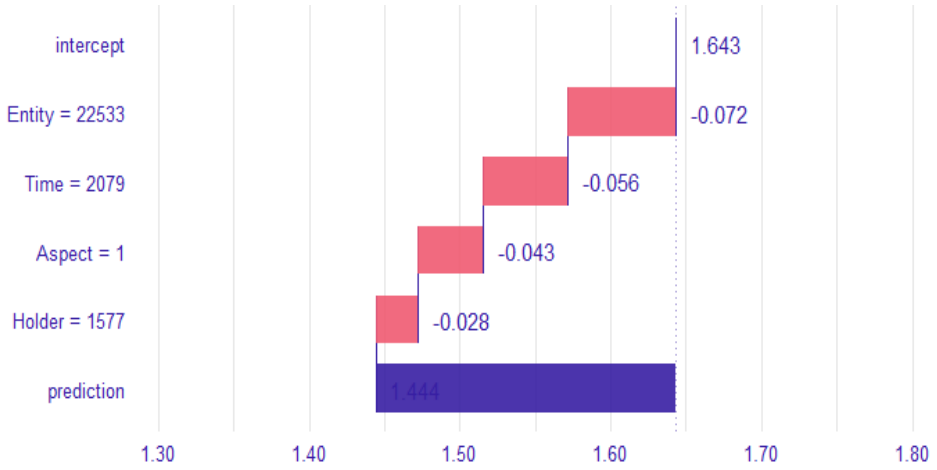
cuối cùng vẫn được tổng hợp bằng phép trung bình cộng giữa các cụm/rừng thành phần (Bước 4 của thực nghiệm ở trên), nhất quán với công thức (3.18) của rừng mật độ quan điểm tổng quát. Cách triển khai đơn tầng này (Kịch bản 1) được lựa chọn cho thực nghiệm nhằm cân bằng giữa chi phí tính toán và độ chính xác; dạng tổng quát của Thuật toán 3.1 với max_depth > 0 và FitGaussianMixture đầy đủ được giữ nguyên như khung lý thuyết, mở ra hướng thực nghiệm sâu hơn (phân tách đệ quy nhiều tầng, ước lượng mật độ hỗn hợp Gaussian) cho các nghiên cứu tiếp theo. Kịch bản 1 trên tập dữ liệu nhà hàng yelpResData (mục 3.7.3.2) được xây dựng theo đúng nguyên tắc tương tự.

Kịch bản 2. Diễn giải mô hình rừng mật độ quan điểm dựa trên bộ năm quan điểm trên tập dữ liệu khách sạn YelpHotelData

Mô hình đề xuất được huấn luyện trên tập dữ liệu *YelpHotelData* đã được xử lý trước thành các bộ năm thành phần quan điểm và được chuyển đổi các bộ năm thành phần quan điểm thành các vectơ trong không gian vectơ đặc trưng để làm đầu vào cho việc huấn luyện mô hình. Kết quả thử nghiệm giá trị SHAP có thể được sử dụng để giải thích sự đóng góp của các đặc trưng vào dự đoán cho một quan sát duy nhất, các giá trị SHAP được tính toán dựa trên các giá trị trong tập dữ liệu để thu được giá trị tầm quan trọng của các đặc trưng.

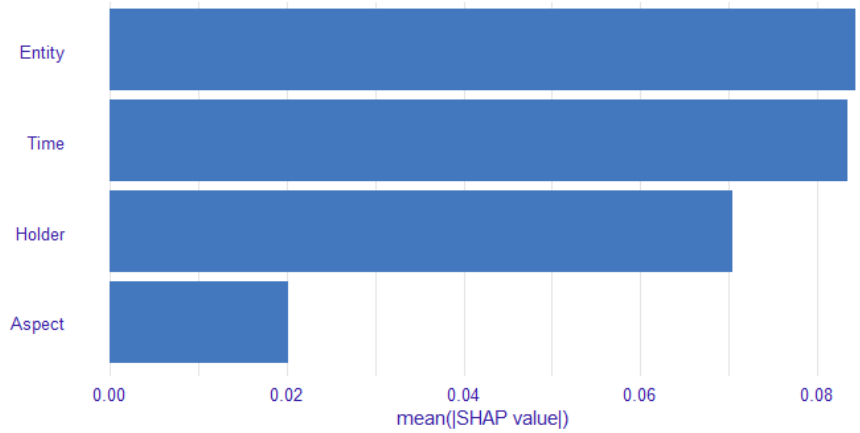
Hình 3.4 minh họa diễn giải cục bộ, biểu đồ SHAP cho một quan sát đại diện, thể hiện quá trình hình thành dự đoán từ giá trị 1,643 thông qua đóng góp tuần tự của các đặc trưng. Đặc trưng Entity có ảnh hưởng tiêu cực mạnh nhất (-0,072), tiếp theo là Time (-0,056), Aspect (-0,043), và Holder (-0,028).

Các đặc trưng đều kéo dự đoán về phía tiêu cực với tổng ảnh hưởng là -0,199 (-(0,072 + 0,056 + 0,043 + 0,028)). Các thanh màu đỏ thể hiện đóng góp tiêu cực, trong khi thanh màu xanh cuối cùng biểu diễn giá trị dự đoán tổng hợp. Phân tích này cho thấy mô hình cân bằng giữa xu hướng tổng và bằng chứng cụ thể từ các đặc trưng đều tiêu cực, minh chứng khả năng diễn giải chi tiết của SHAP ở cấp độ một trường hợp cụ thể.



Hình 3.4: Sự đóng góp của các đặc trưng

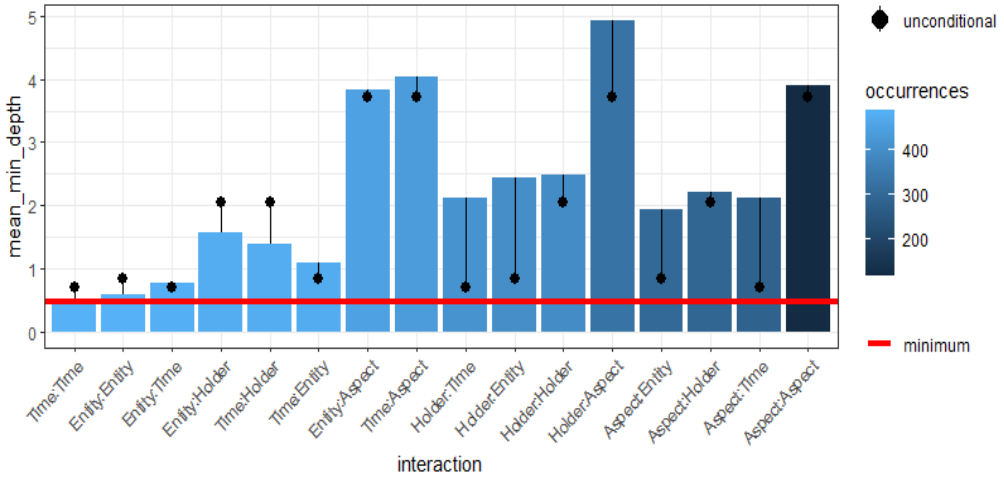
Hình 3.5 minh họa đặc trưng quan trọng toàn cục thông qua trung bình giá trị SHAP (mean|SHAP value|) trên toàn bộ tập dữ liệu. Entity và Time có tầm quan trọng cao nhất và tương đương nhau (≈ 0,087 và 0,085), chiếm vị trí dẫn đầu trong việc ảnh hưởng đến dự đoán của mô hình. Holder đứng thứ ba với giá trị khoảng 0,077, trong khi Aspect có tầm quan trọng thấp nhất (≈ 0,023), chỉ bằng khoảng một phần tư so với Entity và Time. Sự khác biệt lớn về tầm quan trọng bởi vì số lượng giá trị thấp của Aspect (3 giá trị) theo Bảng 3.1, dẫn đến khả năng phân biệt thấp hơn so với Entity (123.461 giá trị khác nhau) và Time có phạm vi thời gian rộng. Kết quả này phù hợp với kết quả từ Hình 3.4, Entity cũng thể hiện ảnh hưởng mạnh nhất.



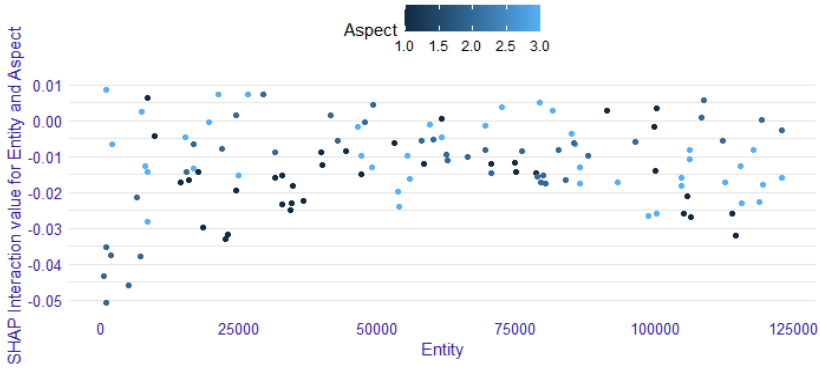
Hình 3.5: Phân bổ tầm quan trọng của đặc trưng trên dữ liệu YelpHotelData

Hình 3.6 minh họa phân tích tương tác SHAP thông qua độ sâu tối thiểu trung bình của các cặp tương tác đặc trưng trên mô hình rừng mật độ. Biểu đồ cho thấy sự khác biệt rõ rệt giữa ba nhóm tương tác. Nhóm thứ nhất bao gồm các tương tác với chính nó và Entity-Time có độ sâu rất thấp từ 0,5-0,7, cho thấy chúng xuất hiện ngay

từ các nút gần gốc của cây, phản ánh vai trò nền tảng trong việc phân chia không gian dữ liệu. Nhóm thứ hai gồm các tương tác với Holder có độ sâu trung bình từ 1,4-2,4, thể hiện tầm quan trọng ở mức độ trung gian. Nhóm thứ ba với các tương tác liên quan đến Aspect có độ sâu rất cao từ 3,9-5,0, đặc biệt Holder-Aspect đạt giá trị cao nhất là 5,0, cho thấy Aspect chỉ được xem xét ở các mức sâu của cây. Mẫu này có thể giải thích bởi số lượng thấp của Aspect có 3 giá trị so với Entity là 123.461 và Time ở phạm vi rộng, dẫn đến khả năng phân biệt hạn chế và do đó được mô hình sử dụng chủ yếu cho tinh chỉnh ở các nhánh sâu thay vì chia tách chính.



Hình 3.6: Các cặp tương tác thường xuyên

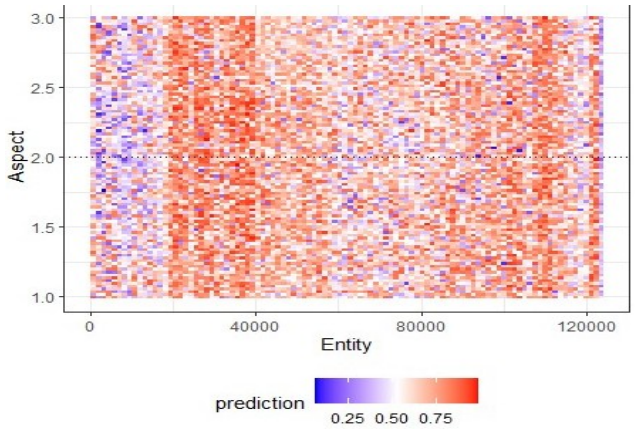


Hình 3.7: Giá trị tương tác SHAP cho các đặc trưng trên dữ liệu YelpHotelData

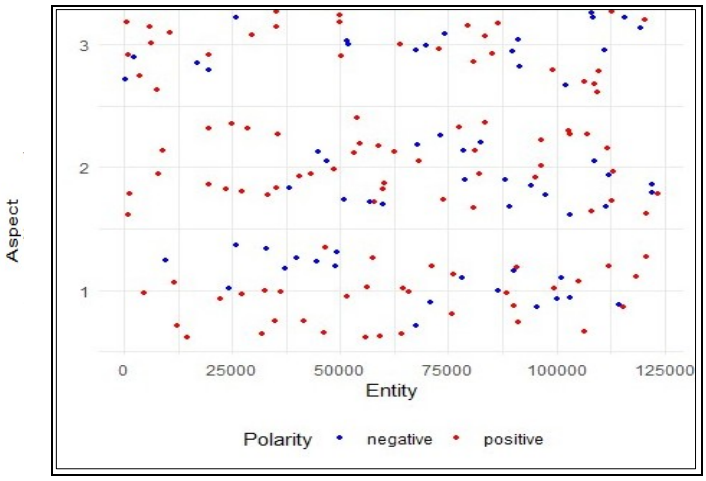
Hình 3.7 minh họa giá trị tương tác SHAP giữa Entity và Aspect, cho thấy xu hướng tiêu cực mạnh với khoảng 90% điểm có giá trị âm tập trung trong khoảng [-0,02; -0,01], chứng tỏ sự kết hợp này kéo dự đoán về phía tình cảm tiêu cực so với ảnh hưởng độc lập của từng đặc trưng. Các entity với ID thấp (< 25.000) thể hiện sự thay đổi cao với giá trị phân tán rộng từ -0,05 đến +0,01, cho thấy đặc điểm đặc biệt

được tìm thấy tạo ra các tương tác không điển hình. Ngược lại, entity với ID 25.000–100.000 tập trung trong vùng hẹp hơn, phản ánh các mẫu hình ổn định hơn.

Hình 3.8 và Hình 3.9 minh họa tương ứng biểu đồ nhiệt dự đoán và phân bố thực tế trên không gian Entity-Aspect. Biểu đồ nhiệt dự đoán (Hình 3.8) cho thấy khoảng 60-70% diện tích màu đỏ cam (dự đoán > 0,75), với Aspect là 2,0 thể hiện màu đậm nhất và vùng Entity 20.000–40.000 tạo thành vùng đỏ đồng nhất. Biểu đồ phân tán thực tế (Hình 3.9) cho thấy hai lớp negative (xanh) và positive (đỏ) đan xen nhau (lớp positive chiếm khoảng 63,5% mẫu), với trộn tốt trên phạm vi thực thể và Aspect từ 2-3 có nhiều mẫu positive hơn. Sự tương đồng về các mẫu giữa hai hình cho thấy mô hình đã nắm bắt được cấu trúc cơ bản của phân bố dữ liệu phù hợp với kiến trúc các cây chuyên biệt của rừng mật độ.



Hình 3.8: Phân phối xác suất dự đoán trong tập huấn luyện



Hình 3.9: Phân phối xác suất dự đoán trong tập kiểm tra YelpHotelData

Bảng 3.5 trình bày kết quả đánh giá hiệu suất của mô hình rừng mật độ ở giai đoạn cuối đó là phân loại tình cảm. Trên tập huấn luyện, mô hình đạt Accuracy 0,84, Precision 0,80, Recall 0,70 và F1 0,75, cho thấy khả năng phân loại tốt với độ chính xác cao. Trên tập kiểm tra, các độ đo tương ứng là 0,75; 0,71; 0,62 và 0,67. Sự chênh lệch nằm trong ngưỡng chấp nhận được. Nguyên nhân có thể do tập kiểm tra chứa các tổ hợp Entity, Holder, Time không có trong tập huấn luyện; độ phức tạp của mô hình cao cho phép ghi nhớ các mẫu huấn luyện; mất cân bằng lớp (lớp positive chiếm 63,5%) dẫn đến sai lệch về lớp đa số; số lượng giá trị lớn của Entity (123.461 giá trị khác nhau, Bảng 3.1) tạo ra nhiều kết hợp chỉ xuất hiện trong tập huấn luyện. Tuy vậy, không có độ đo nào trên tập kiểm tra nhỏ hơn 0,6. Lưu ý rằng tập huấn luyện và tập kiểm tra ở đây được chia ngẫu nhiên theo dòng bộ năm như các bài báo đã công bố, nên các dòng của cùng một đánh giá có thể nằm ở cả hai tập và chỉ số trên tập kiểm tra vẫn lạc quan hơn so với khi chia theo đánh giá gốc (Phụ lục J, mục J.11).

Bảng 3.5: Kết quả phân loại tình cảm của rừng mật độ trên YelpHotelData

Tập dữ liệu khách sạn	Độ chính xác	Độ chính xác dương (P)	Độ bao phủ (R)	Trung bình điều hòa F1
Tập huấn luyện	0,84	0,80	0,70	0,75
Tập kiểm tra	0,75	0,71	0,62	0,67

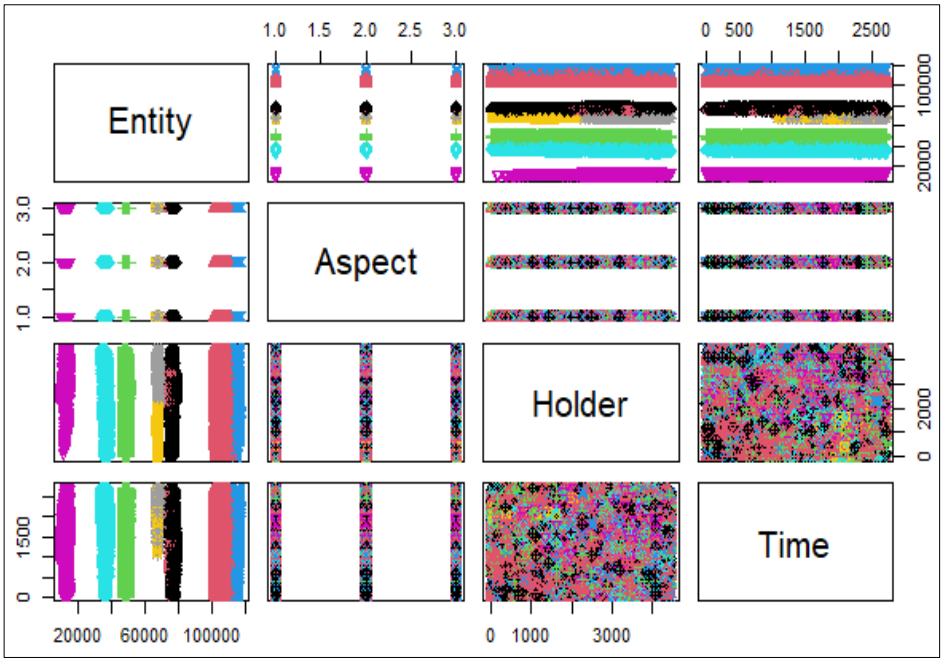
3.7.3.2. Mô hình khai phá quan điểm với rừng mật độ trên tập dữ liệu nhà hàng (yelpResData)

Kịch bản 1. Rừng mật độ quan điểm trên tập dữ liệu nhà hàng (yelpResData)

Tập dữ liệu đánh giá nhà hàng *yelpResData*, sau khi tiền xử lý có 14.791.500 mẫu bộ năm quan điểm, nhiều hơn so với tập dữ liệu đánh giá khách sạn *YelpHotelData*. Thử nghiệm đã được thực hiện trên 112.275 bộ năm quan điểm. Quá trình huấn luyện được thực hiện như sau:

Bước 1: Chia dữ liệu ban đầu thành 2 tập dữ liệu bao gồm tập dữ liệu huấn luyện (80%) và tập dữ liệu kiểm tra (20%).

Bước 2: Phân cụm bằng thuật toán DBSCAN trên tập dữ liệu huấn luyện không dán nhãn. Kết quả có 09 phân cụm mật độ được hình thành từ thuật toán phân cụm DBSCAN trong Hình 3.10.



Hình 3.10: Kết quả phân cụm DBSCAN quan điểm trên tập dữ liệu yelpResData

Bước 3: Lấy 09 phân cụm để huấn luyện các thuật toán rừng ngẫu nhiên, mỗi phân cụm huấn luyện một thuật toán rừng ngẫu nhiên.

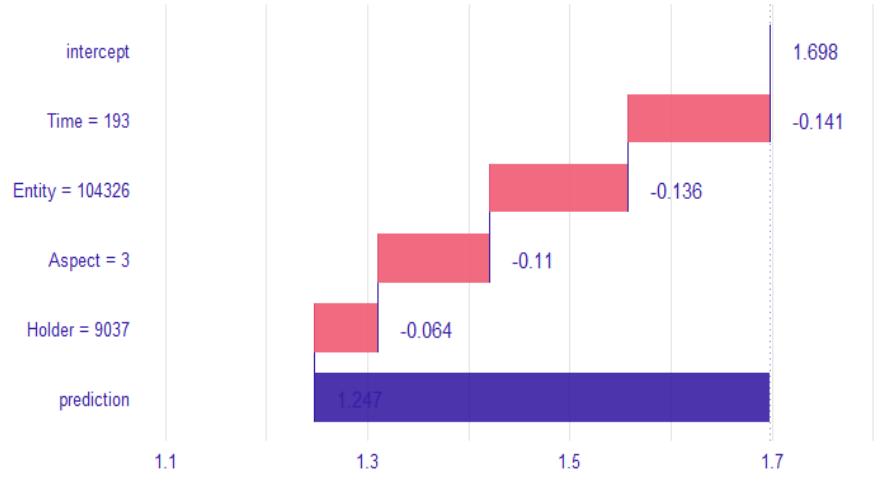
Bước 4: Tổng hợp kết quả dự đoán từng mô hình rừng ngẫu nhiên để cho ra kết quả dự đoán quan điểm về nhà hàng là "positive" hoặc "negative".

Kịch bản 2. Diễn giải mô hình rừng mật độ quan điểm trên tập dữ liệu nhà hàng (yelpResData)

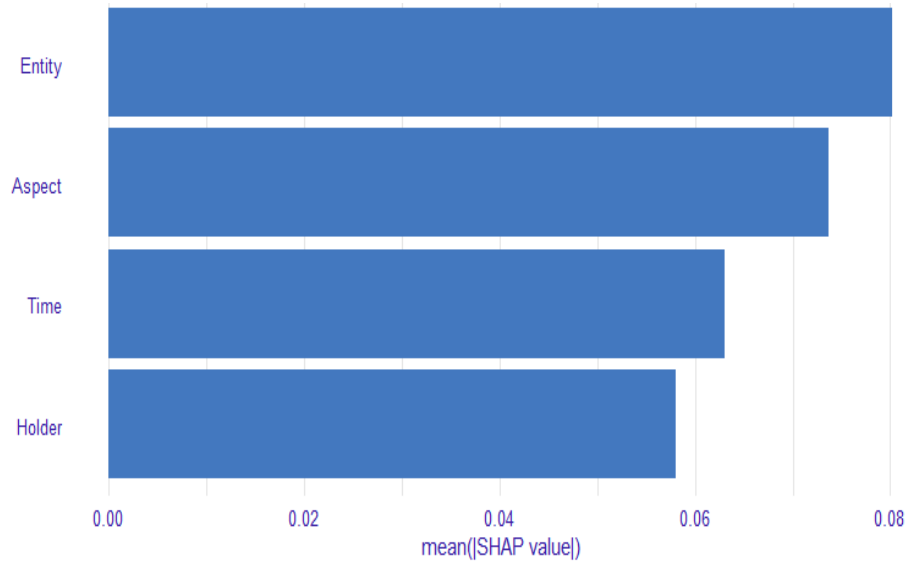
Mô hình đề xuất rừng mật độ quan điểm được huấn luyện trên bộ dữ liệu *YelpResData*. Kết quả thử nghiệm cho thấy giá trị SHAP có thể được sử dụng để giải thích sự đóng góp của các đặc trưng vào dự đoán cho một quan sát trong Hình 3.11. Entity là đặc trưng quan trọng nhất, trong khi Holder là đặc trưng ít quan trọng nhất cho đóng góp dự đoán. Hình 3.11 cho thấy một dự đoán riêng lẻ có giá trị cơ sở 1,698, trong đó tất cả bốn đặc trưng đều có đóng góp tiêu cực. Time (= - 0,141) và Entity (= - 0,136) có tác động mạnh nhất, theo sau là Aspect (= -0,11) và Holder (= - 0,064), dẫn đến dự đoán cuối cùng là 1,247.

Phân tích tầm quan trọng của đặc trưng trong Hình 3.12 cho thấy Entity có tầm quan trọng cao nhất, trung bình giá trị SHAP là 0,078, theo sau là Aspect (0,075) thấp hơn 3,8%, Time (0,065) và Holder (0,058). Mức chênh lệch tương đối nhỏ giữa các đặc trưng cho thấy tất cả các đặc trưng đều đóng góp đáng kể vào các dự đoán. Entity

và Aspect đóng vai trò quan trọng nhất trong xác định tình cảm của các đánh giá về nhà hàng.

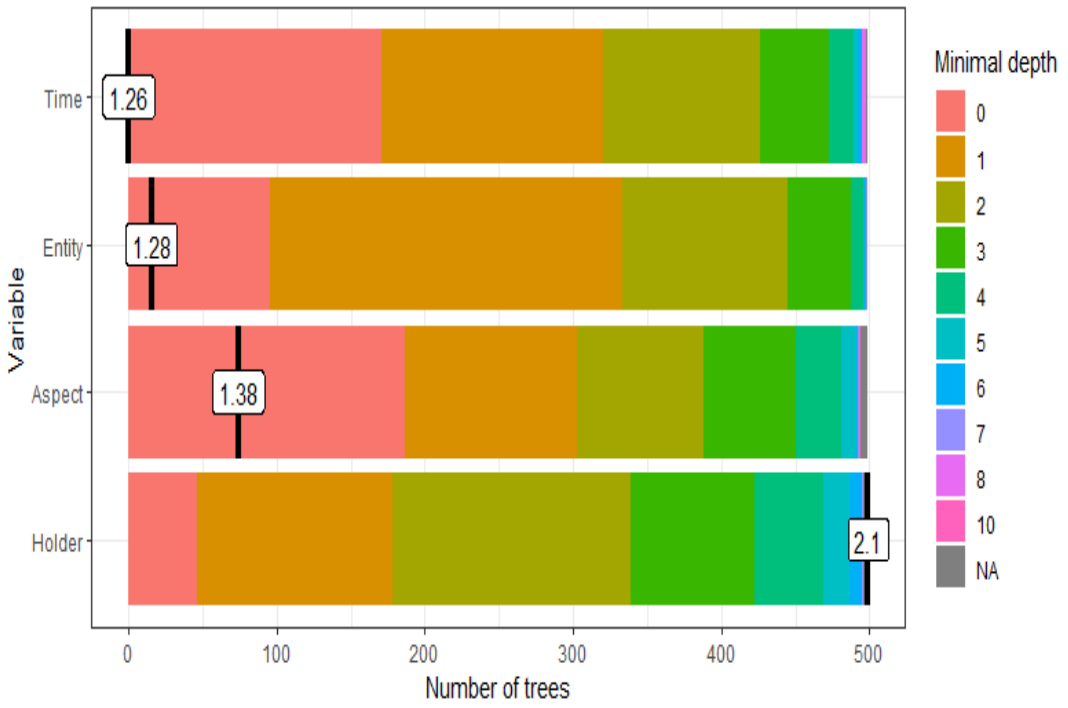


Hình 3.11: Sự đóng góp của các đặc trưng trên dữ liệu YelpResData



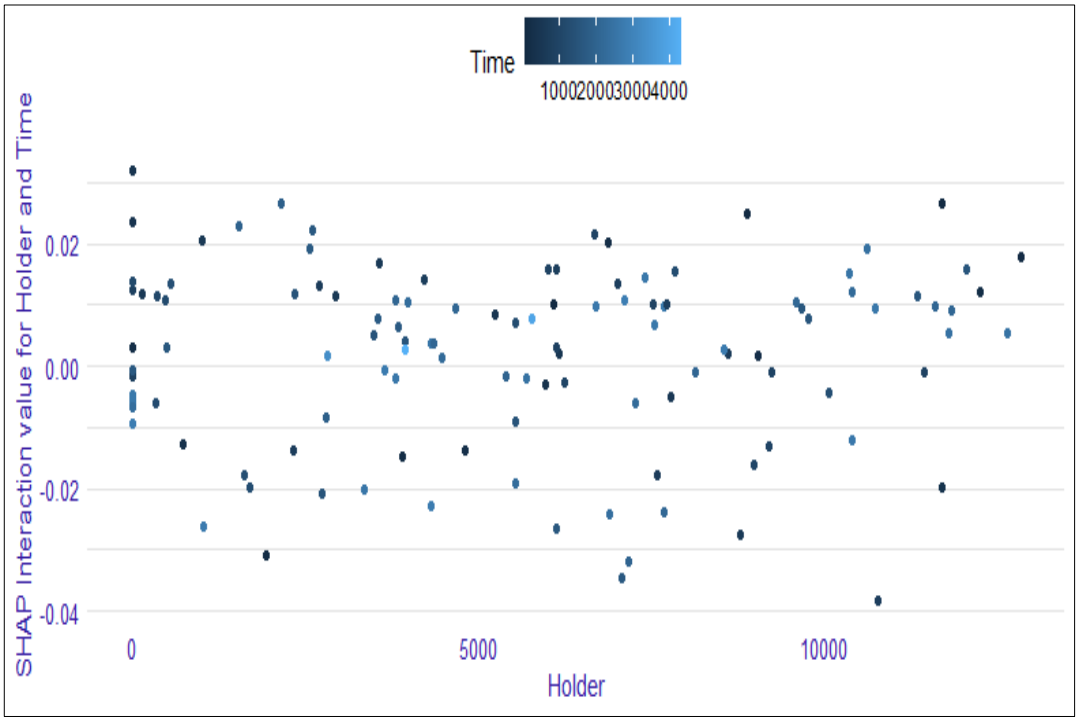
Hình 3.12: Tầm quan trọng của các đặc trưng trên dữ liệu YelpResData

Phân phối độ sâu tối thiểu trong Hình 3.13 cho thấy các mẫu cấu trúc cây với Time (1,26) và Entity (1,28) được phân chia sớm nhất, cho thấy chúng quan trọng đối với việc phân vùng cây ban đầu. Aspect (1,38) được phân tách muộn hơn một chút, trong khi Holder (2,1) có độ sâu trung bình sâu nhất, cho thấy nó được sử dụng chủ yếu cho các sàng lọc chi tiết hơn là các phân tách chính.

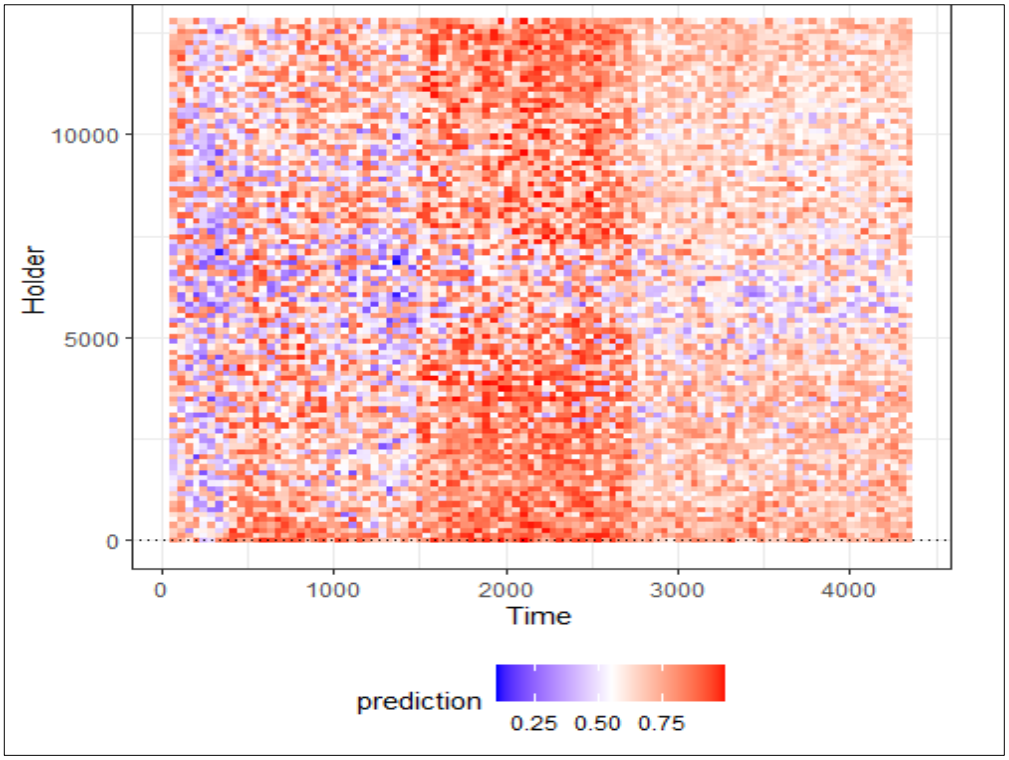


Hình 3.13: Phân phối độ sâu và đặc trưng tối thiểu trên dữ liệu YelpResData

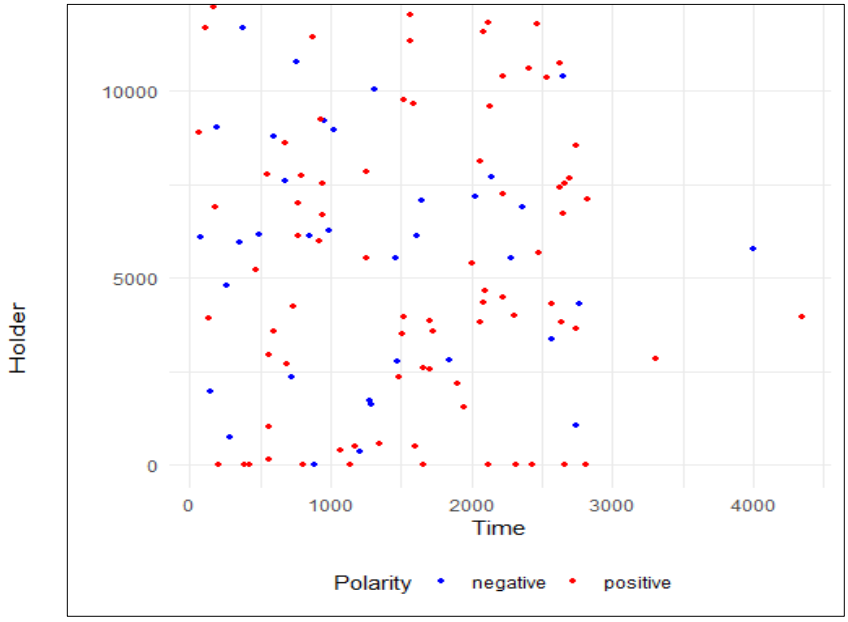
Biểu đồ tương tác SHAP trong Hình 3.14 cho các đặc trưng Holder và Time cho thấy các tương tác yếu đến trung bình với các điểm phân tán rộng rãi, gợi ý các hiệu ứng tương tác phức tạp nhưng không vượt trội.



Hình 3.14: Giá trị tương tác SHAP trên dữ liệu YelpResData



Hình 3.15: Phân phối xác suất dự đoán trong tập huấn luyện trên YelpResData



Hình 3.16: Phân phối xác suất dự đoán trong tập kiểm tra trên YelpResData

Trong Hình 3.15 và Hình 3.16 đã minh họa dự đoán xác suất trên tập dữ liệu *YelpResData*. Hiệu suất trong Bảng 3.6 của mô hình rừng mật độ trên tập dữ liệu huấn luyện nhà hàng về Accuracy, Precision, Recall và F1 lần lượt là 0,85; 0,82; 0,70 và 0,76. Trên tập kiểm tra, các giá trị tương ứng lần lượt là 0,80; 0,72; 0,67 và 0,69.

Bảng 3.6: Kết quả phân loại tình cảm của rừng mật độ trên YelpResData

Tập dữ liệu nhà hàng	Độ chính xác	Độ chính xác dương (P)	Độ bao phủ (R)	F1
Tập huấn luyện	0,85	0,82	0,70	0,76
Tập kiểm tra	0,80	0,72	0,67	0,69

3.7.4. Thảo luận kết quả thực nghiệm

Hình 3.3 và Hình 3.10 minh họa kết quả phân cụm đối với dữ liệu khách sạn và nhà hàng, lần lượt tạo thành 7 cụm và 9 cụm. Phân tích cho thấy các cụm được phân bố trên toàn bộ tập đặc trưng, bao gồm Entity, Aspect, Holder và Time; đáng chú ý, mỗi danh mục đặc trưng đều bao hàm sự kết hợp của nhiều cụm khác nhau. Điều này cho thấy quá trình phân cụm được thực hiện dựa trên các mẫu mật độ trong không gian đặc trưng đa chiều, thay vì phụ thuộc vào từng đặc trưng riêng lẻ. Sự khác biệt về số lượng cụm phản ánh rằng dữ liệu nhà hàng có cấu trúc mật độ phức tạp và đa dạng hơn, do đó đòi hỏi mức độ phân vùng chi tiết hơn so với dữ liệu khách sạn. Mặc dù 9 cụm trong miền nhà hàng đạt hiệu suất phân loại cao hơn, kết quả này không chứng minh rằng số lượng cụm lớn hơn thì hiệu suất tốt hơn, mà cho thấy cấu trúc dữ liệu nhà hàng phù hợp hơn với nhiệm vụ phân loại tình cảm trong nghiên cứu này. Nhìn chung, các kết quả khẳng định rằng rừng mật độ có khả năng tự động điều chỉnh độ chi tiết của phân cụm theo đặc điểm phân bố dữ liệu, trong đó số lượng cụm tối ưu thay đổi linh hoạt với mức độ phức tạp của từng miền dữ liệu.

So sánh kết quả đánh giá mô hình rừng mật độ ở giai đoạn phân loại trên hai tập dữ liệu YelpHotelData và YelpResData cho thấy tầm quan trọng của đặc trưng trên YelpResData phân bố đồng đều hơn giữa các thuộc tính (Entity 0,078; Aspect 0,075; Time 0,065; Holder 0,058) so với YelpHotelData (Entity ≈0,087; Time ≈0,085; Holder ≈0,077; Aspect ≈0,023), đặc biệt Aspect có đóng góp lớn hơn nhiều, qua đó nhiều khả năng góp phần mang lại hiệu suất phân loại tốt hơn (độ chính xác kiểm tra đạt 0,80 so với 0,75). Tuy nhiên, đối với YelpHotelData, mô hình thể hiện độ sâu phân chia nông hơn đối với các thuộc tính Time và Entity, phản ánh chiến lược cấu trúc cây khác biệt, trong đó các phép phân tách được thực hiện sớm và mang tính toàn cục hơn. Các kết quả này cho thấy tầm quan trọng của đặc trưng có mối tương quan chặt chẽ hơn với hiệu suất phân loại so với độ sâu phân chia của cây;

đồng thời, các đặc trưng rõ ràng và mạnh hơn trong dữ liệu nhà hàng cho phép mô hình khai thác hiệu quả hơn, dẫn đến khả năng phân loại tình cảm tốt hơn.

3.7.5. So sánh mô hình đề xuất

So sánh được đánh giá hiệu quả của mô hình rừng mật độ quan điểm F_{DF} với 7 phương pháp tiêu biểu thuộc các nhóm mô hình khác nhau: (1) Các phương pháp học máy truyền thống bao gồm Random Forest [65], Gradient Boosting [84] và Support Vector Machine (SVM) [86]; (2) Các mô hình học sâu được sử dụng gồm: LSTM [85], BiLSTM-Attention [88], BERT [81], RoBERTa [83]. Tất cả các mô hình được huấn luyện và đánh giá trên cùng hai tập dữ liệu YelpHotelData và YelpResData đã biểu diễn dữ liệu dưới dạng bộ năm quan điểm với cùng phân chia dữ liệu huấn luyện/kiểm tra để đảm bảo tính công bằng trong so sánh. Cụ thể, đối với các phương pháp học máy truyền thống dựa trên đặc trưng có cấu trúc gồm: Random Forest và Gradient Boosting dựa trên cây quyết định, SVM dựa trên siêu phẳng phân tách biên cực đại và mô hình rừng mật độ quan điểm đề xuất của luận án (F_{DF}), đặc trưng đầu vào là bộ năm quan điểm đã được mã hóa số theo đúng phương pháp từ Chương 2; trong khi đó, các mô hình học sâu: LSTM, BiLSTM-Attention, BERT, RoBERTa nhận trực tiếp câu văn bản đánh giá gốc tương ứng với từng bộ năm quan điểm làm đầu vào, theo đúng cách sử dụng tiêu chuẩn của các kiến trúc này nhằm phát huy đầy đủ khả năng học biểu diễn ngữ nghĩa.

Đánh giá tập trung vào ba khía cạnh quan trọng nhất: (1) Thời gian huấn luyện để đánh giá chi phí tài nguyên và thời gian phát triển mô hình, (2) Mức sử dụng bộ nhớ để đánh giá yêu cầu phần cứng và khả năng triển khai, (3) Khả năng giải thích để đánh giá tính minh bạch và khả năng đáp ứng yêu cầu pháp lý. Ngoài ra, chỉ số Accuracy và F1 được sử dụng để đánh giá hiệu suất dự đoán.

Bảng 3.7: Kết quả so sánh trên tập dữ liệu YelpHotelData

Mô hình	**Mô hình hóa cấu trúc**	**Độ chính xác**	**Điểm F1**	**Khả năng giải thích**	**Độ phức tạp tính toán**
Mô hình F_{DF} (Đề xuất)	Có	0,75	0,67	Rất cao	Θ(n×mtry×N×logN) + Θ(n×L×N×logN)
RoBERTa [83]	Không	0,74	0,66	Rất thấp	Θ(L^2×d) cho mỗi token (Rất cao - GPU)

BERT [81]	Không	0,74	0,65	Rất thấp	$\Theta(L^2 \times d)$ cho mỗi token (Rất cao - GPU)
BiLSTM-Attention [88]	Không	0,76	0,67	Thấp	$\Theta(n \times L \times d)$ (Trung bình)
Gradient Boosting [84]	Không	0,74	0,66	Cao	$\Theta(n \times N \times \log N)$ (Trung bình)
Random Forest [65]	Không	0,74	0,65	Cao	$\Theta(n \times N \times mtry \times \log N)$ (Thấp)
LSTM [85]	Không	0,71	0,63	Thấp	$\Theta(n \times L \times d^2)$ (Trung bình)
SVM [86]	Không	0,70	0,62	Trung bình	$\Theta(N^2 \sim N^3)$ (Cao)

Bảng 3.8: Kết quả so sánh trên tập dữ liệu YelpResData

Mô hình	**Mô hình hóa cấu trúc**	**Độ chính xác**	**Điểm F1**	**Khả năng giải thích**	**Độ phức tạp tính toán**
Mô hình F_{DF} (Đề xuất)	Có	0,80	0,69	Rất cao	$\Theta(n \times mtry \times N \times \log N) + \Theta(n \times L \times N \times \log N)$
RoBERTa [83]	Không	0,83	0,73	Rất thấp	$\Theta(L^2 \times d)$ cho mỗi token (Rất cao - GPU)
BERT [81]	Không	0,82	0,72	Rất thấp	$\Theta(L^2 \times d)$ cho mỗi token (Rất cao - GPU)
BiLSTM-Attention [88]	Không	0,78	0,68	Thấp	$\Theta(n \times L \times d)$ (Trung bình)
Gradient Boosting [84]	Không	0,78	0,68	Cao	$\Theta(n \times N \times \log N)$ (Trung bình)
Random Forest [65]	Không	0,77	0,66	Cao	$\Theta(n \times N \times mtry \times \log N)$ (Thấp)
LSTM [85]	Không	0,76	0,66	Thấp	$\Theta(n \times L \times d^2)$ (Trung bình)

SVM [86]	Không	0,80	0,69	Trung bình	Θ(N^2~N^3) (Cao)

Mô hình F_{DF} có độ phức tạp huấn luyện cao hơn Random Forest do thêm thành phần DBSCAN Θ(n×L×N×logN) tại các nút lá, nhưng chi phí thực tế thấp hơn lý thuyết vì DBSCAN chỉ chạy trên tập con tại nút lá. Các mô hình BERT/RoBERTa Θ(L^2×d) yêu cầu GPU chuyên dụng, trong khi mô hình F_{DF} chạy trên CPU với chi phí tính toán hợp lý.

Bảng 3.9: Tổng hợp kết quả trên hai tập dữ liệu

Mô hình	**Mô hình hóa cấu trúc**	**Độ chính xác trung bình**	**Điểm F1 trung bình**	**Khả năng giải thích**	**Độ phức tạp tính toán**
Mô hình F_{DF} (Đề xuất)	Có	0,78	0,68	Rất cao	Θ(n×mtry×N×logN) + Θ(n×L×N×logN)
RoBERTa [83]	Không	0,79	0,70	Rất thấp	Θ(L^2×d) cho mỗi token (Rất cao - GPU)
BERT [81]	Không	0,78	0,69	Rất thấp	Θ(L^2×d) cho mỗi token (Rất cao - GPU)
BiLSTM-Attention [88]	Không	0,77	0,68	Thấp	Θ(n×L×d) (Trung bình)
Gradient Boosting [84]	Không	0,76	0,67	Cao	Θ(n×N×logN) (Trung bình)
Random Forest [65]	Không	0,76	0,66	Cao	Θ(n×N×mtry×logN) (Thấp)
LSTM [85]	Không	0,74	0,65	Thấp	Θ(n×L×d^2) (Trung bình)
SVM [86]	Không	0,75	0,66	Trung bình	Θ(N^2~N^3) (Cao)

Ghi chú: Bảng 3.7, Bảng 3.8, Bảng 3.9 là số liệu đánh giá của mô hình rừng mật độ quan điểm được đề xuất và các mô hình so sánh. Trong đó, các mô hình học sâu sử dụng đầu vào là văn bản đánh giá gốc, khác với đầu vào bộ năm của các mô

hình rừng, nên được trình bày như mốc tham chiếu. Mọi giá trị trong bảng là giá trị trên tập kiểm tra (test), không dùng giá trị trên tập huấn luyện. Vì dữ liệu mất cân bằng (khoảng 63–65% mẫu positive) và các mô hình có đầu vào khác nhau, chênh lệch Accuracy giữa các mô hình chỉ có ý nghĩa tham khảo; phép so sánh trên cùng tập mẫu, cùng cách chia và cùng độ đo Balanced Accuracy được trình bày tại Phụ lục J (Bảng J.8).

Kết quả đánh giá đa chiều trong Bảng 3.9 cho thấy mô hình F_{DF} không chỉ đạt hiệu năng dự đoán tốt (Accuracy = 0,78, F1 = 0,68) mà còn mô hình hóa cấu trúc dữ liệu bao gồm biểu diễn ngữ nghĩa tách biệt tốt các nhóm quan điểm khác nhau và cấu trúc mật độ do DBSCAN phát hiện trước khi dự đoán.

Phân tích Bảng 3.7, Bảng 3.8, Bảng 3.9 cho thấy, trên tập dữ liệu YelpHotelData thì mô hình F_{DF} đạt 0,75, cao hơn Random Forest, Gradient Boosting (0,74) và SVM (0,70); các mô hình học sâu đạt 0,71 (LSTM), 0,76 (BiLSTM-Attention), 0,74 (BERT) và 0,74 (RoBERTa). Trên tập dữ liệu YelpResData, mô hình F_{DF} đạt 0,80, bằng SVM (0,80) và cao hơn Random Forest (0,77), Gradient Boosting (0,78); các mô hình học sâu đạt 0,76 (LSTM), 0,78 (BiLSTM-Attention), 0,82 (BERT) và 0,83 (RoBERTa). Tổng hợp hai tập, mô hình F_{DF} đạt độ chính xác trung bình 0,78 (Bảng 3.9), là mô hình có khả năng giải thích "Rất cao" và mô hình hóa cấu trúc.

Bên cạnh khả năng hiểu cấu trúc dữ liệu theo năm khía cạnh (biết số cụm, đặc trưng cụm, hiệu năng từng cụm, gán cụm cho dữ liệu mới và phát hiện ngoại lệ), theo số liệu đã công bố (chia ngẫu nhiên theo dòng bộ năm), mô hình F_{DF} đạt độ chính xác trung bình cao hơn mô hình Random Forest không phân cụm (0,78 so với 0,76, Bảng 3.9); khi chia theo đánh giá gốc (Phụ lục J, Bảng J.11, J.12, J.15), chênh lệch giữa hai mô hình không có ý nghĩa thống kê. Ngoài ra, ưu thế nổi bật của mô hình F_{DF} là khả năng giải thích, mô hình duy nhất đạt mức "Rất cao". Mô hình F_{DF} cung cấp ba tầng giải thích: (1) phân cụm dựa mật độ tự động phát hiện 7-9 cụm dữ liệu; (2) cây quyết định có thể trực quan hóa hoàn toàn và (3) giá trị SHAP [75] thông qua TreeSHAP. Trong khi các mô hình học sâu thường là mô hình hộp đen khó diễn giải [87], khả năng giải thích của rừng mật độ đáp ứng các yêu cầu cho các quyết định tự động [89].

Việc lựa chọn tham số ε và *MinPts* cho DBSCAN trong luận án dựa trên phân tích đồ thị K-lân cận, là một phương pháp bán tự động phổ biến nhưng vẫn cần quan sát thủ công điểm gãy (elbow point). Nghiên cứu gần đây của [180] đề xuất phương pháp tự động hóa hoàn toàn việc xác định ε dựa trên phân phối thống kê khoảng cách k-lân cận gần nhất, đồng thời giải quyết vấn đề gán nhãn cụm không nhất quán tại vùng mật độ chồng lấp, là hướng cải tiến tiềm năng cho việc tự động hóa quy trình lựa chọn tham số của mô hình rừng mật độ trong các nghiên cứu tiếp theo, giảm phụ thuộc vào quan sát trực quan.

Về kỹ thuật diễn giải, mô hình F_{DF} đề xuất sử dụng giá trị tương tác SHAP ở công thức (3.28) để phân tích tương tác cặp đặc trưng. Nghiên cứu thuật toán của [181] đề xuất phương pháp tính toán hiệu quả các tương tác Shapley bậc bất kỳ (any-order) chuyên biệt cho mô hình cây không chỉ giới hạn ở tương tác cặp đôi như cách tiếp cận hiện tại của luận án mà còn mở ra khả năng phân tích tương tác bậc ba trở lên giữa các thành phần bộ năm quan điểm (ví dụ Entity-Aspect-Time đồng thời) trong các nghiên cứu mở rộng sau này.

3.8. Công cụ sử dụng

Thực nghiệm của chương được triển khai thống nhất bằng ngôn ngữ lập trình R: các thực nghiệm chính (mục 3.7) dùng R phiên bản 4.3.2 trong môi trường phát triển tích hợp RStudio 2023.12.1; các thực nghiệm bổ sung và chạy lại tại Phụ lục F–J dùng R phiên bản 4.3.3 và 4.5.2 với cùng các gói và cùng hạt giống ngẫu nhiên.

Đối với xử lý dữ liệu, nghiên cứu sử dụng các gói dplyr, tidyr và data.table; dữ liệu gốc Yelp được đọc từ cơ sở dữ liệu SQLite bằng DBI và RSQLite. Mô hình rừng mật độ quan điểm F_{DF} được xây dựng từ gói dbscan (phân cụm mật độ, đồ thị k-khoảng cách) và randomForest, rpart (rừng phân loại theo từng cụm); giá trị SHAP được tính bằng treeshap và fastshap. Các mô hình so sánh trên đặc trưng bộ năm gồm Random Forest (randomForest), Gradient Boosting (gbm) và SVM (e1071); các kỹ thuật xử lý mất cân bằng lớp tại phụ lục dùng tham số sampsize/strata, classwt của randomForest và gói smotefamily (SMOTE). Các mô hình học sâu LSTM và BiLSTM-Attention được xây dựng trong R bằng gói torch (thực nghiệm so sánh thống nhất) và keras3 (đo cấu hình, thời gian, bộ nhớ). Các mô hình Transformer BERT, RoBERTa được tinh chỉnh bằng thư viện transformers (Hugging Face, nền PyTorch) và được gọi từ kịch bản R (qua reticulate hoặc lệnh hệ thống); dữ liệu vào,

chia tập, chọn ngưỡng và tính độ đo đều thực hiện trong R như các mô hình còn lại. Danh sách chi tiết các thư viện được trình bày trong Bảng D.1 (Phụ lục D).

3.9. Kết luận chương

Chương 3 đã trình bày chi tiết mô hình khai phá quan điểm với rừng mật độ, mô hình hóa cấu trúc dữ liệu theo hai chiều ngữ nghĩa tiềm ẩn và cấu trúc mật độ quan điểm. Mô hình được xây dựng bằng cách sử dụng DBSCAN để phát hiện các cụm quan điểm có mật độ khác nhau trong không gian biểu diễn, sau đó huấn luyện một rừng phân loại riêng biệt cho mỗi cụm dữ liệu đồng nhất và tổng hợp dự đoán. Kết quả thực nghiệm trên hai tập dữ liệu YelpHotelData và YelpResData cho thấy mô hình rừng mật độ quan điểm F_{DF} đạt độ chính xác trung bình 0,78 và F1 là 0,68, cao hơn mô hình rừng ngẫu nhiên (0,76) theo quy trình chia theo dòng của các bài báo (khi chia theo đánh giá gốc tại Phụ lục J, F_{DF} tương đương Random Forest); với đầu vào là văn bản gốc, các mô hình học sâu đạt độ chính xác trung bình từ 0,74 (LSTM) đến 0,79 (RoBERTa), trong khi mô hình rừng mật độ cung cấp khả năng giải thích qua cấu trúc cụm và giá trị SHAP. Đóng góp chính của chương nhằm giải quyết các vấn đề ảnh hưởng kết hợp theo tầm quan trọng của cụm quan điểm khác nhau, cho thấy mô hình hóa cấu trúc dữ liệu giúp giải thích kết quả phân loại theo từng cụm quan điểm. Cách tiếp cận giải thích mô hình thông qua SHAP trong mô hình rừng mật độ F_{DF} đóng góp vào xu hướng XAI (Explainable AI) đang phát triển mạnh [125]. So với các nghiên cứu học quần thể gần đây [59][134], mô hình rừng mật độ quan điểm F_{DF} đề xuất đã bổ sung khả năng phân tích cấu trúc cụm dữ liệu mất cân bằng, đặc điểm chưa được xử lý trong các nghiên cứu trước đó. Tuy nhiên, F_{DF} còn một số hạn chế: (i) DBSCAN dùng khoảng cách Euclidean trên không gian đặc trưng gốc nên chỉ nhận biết các cụm có quan hệ tuyến tính, cục bộ, không phản ánh được quan hệ phi tuyến giữa các quan điểm; (ii) số cụm thu được khá lớn (7 cụm trên YelpHotelData và 9 cụm trên YelpResData) và nhạy với tham số ε, MinPts, làm tăng số rừng cần huấn luyện; (iii) các rừng theo cụm được kết hợp bằng trung bình xác suất cố định, chưa học được trọng số kết hợp từ dữ liệu. Những hạn chế này dẫn đến Chương 4: mô hình rừng đa tạp quan điểm F_{MF} thay khoảng cách Euclidean bằng khoảng cách đa tạp học từ rừng không giám sát và kết hợp các rừng thành phần bằng Stacking.

CHƯƠNG 4. KHAI PHÁ QUAN ĐIỂM VỚI RỪNG ĐA TẠP

4.1. Giới thiệu

Chương này trình bày mô hình rừng đa tạp quan điểm nhằm giải quyết vấn đề dữ liệu quan điểm có cấu trúc đa tạp phức tạp và mối quan hệ phi tuyến. Khác với rừng phân loại F_{CF} (Chương 2) xử lý dữ liệu trực tiếp trong không gian $\mathbb{R}^5$ và rừng mật độ F_{DF} (Chương 3), phân cụm dựa trên khoảng cách Euclidean, rừng đa tạp F_{MF} khai thác cấu trúc nội tại *d'-chiều (d' < d = 5)* của không gian quan điểm thông qua ma trận quan hệ $W \in \mathbb{R}^{NxN}$, sử dụng khoảng cách đa tạp $dist_{\mathcal{M}}$ thay vì khoảng cách Euclidean. Mô hình đề xuất kết hợp ba thành phần: (1) học cấu trúc đa tạp từ rừng không giám sát để xây W; (2) phân cụm đa tạp DBSCAN($dist_{\mathcal{M}}$), khác với DBSCAN dùng khoảng cách Euclidean ở Chương 3; (3) K rừng chuyên môn hóa $\boldsymbol{\mathcal{F}}_k$, kết hợp qua Stacking meta-learner **M**.

Nội dung chính của chương bao hàm kết quả nghiên cứu của Bài báo số 6.

Phát biểu bài toán. Cho tập dữ liệu quan điểm $O = \{o_1, o_2, o_3, \ldots, o_N\}$, mỗi $o_i = (e_i, a_i, s_i, h_i, t_i)$ gồm thực thể e_i, khía cạnh a_i, giá trị tình cảm s_i, người giữ quan điểm h_i và thời điểm t_i. Với giả thuyết bổ sung rằng O nằm trên một đa tạp $\mathcal{M} \subset \mathbb{R}^5$ có chiều nội tại $d' \ll 5$ công thức (4.1). Mô hình học khoảng cách đa tạp $dist_{\mathcal{M}}$ từ rừng ngẫu nhiên không giám sát, phân K cụm $\mathcal{C} = \{C_1, \ldots, C_K, Noise\}$ trên không gian đa tạp, mỗi cụm được phân loại bởi rừng chuyên biệt $\mathcal{F}_k$.

Đầu vào: Ma trận $O_{mat} \in \mathbb{R}^{Nx5}$.

Đầu ra: Nhãn phân loại $\hat{y} \in C = \{\text{positive, negative}\}$, tổng hợp từ K rừng phân loại chuyên biệt qua meta-learner Stacking **M** (Thuật toán 4.3).

Mục tiêu: Tối thiểu hóa cùng hàm sai số $\mathcal{L}$ như Chương 2, Chương 3. Mô hình rừng đa tạp quan điểm (F_{MF}) nhằm cải thiện phân loại bằng cách khai thác cấu trúc $d' < 5$ chiều thực sự của dữ liệu, thay thế khoảng cách Euclidean bằng $dist_{\mathcal{M}}$.

4.2. Phân cụm quan điểm trên không gian đa tạp

4.2.1. Biểu diễn đặc trưng quan điểm

Trong không gian đặc trưng quan điểm, mỗi quan điểm được biểu diễn dưới dạng vector bộ năm:

$$o = (e, a, s, h, t) \in \mathbb{R}^d, \quad d = 5$$

Với tập N quan điểm $O = \{o_1, o_2, \dots, o_N\}$, ma trận dữ liệu có kích thước:

$$\mathrm{O} \in \mathbb{R}^{N \times d} \qquad d = 5$$

Trong đó mỗi hàng là một bộ năm quan điểm.

Lưu ý về ký hiệu: o $\in \mathbb{R}^5$ là biểu diễn đầy đủ của bộ năm. Do tình cảm s là nhãn cần dự đoán, ma trận quan hệ W, khoảng cách đa tạp và phân cụm DBSCAN trong thực nghiệm được xây dựng trên bốn thành phần (e, a, h, t), tức D = 4 chiều quan sát (Phụ lục I); điều kiện d′ < 5 ở chương này vì vậy được hiểu là chiều nội tại d′ nhỏ hơn số chiều quan sát, được ước lượng trực tiếp tại Phụ lục I.

4.2.2. Giả thuyết về đa tạp

Giả thuyết đa tạp cho rằng dữ liệu quan điểm thực tế không phân bố đều trong không gian đặc trưng $\mathbb{R}^d$ mà nằm trên một đa tạp $\mathcal{M}$ có chiều nội tại $d' \ll d$. Trong nghiên cứu này, mặc dù dữ liệu được biểu diễn trong không gian 5 chiều nhưng thực chất nằm trên hoặc gần một đa tạp chiều thấp $\mathcal{M}$ nhúng trong $\mathbb{R}^5$ ở công thức (4.1).

$$O \subset \mathcal{M} \subset \mathbb{R}^5, \dim(\mathcal{M}) = d' < 5 \tag{4.1}$$

Trong đó $d' \in \mathbb{N}$ ($d' < 5$ trong thực nghiệm) là chiều nội tại.

Các đặc trưng bộ năm không độc lập hoàn toàn: tồn tại tương quan mạnh giữa khía cạnh *a* và tình cảm *s*, dữ liệu chủ yếu biến thiên theo trục thực thể *e* và thời gian *t*. Khai thác cấu trúc này giúp loại bỏ nhiễu từ *d = 5* chiều gốc, tập trung vào d' < 5 chiều bản chất.

Giả thuyết đa tạp có căn cứ lý luận rõ ràng từ chính đặc điểm cấu trúc của dữ liệu *yelpHotelData* và *yelpResData*. Mặc dù bộ năm quan điểm $o = (e, a, s, h, t)$ được biểu diễn trong không gian $d = 5$ chiều, thực chất các chiều này không độc lập hoàn toàn mà phụ thuộc vào nhau theo những quy luật ẩn đặc thù của miền đánh giá dịch vụ. Cụ thể, có thể nhận ra ba quan hệ phụ thuộc chính:

Thứ nhất, chiều thực thể (e) và khía cạnh (a) có quan hệ phi tuyến chặt chẽ, một khách sạn cụ thể thường chỉ gắn liền với một tập hạn chế các khía cạnh được

đánh giá như phòng, vị trí và dịch vụ, nên không phải mọi tổ hợp (e, a) đều có thể xuất hiện trong dữ liệu.

Thứ hai, người đánh giá (h) có xu hướng đánh giá nhất quán theo phong cách cá nhân, tạo ra các đường cong có cấu trúc trong không gian 5 chiều thay vì phân bố ngẫu nhiên.

Thứ ba, điểm tình cảm (s) phụ thuộc phi tuyến vào tổ hợp (e, a, h), tức là s không biến thiên độc lập mà bị ràng buộc bởi ba chiều còn lại, đồng thời phân phối điểm sao tập trung mạnh ở hai cực (1★–2★ và 4★–5★) gợi ý cấu trúc cụm tự nhiên trong không gian tình cảm.

Ba quan sát này cùng chỉ ra một kết luận: dữ liệu Yelp thực chất nằm gần một đa tạp d'*-chiều* với $d' < 5$, tức là cấu trúc nội tại của dữ liệu thấp chiều hơn đáng kể so với không gian biểu diễn ban đầu. Tính phù hợp của giả thuyết này sẽ được kiểm chứng qua kết quả so sánh tại §4.8, khi mô hình rừng đa tạp khai thác cấu trúc d'*-chiều* được đối chiếu trực tiếp với mô hình rừng phân loại thông thường làm việc trên toàn bộ không gian $d = 5$ chiều gốc.

4.2.3. Ý tưởng xây dựng ma trận quan hệ đa tạp

Luận án sử dụng rừng cây quyết định không giám sát để xây dựng ma trận quan hệ đa tạp $W \in \mathbb{R}^{N \times N}$.

Với mỗi cây T_k, định nghĩa một hàm ánh xạ lá theo công thức:

$$l_{T_k}(o): \mathbb{R}^d \to \mathcal{L}_{T_k} \subset \mathbb{N} \tag{4.2}$$

trong đó $l_{T_k}(o)$ là chỉ số nút lá mà quan điểm o rơi vào trong cây l_{T_k} và $\mathcal{L}_{T_k}$ là tập tất cả nút lá của cây đó. Dựa trên hàm ánh xạ lá này, ma trận quan hệ thành phần $\mathrm{W}^{(k)}$ của một cây và ma trận quan hệ tổng hợp W của toàn rừng được trình bày tại mục 4.3 và mục 4.4.

4.2.4. Phân cụm DBSCAN trên không gian đa tạp

Sau khi có khoảng cách đa tạp $dist_{\mathcal{M}}$ được định nghĩa tại Mục 4.4, thuật toán DBSCAN [72] được áp dụng trên không gian đa tạp khác với Chương 3 dùng khoảng cách Euclidean. Điểm o_i là điểm lõi khi:

$$|\{o_j \mid dist_{\mathcal{M}}(o_i, o_j) \leq \varepsilon\}| \geq \mathrm{MinPts}. \tag{4.3}$$

Kết quả phân cụm được biểu diễn dưới dạng công thức (4.4),

$$\mathcal{C} = \{\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_\mathrm{K}, \mathrm{Noise}\}, \tag{4.4}$$

trong đó K là số cụm quan điểm trên đa tạp và Noise là tập các quan điểm ngoại lai. Phân cụm trên không gian đa tạp cho phép phát hiện các cụm có cấu trúc phi tuyến, phản ánh tốt hơn mối quan hệ ngữ nghĩa giữa các quan điểm và xác định nhiễu dựa trên mật độ nội tại của dữ liệu.

4.3. Cây đa tạp quan điểm

Cho một tập hợp N mẫu quan điểm không dán nhãn $O = \{o_1, o_2, \ldots, o_N\}$ với $o_i \in \mathbb{R}^d$ và $d = 5$. Mỗi mẫu đầu vào o_i được biểu diễn dưới dạng vector đặc trưng bộ năm $o_i = (e_i, a_i, s_i, h_i, t_i) \in \mathbb{R}^5$. Đầu ra dự đoán là nhãn tình cảm $\hat{y} \in \{Negative, Positive\}$.

Mô hình cây đa tạp [66] tối đa hóa độ lợi thông tin đa tạp tại mỗi nút. Quá trình tách nút j để tối ưu hóa được thực hiện bằng hàm mục tiêu ở công thức (4.5).

$$\Theta_j^* = \underset{\Theta \in \mathcal{T}_j}{\operatorname{argmax}}\, IG_{\mathcal{M}}(O_j, O_{jL}, O_{jR}, W) \tag{4.5}$$

Trong đó $\mathcal{T}_j$ là tập hợp tất cả các ngưỡng phân tách ứng viên tại nút j; O_{jL} và O_{jR} là tập mẫu bên trái và bên phải sau khi phân tách theo ngưỡng Θ.

Hàm $IG_{\mathcal{M}}$ là độ lợi thông tin (Information Gain) đa tạp, được định nghĩa ở công thức (4.6).

$$IG_{\mathcal{M}}(O_j, O_{jL}, O_{jR}, W) = H_{\mathcal{M}}(O_j) - \frac{|O_{jL}|}{|O_j|} H_{\mathcal{M}}(O_{jL}) - \frac{|O_{jR}|}{|O_j|} H_{\mathcal{M}}(O_{jR}) \tag{4.6}$$

Do thuật toán hoạt động trên dữ liệu không dán nhãn, cần xác định Entropy $H_{\mathcal{M}}$ không giám sát - Entropy áp dụng cho dữ liệu không nhãn dựa trên cấu trúc phân phối mật độ. Sử dụng phân phối Gaussian đa biến tại các nút, Entropy vi phân trong nghiên cứu [140] và áp dụng vào không gian đa tạp d'chiều với ma trận hiệp phương sai $\Lambda_{\mathcal{M}}(O)$, Entropy đa tạp $H_{\mathcal{M}}(O)$ được xác định ở công thức (4.7).

$$H_{\mathcal{M}}(O) = \frac{1}{2}\log\left[(2\pi e)^{d'} |\Lambda_{\mathcal{M}}(O)|\right] \tag{4.7}$$

với $|\Lambda_{\mathcal{M}}(O)|$ là định thức của ma trận hiệp phương sai có trọng số đa tạp $\Lambda_{\mathcal{M}}(O)$, thay vì dùng ma trận hiệp phương sai Euclidean thông thường, cây đa tạp sử

dụng ma trận hiệp phương sai có trọng số dựa trên ma trận quan hệ W ở công thức (4.8).

$$\Lambda_{\mathcal{M}}(O) = \frac{\sum_{i,j\in O} W_{ij}\,(o_i - \mu(O))(o_j - \mu(O))^T}{\sum_{i,j\in O} W_{ij}}, \tag{4.8}$$

Trong đó: $\mu(O) = \frac{1}{|O|} \quad_{o\in O} o$ là vector trung bình của tập O và $W_{ij} \in [0,1]$ là phần tử (i,j) của ma trận quan hệ đa tạp W. $W_{ij} \in [0,1]$, phản ánh mức độ gần nhau của hai quan điểm o_i và o_j trên đa tạp.

Thay $\Lambda_{\mathcal{M}}(O)$ từ công thức (4.8) vào công thức (4.6) được khai triển tường minh theo công thức (4.9).

$$IG_{\mathcal{M}} = \frac{1}{2}\left[\log|\,\Lambda_{\mathcal{M}}(O_j)\,| - \sum_{i\in\{L,R\}} \frac{|\,O_{ji}\,|}{|\,O_j\,|}\log|\,\Lambda_{\mathcal{M}}(O_{ji})\,|\right], \tag{4.9}$$

Cách định nghĩa này đảm bảo tiêu chí phân tách của cây đa tạp ưu tiên các phép tách làm giảm phương sai trên đa tạp, thay vì chỉ trong không gian Euclidean từ đó tạo ra các nút con phản ánh tốt hơn cấu trúc nội tại của dữ liệu quan điểm.

Với mỗi cây $T_k^{(w)}$trong rừng, ma trận quan hệ thành phần $W^{(k)} \in \{0,1\}^{N\times N}$ được xác định theo công thức (4.10) dựa trên hàm ánh xạ lá tại công thức (4.2), được kết quả:

$$W_{ij}^k = \mathbb{1}\left[L\left(o_i, T_k^{(w)}\right) = L\left(o_j, T_k^{(w)}\right)\right], \tag{4.10}$$

Trong đó

- $L\left(o_i, T_k^{(w)}\right)$ là chỉ số nút lá mà quan điểm o_i rơi vào cây $T_k^{(w)}$;
- $\mathbb{1}[.]$ là hàm chỉ thị bằng 1 khi điều kiện thỏa mãn và bằng 0 trong trường hợp ngược lại.
- $W_{ij}^k = 1$ khi o_i và o_j cùng nút lá trong cây $T_k^{(w)}$ và bằng 0 khi khác lá. Đây là thành phần cơ sở để xây dựng ma trận quan hệ tổng hợp tại Mục 4.4.

4.4. Rừng đa tạp quan điểm

4.4.1. Ma trận quan hệ đa tạp

Ma trận quan hệ đa tạp W được tổng hợp từ toàn bộ n_W cây không giám sát. Với rừng gồm n_W cây không giám sát $T_k^{(w)}$, trước tiên, với mỗi cây $T_k^{(w)}$, ma trận quan hệ thành phần $W^{(k)} \in \{0,1\}^{N\times N}$ được xác định theo công thức:

$$W_{ij}^{(k)} = \mathbf{1}\ l_{T_k^{(w)}}(o_i) = l_{T_k^{(w)}}(o_j) \tag{4.11}$$

Trong đó $W_{ij}^{(k)} = 1$ khi quan điểm o_i và o_j cùng nút lá trong cây $T_k^{(w)}$, bằng 0 khi khác lá. Ma trận quan hệ đa tạp tổng hợp $W \in \mathbb{R}^{N\times N}$ được xác định bằng trung bình các ma trận thành phần $W^{(k)}$ theo công thức:

$$W_{ij} = \frac{1}{n_W}\sum_{k=1}^{n_W} W_{ij}^{(k)} = \frac{1}{n_W}\sum_{k=1}^{n_W} \mathbf{1}\ l_{T_k^{(w)}}(o_i) = l_{T_k^{(w)}}(o_j) \tag{4.12}$$

Phép tính $\frac{1}{n_W}$ chuẩn hóa tổng đếm thành tần suất tương đối, xấp xỉ xác suất để o_i và o_j đồng nút lá, đảm bảo $W_{ij} \in [0,1]$, bất kể n_W lớn hay nhỏ. Ma trận W phản ánh mức độ gần nhau của các quan điểm trên đa tạp:

- $W_{ij} \approx 1$: hai quan điểm o_i và o_j thường xuyên đồng nút lá, tức là gần nhau trên đa tạp;
- $W_{ij} \approx 0$: hai quan điểm o_i và o_j hiếm khi đồng nút lá, tức là xa nhau trên đa tạp;
- $0 < W_{ij} < 1$: phản ánh mức độ tương đồng một phần.

4.4.2. Khoảng cách đa tạp

Từ ma trận W đã xây dựng tại mục 4.4.1, khoảng cách đa tạp giữa hai quan điểm o_i và o_j được định nghĩa cho 2 trường hợp là mẫu quan điểm đã biết và mẫu quan điểm mới.

Trường hợp 1- Khoảng cách giữa hai mẫu quan điểm o_i và o_j đã biết thuộc tập quan điểm O.

Khi cả o_i và o_j đều thuộc tập huấn luyện, giá trị W_{ij} đã có sẵn trong W. Khoảng cách đa tạp được xác định trực tiếp theo công thức:

$$dist_{\mathcal{M}}(o_i, o_j) = \sqrt{1 - W_{ij}} \quad (4.13)$$

Công thức (4.13) đảm bảo: $dist_{\mathcal{M}} = 0$ khi $W_{ij} = 1$, có nghĩa là hai quan điểm o_i và o_j luôn đồng nút lá; $dist_{\mathcal{M}} = 1$ khi $W_{ij} = 0$, có nghĩa là hai quan điểm o_i và o_j không bao giờ đồng nút lá; Với $0 < W_{ij} < 1$, khoảng cách tăng đơn điệu khi W_{ij} giảm. Khoảng cách $dist_{\mathcal{M}}$ được sử dụng trực tiếp trong điều kiện lân cận DBSCAN tại công thức (4.3).

Ma trận W đóng vai trò xấp xỉ rời rạc của cấu trúc đa tạp tiềm ẩn trong dữ liệu quan điểm, đồng thời cung cấp nền tảng cho việc tính $\Lambda_{\mathcal{M}}(O)$ tại công thức (4.8) và $IG_{\mathcal{M}}$ tại công thức (4.9).

Trường hợp 2- Khoảng cách giữa hai mẫu quan điểm o_i và o_j mới không thuộc tập quan điểm O.

Khi có mẫu quan điểm mới $o \notin O$ chưa xuất hiện trong tập huấn luyện, không thể tra trực tiếp ma trận $W \in \mathbb{R}^{N \times N}$ vì o chưa có hàng tương ứng. Thay vào đó, quan hệ đa tạp của o với k_{rep} điểm đại diện $\{o_1, o_2, \dots, o_{k_{rep}}\} \subset O$ với ($k_{rep} \ll N$) được ước lượng bằng cơ chế đếm đồng nút lá qua rừng FMF đã huấn luyện theo công thức:

$$\widehat{W}(o, o_i) = \frac{1}{n} \sum_{\ell}^{n} \mathbb{1}[L(o, T_\ell) = L(o_i, T_\ell)] \quad i = 1, \dots, \text{krep} \quad (4.14)$$

Trong đó:

- $T_\ell \in F_{MF}$: cây phân loại thứ ℓ ($\ell = 1, 2, 3, \dots, n$) trong rừng F_{MF} – đây là cây giai đoạn 2, phân biệt với cây không giám sát $T_k^{(w)}$ ở giai đoạn 1.
- $L(o, T_\ell)$: là chỉ số nút lá mà mẫu mới o rơi vào khi duyệt cây $T_\ell \in F_{MF}$.
- $L(o_i, T_\ell)$: chỉ số nút lá của điểm đại diện $o_i \in O$ trong cùng cây T_ℓ.
- n: số cây trong rừng F_{MF};
- $\widehat{W}(o, o_i)$: ước lượng quan hệ đa tạp cho mẫu mới, với $\widehat{W}(o, o_i) \in [0,1]$.
- $dist_{\mathcal{M}}(o, o_i)$: khoảng cách đa tạp giữa mẫu mới o và điểm đại diện o_i.
- k_{rep}: là số điểm đại diện khi ánh xạ mẫu mới ($k_{rep} \ll$ N).

Từ $\widehat{W}(o, o_i)$, khoảng cách đa tạp giữa mẫu mới o và điểm đại diện o_i được xác định theo công thức:

$$dist_{\mathcal{M}}(o, o_i) = \sqrt{1 - \widehat{W}(o, o_i)}, \quad i = 1, \dots, \text{krep} \quad (4.15)$$

Công thức (4.15) là trường hợp mở rộng của công thức (4.13) cho mẫu mới, thay giá trị W_{ij} chính xác bằng giá trị ước lượng $\widehat{W}(o, o_i)$. Công thức (4.14) và công thức (4.15) tái sử dụng chính cơ chế đếm đồng nút lá của rừng đa tạp quan điểm đã huấn luyện, áp dụng mẫu mới o vào từng cây T_ℓ, đếm số lần o và o_i rơi vào cùng nút lá.

4.5. Thuật toán và đồ thị tính toán mô hình rừng đa tạp quan điểm

4.5.1. Thuật toán và độ phức tạp

Trong mục này, mã giả thuật toán xây dựng và dự đoán cho rừng đa tạp quan điểm được trình bày. Thay vì sử dụng các phương pháp chiếu đa tạp truyền thống như ISOMAP hay UMAP, luận án đề xuất học cấu trúc đa tạp trực tiếp từ rừng ngẫu nhiên không giám sát để tận dụng tính chất đồng xuất hiện tại nút lá để xây dựng ma trận quan hệ đa tạp W mà không cần giải bài toán trị riêng toàn cục. Trong nội dung này, mã giả 3 thuật toán bao gồm: thuật toán xây dựng rừng đa tạp đơn lẻ (BuildManifoldForest), thuật toán dự đoán của rừng đa tạp (PredictManifoldForest) và thuật toán xây dựng kết hợp các rừng đa tạp (BuildManifoldForestEnsemble) được trình bày như sau:

Thuật toán 4.1. BuildManifoldForest

Đầu vào:

$O = \{o_i\}_{i=1}\ldots N$: tập dữ liệu huấn luyện, với: $o_i = (e_i, a_i, s_i, h_i, t_i)$ là bộ năm quan điểm thứ *i.*

n : số cây trong rừng.

n_w : số cây không giám sát dùng để xây ma trận W.

max_depth : độ sâu tối đa của cây.

min_split : số mẫu tối thiểu để tiếp tục phân tách.

d': số chiều nội tại ($d' < 5$).

Đầu ra: (F_{MF}, W): rừng đa tạp và ma trận quan hệ đa tạp $W \in \mathbb{R}^{NxN}$

// Giai đoạn 1 — Xây dựng ma trận quan hệ đa tạp (không giám sát)

1: $W \leftarrow 0 \in \mathbb{R}^{NxN}$ *// khởi tạo ma trận quan hệ*

2: for k = 1 to n_w do

3: $O_k^* \leftarrow$ Bootstrap(O, N) *// lấy mẫu có hoàn lại*

4: $T_k \leftarrow$ BuildRandomTree(O_k*, max_depth) *// cây ngẫu nhiên không giám sát, phân tách ngẫu nhiên.*

5: for each pair (o_i, o_j) do

6: if $L(o_i, T_k) = L(o_j, T_k)$ then *// đồng nút lá*

7: $W[i][j] \leftarrow W[i][j] + 1$

8: end if

9: end for

10: end for

11: $W \leftarrow W/n_w$ *// chuẩn hóa, xác suất đồng xuất hiện – công thức (4.12).*

// Giai đoạn 2 - Xây dựng rừng phân loại trên không gian đa tạp

12: $F \leftarrow \emptyset$

13: for i = 1 to n do

14: O_i* $\leftarrow$ Bootstrap(O, N)

15: $T_i \leftarrow$ BuildManifoldTree(O_i*, W, max_depth, min_split, depth=0)

16: $F_{MF} \leftarrow F_{MF} \cup \{Ti\}$

17: end for

18: return (F_{MF}, W)

Hàm phụ BuildManifoldTree(O, W, max_depth, min_split, depth)

Đầu vào:

O : tập bộ năm quan điểm tại nút;

W : ma trận quan hệ đa tạp N×N;

max_depth : độ sâu tối đa của cây.

min_split : số mẫu tối thiểu để phân tách.

depth : độ sâu hiện tại của nút.

Đầu ra: T_i cây đa tạp con được xây dựng tại nút hiện tại

// Bước 1- Kiểm tra điều kiện dừng đệ quy

1: if depth = max_depth or |O| < min_split or pure(O) then

2: return Leaf(majority_class(O)) *// nút lá là nhãn đa số*

3: end if

// Bước 2 - Chọn ngẫu nhiên tập đặc trưng ứng viên

4: Features ← RandomSubset(χ, M) // χ = {e, a, s, h, t}

// Bước 3 - Tìm phân tách tối ưu theo IG đa tạp

5: best_gain ← −∞ *// Đảm bảo phân tách đầu tiên luôn được chọn*

6: for each f ∈ Features, θ ∈ thresholds(O, f) do

7: $O_L \leftarrow \{o \in O : o[f] \leq \theta\}$; $O_R \leftarrow \{o \in O : o[f] > \theta\}$

8: gain ← $IG_{\mathcal{M}}$ (O, O_L, O_R, W) *// theo công thức (4.7) và (4.9)*

9: if gain > best_gain then

10: best_gain ← gain

11: f* ← f; θ* ← θ *// cập nhật phân tách tốt nhất*

12: end if

13: end for

// Bước 4 - Phân tách dữ liệu và gọi đệ quy

14: $O_L \leftarrow \{ o \in O : o[f^*] \leq \theta^* \}$

15: $O_R \leftarrow \{ o \in O : o[f^*] > \theta^* \}$

16: T_L ← BuildManifoldTree(O_L, W, max_depth, min_split, depth+1)

17: T_R ← BuildManifoldTree(O_R, W, max_depth, min_split, depth+1)

18: return Node(f*, θ*, T_L, T_R) *// nút trong với phân tách tối ưu (f*, θ*)*

Độ phức tạp của thuật toán 4.1- BuildManifoldForest

Độ phức tạp huấn luyện được tách thành hai giai đoạn:

Giai đoạn 1 - Xây ma trận quan hệ đa tạp W (n_W cây $T_k^{(w)}$ không giám sát):

Xây dựng n_W cây không giám sát $T_k^{(w)}$. Mỗi cây được xây bằng phân tách ngẫu nhiên, chiều sâu trung bình $logN$. Tại mỗi nút duyệt N mẫu để ghi nhận đồng nút lá vào W. Chi phí một cây là $\Theta(N \times logN)$, tổng n_W cây là:

$$\Theta(n_W \times N \times logN) \tag{4.16}$$

Ở giai đoạn này thuật toán chạy một lần duy nhất, ma trận W được chia sẻ cho giai đoạn tiếp theo.

Giai đoạn 2 - Xây dựng rừng phân loại F_{MF} (n cây T_i có giám sát):

Xây dựng *n* cây phân loại T_i bằng hàm phụ BuildManifoldTree. Tại mỗi nút, hàm $IG_{\mathcal{M}}$ tính $\Lambda_{\mathcal{M}}(O)$ trên không gian d' chiều nội tại $(d' < 5)$. Chi phí tìm ngưỡng phân tách tối ưu tại mỗi nút là $\Theta(N \times d')$, tổng n cây với chiều sâu $logN$:

$$\Theta(n \times N \times logN \times d') \tag{4.17}$$

Vì $d' \leq d$, nên giai đoạn này nhanh hơn rừng phân loại F_{CF} khoảng d/d' lần tại bước tính ngưỡng.

Tổng độ phức tạp của thuật toán 4.1:

$$\begin{aligned} &\Theta(n_W \times N \times logN) + \Theta(n \times N \times logN \times d') \\ &= \Theta\big(N \times logN \times (n_W + n \times d')\big) \end{aligned} \tag{4.18}$$

Trong đó:

- n_W: số cây không giám sát dùng xây ma trận W;
- n: số cây trong rừng phân loại đa tạp;
- N: số mẫu huấn luyện;
- d': số chiều nội tại với $d' < d = 5$.

Thuật toán 4.2. PredictManifoldForest

Đầu vào:

(F_{MF}, W): rừng đa tạp quan điểm đã huấn luyện (đầu ra của Thuật toán 4.1);

o = (e, a, s, h, t) : mẫu quan điểm mới cần phân loại.

Đầu ra:

ŷ ∈ C = {positive, negative}: nhãn phân loại quan điểm.

1: $\widehat{W}(o, o_i) \leftarrow$ ComputeManifoldDist(o, o_i, F_{MF}) *// ước lượng $\widehat{W}$ mẫu mới.*

2: votes ← [ ]

3: for ℓ = 1 to n do

4: votes ← votes ∪ { Traverse(T_ℓ, o, W) } *// duyệt cây T_i theo khoảng cách đa tạp*

5: end for

6: ŷ ← MajorityVote(votes) *// biểu quyết đa số*

7: return ŷ

Độ phức tạp của Thuật toán 4.2-PredictManifoldForest

Độ phức tạp của dự đoán rừng đa tạp gồm hai bước:

Bước 1- Tính giá trị ước lượng $\widehat{W}(o, o_i)$ cho k_{rep} cặp (o, o_i) mẫu mới.

Mẫu mới $o \notin O$ nên phải duyệt từ gốc đến lá mỗi cây $T_\ell \in F_{MF}$, chi phí $\Theta(logN)$ mỗi cây. Với k_{rep} điểm đại diện, mỗi điểm duyệt n cây thì chi phí là:

$$\Theta\left(k_{rep} \times\ n \times logN\right) \tag{4.19}$$

Bước 2- Duyệt n cây để phân loại

Duyệt n cây $T_\ell \in F_{MF}$ từ gốc đến lá, mỗi cây tốn chi phí $\Theta(logN)$:

$$\Theta(\ n \times logN) \tag{4.20}$$

Tổng độ phức tạp thuật toán 4.2:

$$\begin{aligned} &\Theta\left(k_{rep} \times n\ \times \log N + \ n \times logN\right) \\ &=\ \Theta\left((k_{rep} + 1) \times n\ \times \log N\right) \end{aligned} \tag{4.21}$$

Trong đó:

- n: số lượng cây trong rừng đa tạp;
- N: số lượng mẫu dữ liệu huấn luyện được sử dụng để xây dựng cây;
- $n \cdot logN$: chi phí duyệt n cây từ gốc đến lá, mỗi cây có chiều sâu trung bình $logN$, duyệt theo khoảng cách đa tạp từ ma trận W.
- k_{rep}: số điểm đại diện khi ánh xạ mẫu mới $k_{rep} \ll N$.

Thuật toán 4.3. BuildManifoldForestEnsemble

Đầu vào:

O : tập dữ liệu huấn luyện

n, n_W : số cây phân loại và số cây không giám sát

max_depth, min_split : tham số xây cây

ε (eps), MinPts : tham số phân cụm DBSCAN trên đa tạp (dùng $d_{\mathcal{M}}$, ≠ Euclidean).

O_{Val}: tập xác thực để huấn luyện meta-learner

Đầu ra:

$\{(\boldsymbol{\mathcal{F}}_1,W), \ldots, (\boldsymbol{\mathcal{F}}_k,W)\}$: K rừng chuyên môn hóa theo K cụm đa tạp;

M: meta-learner Stacking đã huấn luyện.

// Bước 1 — Xây ma trận quan hệ đa tạp W

1: (_, W) ← BuildManifoldForest_Phase1(O, n_w) *// chạy Giai đoạn 1 của thuật toán 4.1 → W; không chạy Giai đoạn 2.*

// Bước 2 — Phân cụm đa tạp bằng DBSCAN

2: $\mathcal{C} \leftarrow$ DBSCAN($dist_{\mathcal{M}}$ (W), ε, MinPts) *// phân cụm DBSCAN, tính khoảng cách đa tạp theo công thức (4.13).*

3: $\mathcal{C} = \{C_1, C_2, \ldots, C_k, \text{Noise}\}$

// Bước 3 — Xây K rừng $\boldsymbol{\mathcal{F}}_k$ chuyên môn hóa ở giai đoạn 2 của thuật toán 4.1 trên từng cụm

4: Forests $\leftarrow \emptyset$

5: for k = 1 to K do

6: $\boldsymbol{\mathcal{F}}_k \leftarrow$ BuildManifoldForest_Phase2(C_k, W, n, max_depth, min_split)

// Giai đoạn 2 thuật toán 4.1 chạy riêng trên cụm C_k.

7: Forests $\leftarrow$ Forests $\cup$ {($\boldsymbol{\mathcal{F}}_k$, W)}

8: end for

// Bước 4 — Tổng hợp dự đoán bằng học quần thể Stacking

9: $\hat{Y}_{val} \leftarrow [\,]$

10: for k = 1 to K do

11: $\hat{y}_k \leftarrow$ PredictManifoldForest($\boldsymbol{\mathcal{F}}_k$, W, O_{Val})

// Thuật toán 4.2 dùng ký hiệu $F_{MF} \leftarrow \mathcal{F}_k$

12: $\hat{Y}_{val} \leftarrow \hat{Y}_{val} \cup \{\hat{y}_k\}$

13: end for

14: **M** $\leftarrow$ TrainMetaLearner($\hat{Y}_{val}$, O_{Val}.labels)

15: return (Forests, **M**)

Độ phức tạp Thuật toán 4.3 - BuildManifoldForestEnsemble

Thuật toán 4.3 bao gồm bốn bước độc lập, thực hiện tuần tự:

Bước 1: Xây dựng ma trận quan hệ đa tạp W (Giai đoạn 1 Thuật toán 4.1):

$$\Theta(n_W \times N \times \log N) \tag{4.22}$$

Chi phí xây n_W cây không giám sát $T_k^{(w)}$, mỗi cây có chiều sâu trung bình log N tầng. Đây là chi phí một lần duy nhất, $W \in \mathbb{R}^{N\times N}$ được chia sẻ cho tất cả K rừng $\boldsymbol{\mathcal{F}}_k$ không cần tính lại.

Về khả năng mở rộng quy mô khi N rất lớn (ví dụ tập yelpResData gốc trước khi lấy mẫu có gần 14,8 triệu bộ năm, mục 3.7.2.2) và ma trận $W \in \mathbb{R}^{N\times N}$ không thể lưu trữ toàn phần trong bộ nhớ, luận án định hướng ba phương án cho các nghiên cứu tiếp theo: (i) lấy mẫu con (subsampling) trên tập huấn luyện trước khi xây W, tương tự cách tiếp cận N = 112.275 đã dùng xuyên suốt các thực nghiệm hiện tại; (ii) biểu diễn thưa (sparse representation) của W, chỉ lưu các cặp (o_i, o_j) đồng nút lá thay vì toàn bộ ma trận N×N, tận dụng tính chất W chủ yếu gồm các giá trị 0 khi n_W không quá lớn so với N; (iii) tính W cục bộ theo từng cụm DBSCAN hoặc từng nút thay vì toàn cục, giới hạn phép tính đồng nút lá trong phạm vi từng cụm C_k, giúp giảm chi phí bộ nhớ và thời gian từ $\Theta(N^2)$ xuống gần tuyến tính theo N. Luận án chưa cài đặt và đánh giá thực nghiệm ba phương án này; đây là hướng mở rộng cụ thể cho các nghiên cứu tiếp theo khi áp dụng mô hình rừng đa tạp cho dữ liệu có quy mô lớn hơn.

Bước 2: Phân cụm đa tạp DBSCAN

$$\Theta(N \times \log N) \qquad \text{(k-d tree trên d′chiều nội tại)} \tag{4.23}$$

DBSCAN chạy trên N điểm với khoảng cách $dist_{\mathcal{M}}(o, o_i) \in [0,1]$ theo công thức (4.13). Nếu dùng k-d tree với $d' < 5$ chiều nội tại thì chi phí tìm lân cận mỗi điểm là $\Theta(\log N)$. Chi phí Bước 2 nhỏ hơn Bước 1 do không có hệ số n_W.

Bước 3 – Xây dựng K rừng $\mathcal{F}_k$ chuyên môn hóa:

Mỗi rừng $\mathcal{F}_k$ được xây dựng trên cụm C_k với $N_k = |C_k|$ mẫu theo thuật toán 4.1 ở Giai đoạn 2:

$$\sum_{k=1}^{K} \Theta(n \times N_k \times \log N_k \times d') \tag{4.24}$$

trong đó $N_k = |C_k|$ là số mẫu trong cụm k. Vì $\sum_{k=1}^{K} N_k \leq N$ nên:

$$\sum_{k=1}^{K} N_k \times \log N_k \leq N \times \log N \tag{4.25}$$

Do đó tổng Bước 3 bị chặn trên bởi $\Theta(n \times N \times log\, N \times d')$.

Bước 4 – Huấn luyện meta-learner Stacking:

Độ phức tạp thuật toán của quá trình huấn luyện meta-learner Stacking gồm: tạo ma trận đặc trưng $K \times N_{val}$ từ K dự đoán tốn chi phí $\Theta(K \times N_{val})$ và huấn luyện Random Forest meta-learner trên N_{val} mẫu xác thực tốn chi phí $\Theta(N_{val} \times \log N_{val})$:

$$\Theta(K \times N_{val} + N_{val} \times \log N_{val}) \quad (4.26)$$

trong đó $N_{val} = |O_{val}|$ là kích thước tập xác thực.

Tổng hợp độ phức tạp của Thuật toán 4.3:

$$\Theta(N \times \log N \times (n_W + n \times d')) + \Theta(K \times N_{val}) \quad (4.27)$$

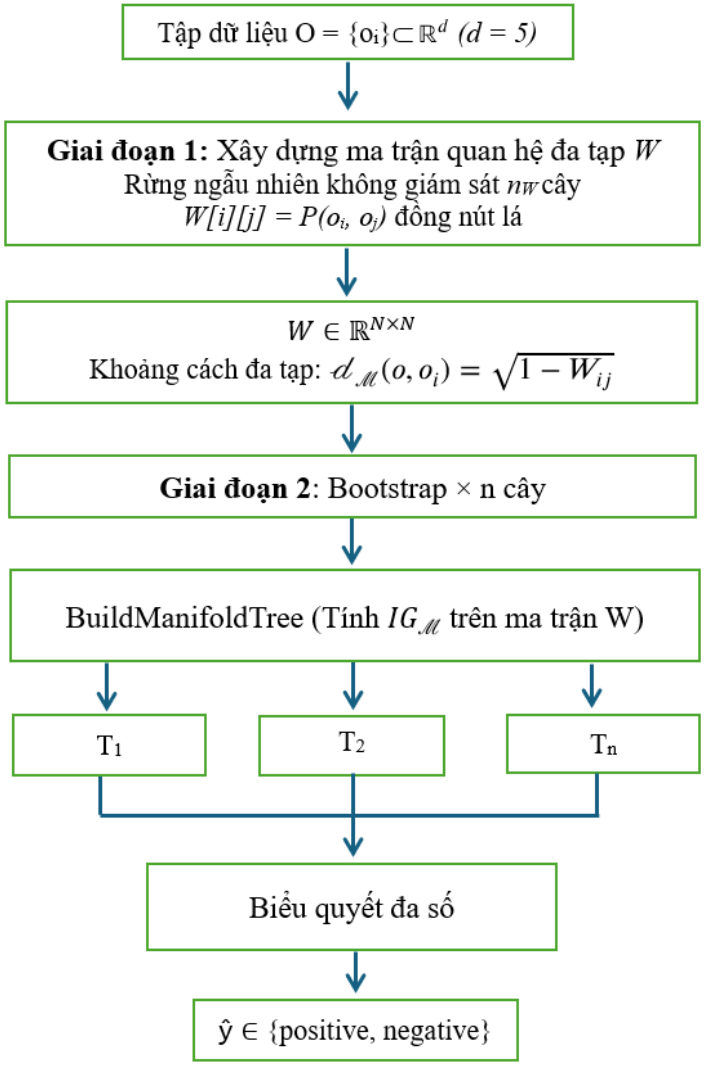
Trong đó:

- n_W: số cây không giám sát $T_k^{(w)}$ để xây dựng W (Bước 1).
- n: số cây phân loại T_i trong mỗi rừng $\boldsymbol{\mathcal{F}}_k$ (Bước 3).
- N: số mẫu huấn luyện.
- d': chiều nội tại của đa tạp $\mathcal{M}$, $d' < 5$.
- K: số cụm từ DBSCAN.

4.5.2. Đồ thị tính toán mô hình rừng đa tạp quan điểm

Đồ thị tính toán được đưa vào để hai giai đoạn chính được biểu diễn rõ ràng: Giai đoạn 1 xây dựng ma trận quan hệ đa tạp W từ rừng không giám sát; Giai đoạn 2 xây dựng rừng phân loại sử dụng W làm cơ sở tính. Kết quả dự đoán thu được qua biểu quyết đa số (Hình 4.1).

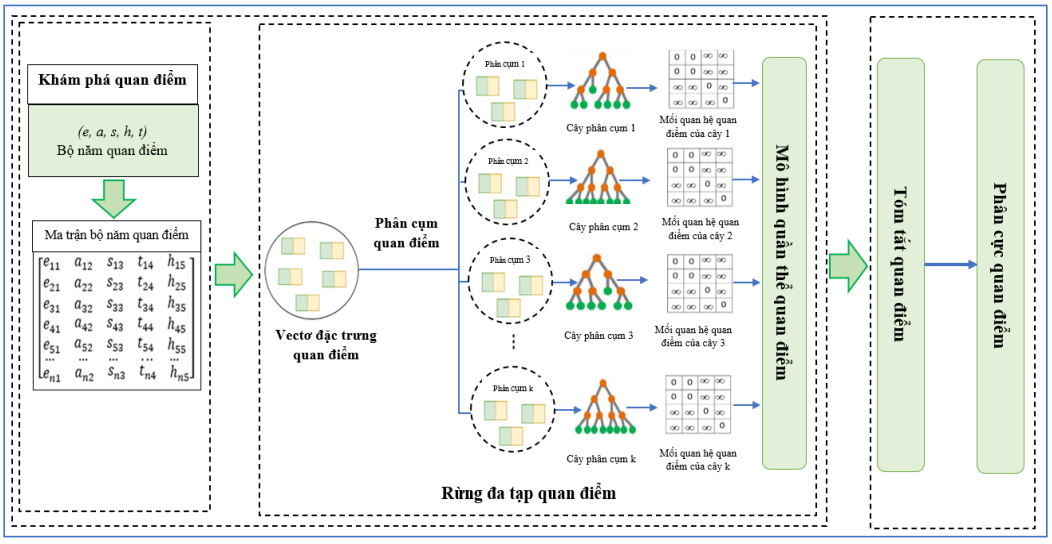
Đồ thị tính toán cho thấy hai giai đoạn được phân tách rõ ràng, trong giai đoạn 1, ma trận quan hệ đa tạp W được học từ tập O thông qua rừng ngẫu nhiên không giám sát, không cần nhãn và không cần giải bài toán trị riêng. Trong giai đoạn 2, W được dùng để định nghĩa khoảng cách đa tạp phục vụ tiêu chí phân tách $IG_{\mathcal{M}}$ trong quá trình xây rừng phân loại có giám sát. Điểm khác biệt so với ISOMAP/UMAP là toàn bộ cấu trúc đa tạp được học nội tại từ dữ liệu quan điểm mà không cần siêu tham số chiều nội tại $d_{\mathcal{M}}$ hay số láng giềng k-NN cho bước chiếu.



Hình 4.1: Đồ thị tính toán của mô hình rừng đa tạp quan điểm

4.6. Mô hình tổng quan khai phá quan điểm với rừng đa tạp

Mô hình tổng quan rừng đa tạp quan điểm được đề xuất thông qua Hình 4.2.



Hình 4.2: Sơ đồ hóa rừng đa tạp quan điểm

Hình 4.2 mô tả kiến trúc của mô hình Rừng đa tạp quan điểm, bao gồm bốn giai đoạn chính thực hiện tuần tự: (1) tiền xử lý và biểu diễn dữ liệu quan điểm, (2) phân cụm dựa trên cấu trúc đa tạp, (3) xây dựng mô hình dự đoán chuyên môn hóa và (4) tổng hợp kết quả cuối cùng.

Giai đoạn 1: Dữ liệu quan điểm đầu vào được biểu diễn dưới dạng bộ năm (e, a, s, h, t) và chuyển đổi thành ma trận bộ năm quan điểm kích thước N×5, với mỗi hàng biểu diễn một quan điểm.

Giai đoạn 2: Vector đặc trưng từ giai đoạn 1 được sử dụng để xây dựng ma trận quan hệ đa tạp thông qua rừng quyết định không giám sát. Dựa trên ma trận này, thuật toán DBSCAN phân cụm dữ liệu thành K cụm quan điểm, mỗi cụm chứa các quan điểm có cấu trúc đa tạp tương đồng. Cách tiếp cận này cho phép phát hiện các cụm có cấu trúc phi tuyến tính phức tạp mà phương pháp dựa trên khoảng cách Euclidean không thể tìm ra.

Giai đoạn 3: Đối với mỗi cụm xây dựng cây đa tạp quan điểm chuyên môn hóa để học các đặc trưng riêng biệt của cụm đó, tận dụng tính đồng nhất về cấu trúc đa tạp. Mỗi cây tạo ra ma trận mối quan hệ quan điểm của cụm tương ứng, dự đoán phân cực tình cảm tích cực hoặc tiêu cực cho các quan điểm trong cụm.

Giai đoạn 4: Các dự đoán từ K mô hình chuyên môn hóa được tổng hợp thông qua kỹ thuật học quần thể stacking với meta-learner. Meta-learner học cách kết hợp tối ưu các dự đoán, gán trọng số phù hợp cho mỗi cụm dựa trên độ tin cậy và hiệu năng, đưa ra kết quả dự đoán phân cực tình cảm cuối cùng.

4.7. Thực nghiệm

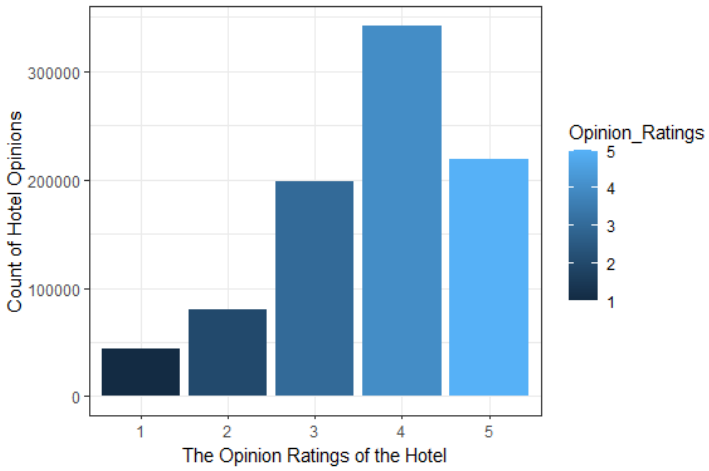
4.7.1. Tập dữ liệu sử dụng

Các thử nghiệm được thực hiện trên hai bộ dữ liệu *yelpHotelData* và *yelpResData* về các đánh giá khách sạn và nhà hàng. Tập dữ liệu *YelpHotelData* bao gồm 3 bảng dữ liệu có tên là Hotel, Review và Reviewer. Tương tự, tập dữ liệu YelpResData cũng bao gồm 3 bảng dữ liệu có tên là Restaurant, Review và Reviewer, chi tiết của hai tập dữ liệu được trình bày ở phần Phụ lục A, Phụ lục B.

4.7.2. Tiền xử lý dữ liệu

4.7.2.1. Tiền xử lý dữ liệu yelpHotelData

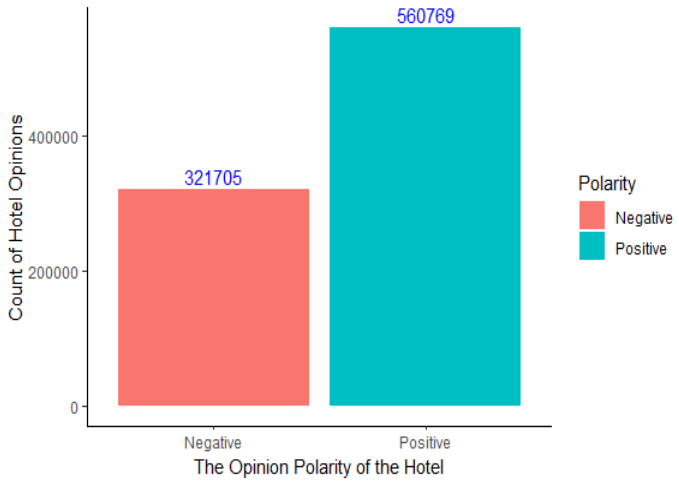
Tập dữ liệu *yelpHotelData* ban đầu bao gồm 3 bảng dữ liệu *Hotel, Review và Reviewer* có mối quan hệ với nhau. Kết quả chuyển đổi từ dữ liệu ban đầu thành dữ liệu cấu trúc bộ năm quan điểm có 882.474 mẫu.



Hình 4.3: Phân phối xếp hạng quan điểm khách sạn

Hình 4.3 minh họa phân phối xếp hạng quan điểm khách sạn. Giá trị quan điểm của các đặc trưng được đánh giá theo xếp hạng (từ 1★ đến 5★), tương ứng với 5 mức độ phân cực bao gồm "very positive", "positive", "neutral", "negative" và "very negative". Thực nghiệm được thực hiện trên hai nhãn nhị phân "negative" và "positive". Các mẫu có phân cực "very negative" (1★) và "negative" (2★) được gộp thành nhãn negative; các mẫu có phân cực "positive" (4★) và "very positive" (5★) được gộp thành nhãn positive; các mẫu trung lập "neutral" (3★) được gộp vào nhãn negative (xem giải thích và trích dẫn tại §3.7.2.2). Kết quả chuyển đổi quan điểm khách sạn có 560.769 mẫu positive và 321.705 mẫu negative trong Hình 4.4.

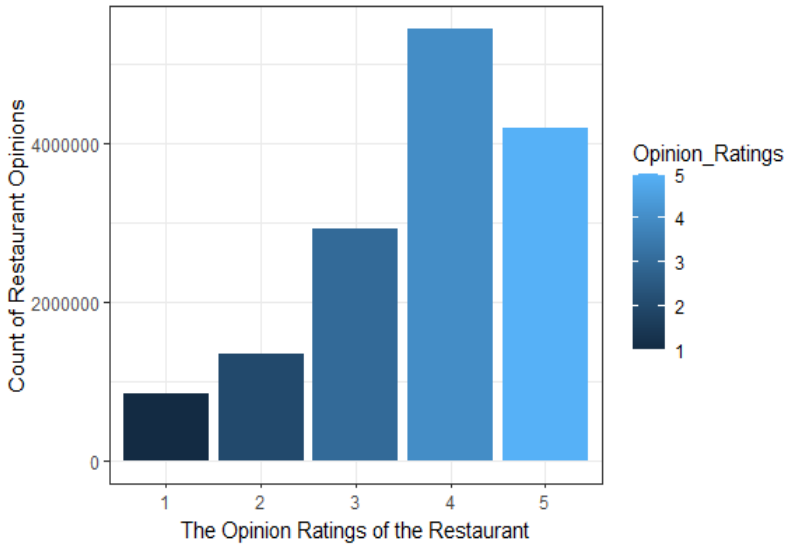
Ghi chú: 560.769 (positive, 4★–5★) + 321.705 (negative, gộp 1★–3★) = 882.474, khớp đúng tổng số mẫu bộ năm đã nêu ở đầu mục 4.7.2.1 – xem giải thích đầy đủ về ngưỡng nhị phân bất đối xứng và trích dẫn tại §3.7.2.2.



Hình 4.4: Phân phối phân cực quan điểm khách sạn

4.7.2.2. Tiền xử lý dữ liệu yelpResData

Kết quả chuyển đổi từ dữ liệu ban đầu thành cấu trúc dữ liệu bộ năm quan điểm có 14.791.500 mẫu.

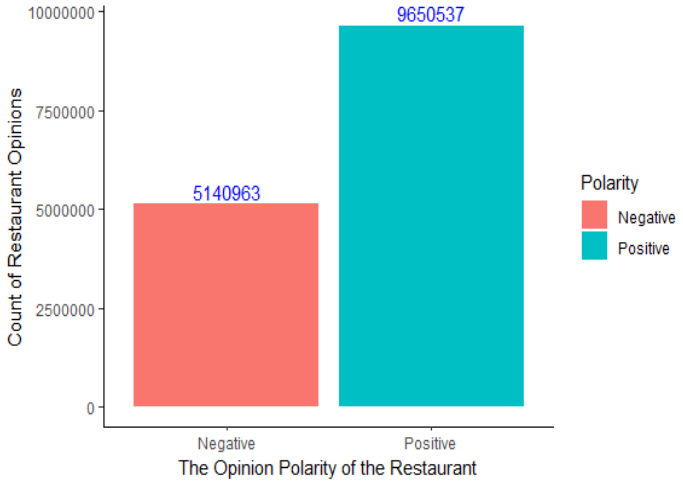


Hình 4.5: Phân phối xếp hạng quan điểm trên bộ dữ liệu yelpResData

Hình 4.5 minh họa phân phối xếp hạng quan điểm nhà hàng theo các đặc trưng thông qua các mức đánh giá (từ 1★ đến 5★), tương ứng với 5 mức độ phân cực quan điểm gồm "very negative", "negative", "neutral", "positive" và "very positive". Thực nghiệm được thực hiện trên hai cấp độ phân cực quan điểm "positive" và "negative" nên các mẫu có phân cực quan điểm "very negative" và "negative" được chuyển thành mẫu "negative" và gộp các mẫu có phân cực "positive" và "very positive" thành một mẫu "positive"; mẫu trung lập "neutral" (3★) cũng được gộp vào nhãn negative (xem

giải thích và trích dẫn tại §3.7.2.2). Kết quả chuyển đổi dữ liệu có tổng cộng 14.791.500 mẫu (5.140.963 mẫu negative và 9.650.537 mẫu positive) trong Hình 4.6.

Ghi chú: 5.140.963 (negative, gộp 1★–3★) + 9.650.537 (positive, 4★–5★) = 14.791.500, khớp đúng tổng số mẫu bộ năm đã nêu ở đầu mục 4.7.2.2 – cùng ngưỡng nhị phân bất đối xứng áp dụng cho yelpHotelData, xem giải thích và trích dẫn tại §3.7.2.1/§3.7.2.2



Hình 4.6: Phân phối quan điểm nhà hàng là positive và negative.

Xử lý mất cân bằng nhãn: Tương tự §3.7.2.2 (Chương 3), phân phối nhãn trên hai tập Yelp (sau chuyển đổi bộ năm, Hình 4.4 và Hình 4.6) có tỷ lệ mất cân bằng xấp xỉ 1,7:1 đối với yelpHotelData và 1,9:1 đối với yelpResData (positive:negative). Trong mô hình rừng đa tạp, cơ chế phân cụm DBSCAN trên không gian đa tạp tự nhiên tạo ra các cụm riêng biệt phản ánh cấu trúc nội tại của dữ liệu, trong đó các mẫu thiểu số (negative) tập trung trong các cụm đặc thù. Mỗi rừng ngẫu nhiên chuyên biệt theo cụm vì vậy học được tốt hơn trên vùng cục bộ. Bổ sung thêm: tập dữ liệu được chia theo tỷ lệ 60/20/20 (train/validation/test) bằng phương pháp lấy mẫu phân tầng; kiểm định chéo k-fold (k = 5 với yelpHotelData, k = 10 với yelpResData) được áp dụng để đánh giá độ ổn định của mô hình.

4.7.3. Các kịch bản thực nghiệm

4.7.3.1. Mô hình khai phá quan điểm với rừng đa tạp dựa trên bộ năm quan điểm trên tập dữ liệu khách sạn yelpHotelData

Từ tập dữ liệu quan điểm khách sạn yelpHotelData sau khi tiền xử lý. Quá trình xây dựng thuật toán rừng đa tạp quan điểm thực hiện các bước sau:

Bước 1: Chia dữ liệu thành 3 tập dữ liệu: tập dữ liệu quan điểm huấn luyện (60%), tập dữ liệu quan điểm xác thực (20%) và tập dữ liệu quan điểm kiểm thử (20%). Thực hiện phân cụm dữ liệu với các thiết lập tham số mô hình tại Bảng 4.1 trên tập huấn luyện không dán nhãn. Kết quả có 3 phân cụm.

Bảng 4.1: Thiết lập tham số mô hình Rừng đa tạp quan điểm trong thực nghiệm

Tham số	**Ý nghĩa**	**Giá trị thử nghiệm**	**Giá trị được chọn**	**Cơ sở lựa chọn**
ε (eps)	Bán kính vùng lân cận: điểm o_i là lân cận của o_j nếu $dist_{\mathcal{M}}(o_i, o_j) \leq \varepsilon$. Ở đây $dist_{\mathcal{M}}$ là khoảng cách đa tạp.	0,1; 0,3; 0,5; 0,7; 1,0	0,3	Căn cứ vào biểu đồ k-distance trên không gian đa tạp. ε = 0,3 (nhỏ hơn ε = 0,5 dùng ở Chương 3) vì khoảng cách đa tạp $dist_{\mathcal{M}} = \sqrt{1 - W_{ij}} \in [0,1]$ có phân phối dày đặc hơn Euclidean, cần ngưỡng nhỏ hơn để tạo cụm chặt
MinPts	Số điểm tối thiểu trong vùng lân cận để hình thành một cụm lõi	3, 5, 7, 10	5	Quy tắc kinh nghiệm MinPts ≥ d+1 với d = 4 (số chiều đặc trưng đầu vào e, a, h, t sau khi tách nhãn tình cảm s); MinPts = 5 = d+1 cho kết quả ổn định. Giữ nguyên như Chương 3 để so sánh công bằng giữa hai mô hình

Bước 2: Xây dựng 3 mô hình rừng ngẫu nhiên trên 3 phân cụm là Rừng ngẫu nhiên 1 (RF1), Rừng ngẫu nhiên 2 (RF2), Rừng ngẫu nhiên 3 (RF3) với phương pháp xác thực chéo (k-folds = 5)

Bước 3: Lấy kết quả dự đoán của 3 mô hình RF1, RF2, RF3 từ tập dữ liệu xác thực (20%). Xây dựng mô hình rừng ngẫu nhiên (RF) bằng phương pháp học quần thể Stacking từ kết quả dự đoán của 3 mô hình RF1, RF2, RF3 để tạo thành mô hình học quần thể quan điểm về khách sạn (rừng đa tạp quan điểm (F_{MF}) về khách sạn).

Bước 4: So sánh hiệu suất của các mô hình RF1, RF2, RF3 và rừng đa tạp quan điểm (F_{MF}). Kết quả đánh giá rừng đa tạp quan điểm trên tập dữ liệu huấn luyện quan điểm khách sạn có độ chính xác tổng thể là 94% (xem ghi chú Bảng 4.2). Trong tập dữ liệu thử nghiệm, độ chính xác tổng thể là 78%, độ chính xác cụ thể là 74%, độ thu hồi là 62% và F1 là 67% trong Bảng 4.2.

Bảng 4.2: Đánh giá hiệu suất của mô hình rừng đa tạp quan điểm khách sạn

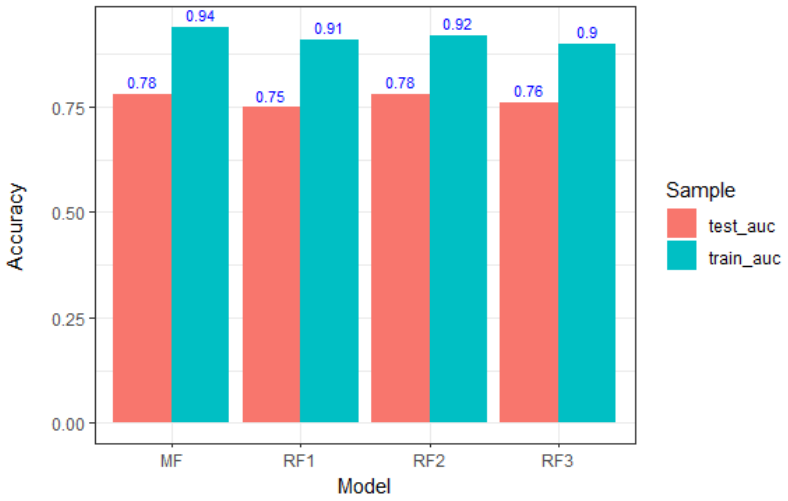
Tập dữ liệu khách sạn	**Độ chính xác**	**Độ chính xác dương (P)**	**Độ bao phủ (R)**	**Trung bình điều hòa F1**
Tập huấn luyện	0,94	–	–	–
Tập kiểm tra	0,78	0,74	0,62	0,67

Ghi chú: Các giá trị trên tập kiểm tra của Bảng 4.2 được dùng để so sánh với các mô hình khác (Bảng 4.4).

Kết quả so sánh được trình bày tại Hình 4.7. Trong đó, đối với tập huấn luyện thì độ chính xác của Random Forest 3 là 0,9 thấp nhất, độ chính xác của Random Forest 2 là 0,92, độ chính xác của Random Forest 1 là 0,91, độ chính xác của Manifold Forest là 0,94, cao nhất. Đối với tập kiểm tra thì độ chính xác của Random Forest 3 là 0,76, độ chính xác của Random Forest 2 là 0,78, độ chính xác của Random Forest 1 là 0,74 thấp nhất, độ chính xác của Manifold Forest là 0,78, cao nhất (bằng Random Forest 2).

Về đóng góp của cơ chế Stacking: trên tập kiểm tra yelpHotelData, rừng thành phần RF2 đạt độ chính xác 0,78, bằng với rừng đa tạp quan điểm F_{MF} (0,78). Như vậy, trên tập dữ liệu này Stacking không vượt được rừng thành phần tốt nhất; vai trò của Stacking thể hiện ở chỗ tự động học cách kết hợp K rừng trên tập xác thực mà không cần biết trước rừng của cụm nào cho kết quả tốt nhất, nhờ đó cho độ chính xác không thấp hơn rừng thành phần tốt nhất (RF2) và cao hơn RF1 (0,74), RF3 (0,76). Để tách bạch riêng đóng góp của Stacking, Phụ lục H thực hiện thí nghiệm loại trừ (ablation) 2×2 theo hai trục là không gian phân cụm (Euclidean so với đa tạp) và cách kết hợp K rừng (trung bình so với Stacking), lặp lại 5 lần trên N = 112.275. Kết quả cho thấy khi cố định không gian phân cụm Euclidean, việc thay kết hợp trung bình bằng Stacking không tạo khác biệt có ý nghĩa thống kê trên tập Hotel (kiểm định

McNemar, p = 0,7556) mà chủ yếu tái phân bổ giữa Recall và Specificity; ngược lại, mọi cặp so sánh khác nhau về không gian phân cụm đều khác biệt có ý nghĩa. Điều này cho thấy yếu tố quyết định sự khác biệt hiệu năng giữa F_{DF} và F_{MF} là không gian phân cụm, còn Stacking đóng vai trò cơ chế kết hợp ổn định, không phải nguồn cải thiện độ chính xác chính.



Hình 4.7: So sánh độ chính xác các mô hình trên bộ dữ liệu yelpHotelData

Lưu ý về mối quan hệ giữa Thuật toán 4.3 và kịch bản thực nghiệm ở trên: Thuật toán 4.3 ở Bước 3 gọi hàm BuildManifoldForest_Phase2 - tức Giai đoạn 2 của Thuật toán 4.1 - xây mỗi rừng thành phần Fk bằng hàm phụ BuildManifoldTree, trong đó tiêu chí tách nút là Entropy đa tạp có trọng số theo ma trận quan hệ W (công thức (4.7) và (4.9)), khác với tiêu chí Entropy nhãn (Gini/Shannon) thông thường. Thực nghiệm trình bày ở trên, tại Bước 2, xây mỗi Fk (RF1, RF2, RF3) là một rừng ngẫu nhiên tiêu chuẩn, sử dụng tiêu chí tách nút thông thường trên nhãn, không dùng trực tiếp Entropy đa tạp có trọng số W tại bước này. Sự thay thế này có cơ sở: ma trận quan hệ đa tạp W đã được khai thác trọn vẹn ở Bước 1 của thực nghiệm ở trên (xây W từ rừng không giám sát, rồi dùng khoảng cách đa tạp làm tiêu chí phân cụm) - mỗi cụm thu được do đó đã là một vùng đồng nhất trên đa tạp. Việc dùng một rừng phân loại có giám sát tiêu chuẩn bên trong từng cụm đã đồng nhất đó cho phép khai thác trực tiếp nhãn cực tính sẵn có mà không cần tính lại Entropy đa tạp cho từng nút con, giảm đáng kể chi phí tính toán so với BuildManifoldTree đầy đủ. Bước 3 của thực nghiệm ở trên (Stacking trên tập xác thực) không thay đổi so với Thuật toán 4.3. Dạng

tổng quát của Thuật toán 4.1 với tiêu chí Entropy đa tạp W-weighted tại mọi nút được giữ nguyên như khung lý thuyết, mở ra hướng thực nghiệm sâu hơn cho các nghiên cứu tiếp theo. Thực nghiệm trên tập dữ liệu nhà hàng yelpResData (mục 4.7.3.2, tiếp theo) được xây dựng theo đúng nguyên tắc tương tự.

4.7.3.2. Mô hình khai phá quan điểm với rừng đa tạp dựa trên bộ năm quan điểm trên tập dữ liệu nhà hàng yelpResData

Mô hình rừng đa tạp quan điểm trên tập dữ liệu quan điểm nhà hàng yelpResData được thực hiện các bước sau:

Bước 1: Chia dữ liệu thành 3 phần: tập dữ liệu huấn luyện (60%), tập dữ liệu xác thực (20%) và tập kiểm tra (20%). Thực hiện phân cụm dữ liệu trên tập huấn luyện. Kết quả có 4 phân cụm.

Bước 2: Xây dựng 4 mô hình rừng ngẫu nhiên độc lập bao gồm RF1, RF2, RF3 và RF4 tương ứng trên 4 phân cụm và thực hiện phương pháp xác thực chéo (k-folds = 10).

Việc chọn k-fold = 5 cho tập khách sạn (mục 4.7.3.1) và k-fold = 10 cho tập nhà hàng ở bước này dựa trên hai căn cứ: thứ nhất, số phân cụm DBSCAN thu được trên mỗi tập (3 cụm với khách sạn, 4 cụm với nhà hàng) khiến cỡ mẫu trung bình mỗi cụm khác nhau dù cả hai tập đều được lấy mẫu về cùng N = 112.275 (mục 3.7.3.1) - tập nhà hàng chia thành nhiều cụm hơn nên mỗi cụm vẫn đủ lớn để chia ổn định thành 10 fold, trong khi tập khách sạn cần giữ k nhỏ hơn để mỗi fold có đủ số mẫu huấn luyện; thứ hai, thời gian huấn luyện khả dụng của từng tập trong phạm vi thực nghiệm. Đây là lựa chọn dựa trên kinh nghiệm và ràng buộc tính toán, luận án chưa thực hiện khảo sát định lượng đầy đủ (ví dụ quét $k \in \{3, 5, 10, 20\}$ và so sánh phương sai kết quả) để xác định giá trị k tối ưu cho từng tập dữ liệu; đây là hướng cần bổ sung ở nghiên cứu tiếp theo, theo các phương pháp tối ưu siêu tham số có hệ thống [95].

Bước 3: Lấy kết quả dự đoán của 4 mô hình RF1, RF2, RF3, RF4 từ tập dữ liệu xác thực (20%). Xây dựng mô hình Random Forest (RF) bằng phương pháp học quần thể Stacking từ kết quả dự đoán của 4 mô hình RF1, RF2, RF3, RF4.

Bước 4: So sánh hiệu suất của các mô hình rừng thành phần với rừng đa tạp quan điểm (F_{MF}). Kết quả đánh giá rừng đa tạp quan điểm trên tập dữ liệu huấn luyện quan điểm nhà hàng có độ chính xác tổng thể là 0,95 (xem ghi chú Bảng 4.3). Trên tập

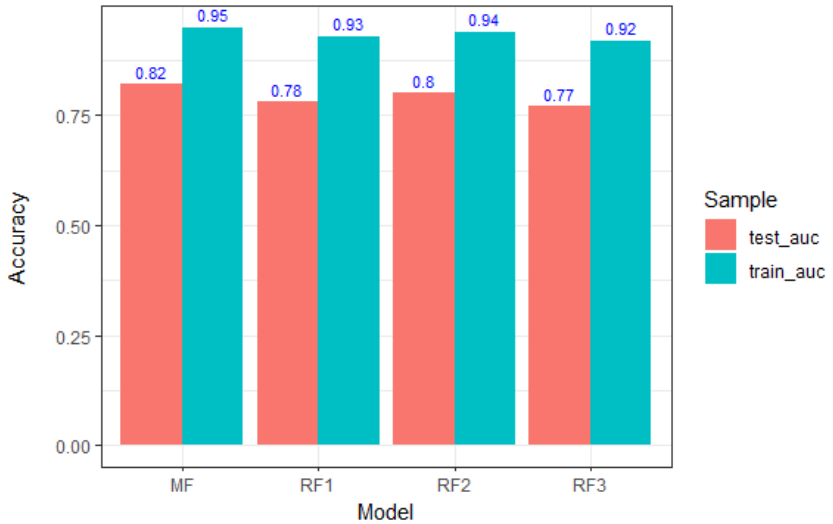
dữ liệu quan điểm kiểm tra, độ chính xác tổng thể là 0,82, độ chính xác cụ thể là 0,75, độ thu hồi là 0,67 và F1 là 0,71 trong Bảng 4.3. Chênh lệch giữa tập huấn luyện và tập kiểm tra (Accuracy 0,94–0,95 so với 0,78–0,82, Bảng 4.2, Bảng 4.3, Hình 4.7, Hình 4.8) phản ánh mức khớp quá mức thường gặp của rừng ngẫu nhiên trên dữ liệu huấn luyện; vì vậy, mọi so sánh với các mô hình khác (Bảng 4.4) chỉ sử dụng giá trị trên tập kiểm tra (Accuracy 0,78 và 0,82, trung bình 0,80).

Bảng 4.3: Đánh giá hiệu suất của mô hình rừng đa tạp quan điểm nhà hàng

Tập dữ liệu nhà hàng	**Độ chính xác**	**Độ chính xác dương (P)**	**Độ bao phủ (R)**	**F1**
Tập huấn luyện	0,95	–	–	–
Tập kiểm tra	0,82	0,75	0,67	0,71

Ghi chú: Các giá trị trên tập kiểm tra của Bảng 4.3 được dùng để so sánh với các mô hình khác (Bảng 4.4).

Kết quả so sánh được trình bày tại Hình 4.8. Trong đó, đối với tập huấn luyện thì độ chính xác của Random Forest 3 là 0,92, thấp nhất, độ chính xác của Random Forest 2 là 0,94, độ chính xác của Random Forest 1 là 0,93, độ chính xác của Manifold Forest là 0,95 cao nhất. Đối với tập kiểm tra thì độ chính xác của Random Forest 3 là 0,77 thấp nhất, độ chính xác của Random Forest 2 là 0,8, độ chính xác của Random Forest 1 là 0,78, độ chính xác của Manifold Forest là 0,82 cao nhất.



Hình 4.8: So sánh độ chính xác các mô hình trên bộ dữ liệu yelpResData

4.8. So sánh tổng hợp

Về đầu vào của các mô hình so sánh trong Bảng 4.4: tương tự phương pháp luận đã trình bày tại §3.7.5, các phương pháp học máy truyền thống dựa trên đặc trưng có cấu trúc (Random Forest, Gradient Boosting, SVM, và ba mô hình rừng đề xuất) sử dụng đặc trưng bộ năm quan điểm đã mã hóa số làm đầu vào; các mô hình học sâu (LSTM, BiLSTM-Attention) và các mô hình Transformer (BERT, RoBERTa) sử dụng câu văn bản đánh giá gốc làm đầu vào theo đúng cách vận hành tiêu chuẩn của các kiến trúc này. Tính công bằng so sánh được đảm bảo ở cấp độ cùng tập mẫu, cùng nhãn tình cảm và cùng tỉ lệ chia huấn luyện/kiểm tra. Phụ lục H đánh giá lại cả ba mô hình F_{CF}, F_{DF}, F_{MF} trên cùng một tỉ lệ chia 60/20/20 nhằm loại trừ hoàn toàn yếu tố khác biệt tỉ lệ chia khi so sánh trực tiếp giữa ba mô hình.

Theo số liệu đã công bố (theo cách chia dữ liệu ngẫu nhiên theo dòng bộ năm), Rừng đa tạp đạt độ chính xác cao nhất trong ba mô hình rừng đề xuất, cao hơn các mô hình học quần thể và học máy cùng tầng (Random Forest 0,76, Gradient Boosting 0,76, SVM 0,75); với đầu vào là văn bản gốc, các mô hình học sâu (LSTM, BiLSTM-Attention) và Transformer (BERT, RoBERTa) đạt độ chính xác trung bình từ 0,74 (LSTM) đến 0,79 (RoBERTa) (Bảng 4.4). So với Rừng mật độ, mức cải thiện trên tập dữ liệu YelpHotelData là +3% (0,78 so với 0,75), trên tập dữ liệu YelpResData là +2% (0,82 so với 0,80), trung bình cải thiện +2% (0,80 so với 0,78). Khi chia theo đánh giá gốc (Phụ lục J), mức cải thiện này không được tái hiện: F_{MF} thấp hơn F_{DF} và Random Forest do bước phân cụm đa tạp chỉ học trên mẫu con. Các lý do có thể của mức cải thiện này: khai thác cấu trúc phi tuyến thông qua học đa tạp, phân cụm trên không gian đa tạp phù hợp hơn so với không gian Euclidean, ma trận quan hệ phản ánh ngữ nghĩa tốt hơn khoảng cách Euclidean đơn thuần. Cả ba mô hình rừng đề xuất đều sử dụng cùng không gian đặc trưng đầu vào bốn chiều (entity, aspect, holder, time) sau khi tách thành phần tình cảm (sentiment) làm nhãn phân loại, với số đặc trưng ngẫu nhiên tại mỗi nút m_try = $\sqrt{4}$ = 2 thống nhất giữa ba chương. So sánh tổng hợp thiết lập các tham số của ba mô hình tại Bảng 4.5.

Bảng 4.4: So sánh Rừng đa tạp với các mô hình chuẩn

Mô hình	**Độ chính xác trên tập test YelpHotel.**	**Độ chính xác trên tập test YelpRes.**	**Độ chính xác Trung bình**	**Khả năng giải thích**	**Độ phức tạp tính toán**
Rừng đa tạp (F_{MF})	0,78	0,82	0,80	Rất cao	Θ(N×logN×(nW+n×d')) + Θ($K \times N_{val}$) (Kết hợp 2 giai đoạn)
Rừng mật độ (F_{DF})	0,75	0,80	0,78	Rất cao	Θ(n×mtry×N×logN) + Θ(n×L×N×logN) (Trung bình)
Rừng phân loại (F_{CF})	0,76	0,77	0,77	Rất cao	Θ(n×N×mtry×logN) (Thấp)
BiLSTM-Attention [88]	0,76	0,78	0,77	Thấp	Θ(n×L×d) (Trung bình)
Random Forest [65]	0,74	0,77	0,76	Cao	Θ(n×N×mtry×logN) (Thấp)
Gradient Boosting [84]	0,74	0,78	0,76	Cao	Θ(n×N×logN) (Trung bình)
SVM [86]	0,70	0,80	0,75	Trung bình	Θ(N^2~N^3) (Cao)
LSTM [85]	0,71	0,76	0,74	Thấp	Θ(n×L×d^2) (Trung bình)
RoBERTa [83]	0,74	0,83	0,79	Rất thấp	Θ(L^2×d) cho mỗi token (Rất cao - GPU)
BERT [81]	0,74	0,82	0,78	Rất thấp	Θ(L^2×d) cho mỗi token (Rất cao - GPU)

Ghi chú: số liệu của rừng mật độ và rừng đa tạp theo Bài báo 5 và Bài báo 6. Các mô hình học sâu (LSTM, BiLSTM-Attention) và Transformer (BERT, RoBERTa) sử dụng đầu vào là văn bản đánh giá gốc nên được trình bày như mốc tham chiếu. Mọi giá trị trong bảng là giá trị trên tập kiểm tra (test), không dùng giá trị trên tập huấn luyện. Vì dữ liệu mất cân bằng (khoảng 63–65% mẫu positive) và các mô hình có đầu vào khác nhau, chênh lệch Accuracy giữa các mô hình chỉ có ý

nghĩa tham khảo; phép so sánh trên cùng tập mẫu, cùng cách chia và cùng độ đo Balanced Accuracy được trình bày tại Phụ lục J (Bảng J.8).

Ghi chú: các mô hình trong Bảng 4.4 (bao gồm F_{CF}) đều được huấn luyện và đánh giá trên hai tập dữ liệu YelpHotelData và YelpResData (Chương 3, 4), khác với thực nghiệm minh họa ở Chương 2 vốn thực hiện trên Amazon Food Reviews và TripAdvisor Hotel Reviews. Độ chính xác 0,76–0,77 của F_{CF} tại Bảng 4.4 là kết quả khi áp dụng lại kiến trúc rừng phân loại (Chương 2) trên dữ liệu Yelp, không phải trích dẫn số liệu Amazon/TripAdvisor của Chương 2.

Mô hình F_{MF} có độ phức tạp huấn luyện Θ(N×logN×(nW+n×d')) + $\Theta(K \times N_{val})$ gồm 2 giai đoạn là xây dựng ma trận đa tạp W và xây rừng đa tạp, nhưng giai đoạn xây dựng ma trận đa tạp W chỉ chạy một lần và được chia sẻ. Mô hình F_{DF} thêm chi phí DBSCAN tại nút lá và mô hình F_{CF} là đơn giản nhất. Cả ba mô hình đề xuất không cần GPU; BERT/RoBERTa có độ phức tạp Θ(L^2×d) và thường được huấn luyện trên phần cứng chuyên dụng.

Bảng 4.5: So sánh thiết lập tham số ba mô hình rừng

Tham số	**Ch.2 — F_{CF}**	**Ch.3 — F_{DF}**	**Ch.4 — F_{MF}**
Số cây (n)	100	100 (mỗi cụm)	100 (mỗi rừng $\mathcal{F}_k$)
m_{try}	$2\left(=\left\lfloor\sqrt{4}\right\rfloor\right)$	$2\left(=\left\lfloor\sqrt{4}\right\rfloor\right)$	$2\left(=\left\lfloor\sqrt{4}\right\rfloor\right)$
Hàm Entropy	Entropy $H(O)$	Entropy $H_D(O)$	Entropy $H_{\mathcal{M}}(O)$
Tiêu chí tách	IG	IG_D	$IG_{\mathcal{M}}$
Khoảng cách	-	Euclidean	$dist_{\mathcal{M}} = \sqrt{1 - W_{ij}}$
DBSCAN ε	-	0,5	0,3
DBSCAN MinPts	-	5	5
Số cụm K	-	7 (Hotel), 9 (Rest.)	3 (Hotel), 4 (Rest.)
Tỷ lệ train/test	80% / 20%	80% / 20%	60% / 20% / 20%
Meta-learner	Majority Voting (Biểu quyết đa số)	Trung bình mật độ	Stacking (Random Forest)
class_weight	balanced	balanced	balanced

Mở rộng phân tích từ Bảng 4.2 và Bảng 4.3, kết quả huấn luyện mô hình cho thấy rừng đa tạp quan điểm F_{MF} đạt độ chính xác tập huấn luyện cao trên cả hai miền dữ liệu (0,94 trên tập dữ liệu yelpHotelData và 0,95 trên tập dữ liệu yelpResData), trong

khi độ chính xác tập kiểm tra giảm xuống còn 0,78 và 0,82 tương ứng. Khoảng cách giữa hai tập (0,16 và 0,13) phản ánh mức khớp quá mức trên tập huấn luyện của các rừng thành phần (xem Bước 4, mục 4.7.3.2); Meta-learner được học trên tập xác thực tách riêng nên không làm tăng khoảng cách này. Vì vậy mọi so sánh chỉ dựa trên tập kiểm tra, trong đó mô hình F_{MF} đạt độ chính xác bằng hoặc cao hơn các rừng thành phần được báo cáo tại Hình 4.7 và Hình 4.8 (RF1, RF2, RF3) trên cả hai bộ dữ liệu (trên yelpHotelData, RF2 đạt 0,78, bằng rừng đa tạp - xem mục 4.7.3.1).

So sánh hai miền dữ liệu, mức cải thiện (+3% trên yelpHotelData và +2% trên yelpResData) của mô hình rừng đa tạp F_{MF} so với mô hình rừng mật độ F_{DF} nhất quán trên cả hai tập (trong Bảng 4.4), gợi ý giả thuyết đa tạp phù hợp với cả hai miền dữ liệu; lưu ý mức cải thiện này thu được theo quy trình chia theo dòng của các bài báo và không được tái hiện khi chia theo đánh giá gốc (Phụ lục J, Bảng J.11, J.15). Đồng thời, số lượng phân cụm trên không gian đa tạp (bao gồm 3 cụm với tập dữ liệu yelpHotelData và 4 cụm với tập dữ liệu yelpResData) thấp hơn đáng kể so với số cụm DBSCAN nguyên thủy ở Chương 3 (bao gồm 7 và 9 cụm tương ứng), gợi ý không gian đa tạp biểu diễn cấu trúc dữ liệu gọn hơn không gian đặc trưng gốc.

Như vậy, chuỗi tiến triển từ mô hình rừng phân loại F_{CF} đến mô hình rừng mật độ quan điểm F_{DF} và mô hình rừng đa tạp quan điểm F_{MF} cho thấy việc khai thác dần các cấu trúc ẩn của dữ liệu, từ phân loại trực tiếp, đến mật độ cụm, đến hình học đa tạp, đem lại cải thiện hiệu năng theo số liệu đã công bố (chia theo dòng) và tăng dần khả năng diễn giải; khi chia theo đánh giá gốc (Phụ lục J), mức cải thiện về độ chính xác không còn rõ rệt, nên giá trị chính của chuỗi mô hình nằm ở khả năng diễn giải theo cấu trúc dữ liệu (cụm, đa tạp), trong khi độ chính xác của F_{CF}, F_{DF} tương đương các mô hình học máy dùng cùng bộ năm.

4.9. Công cụ sử dụng

Thực nghiệm của chương được triển khai thống nhất bằng ngôn ngữ R phiên bản 4.3.2 (RStudio 2023.12.1), dùng chung môi trường và các gói đã nêu tại mục 3.8. Mô hình rừng đa tạp quan điểm F_{MF} sử dụng randomForest cho rừng không giám sát (phân biệt dữ liệu thật/giả) để xây dựng ma trận quan hệ đa tạp W và cho các rừng thành phần theo cụm, dbscan cho phân cụm trên khoảng cách đa tạp, rpart và tree cho cây quyết định, randomForest cho bộ kết hợp Stacking trên tập xác thực; ước lượng

chiều nội tại dùng gói intrinsicDimension; xử lý dữ liệu và trực quan hóa dùng dplyr, tidyr, data.table, ggplot2. Các mô hình so sánh trong Bảng 4.4 dùng cùng các gói như mục 3.8. Danh sách các thư viện và phiên bản được trình bày tại Phụ lục D.

4.10. Kết luận chương

Chương 4 đã đề xuất một tiếp cận mới là mô hình khai phá quan điểm với rừng đa tạp nhằm giải quyết vấn đề ảnh hưởng của việc phân cụm quan điểm theo mối quan hệ phi tuyến giữa các cụm quan điểm khác nhau. Điểm đặc trưng chính của mô hình nằm ở khả năng phát hiện và khai thác cấu trúc đa tạp chiều thấp, thông qua việc xây dựng ma trận quan hệ đa tạp bằng rừng quyết định không giám sát. Khác với rừng mật độ sử dụng khoảng cách Euclidean, rừng đa tạp quan điểm mô hình hóa mối quan hệ giữa các quan điểm trên đa tạp thông qua ma trận quan hệ W, cho phép phát hiện các cụm có cấu trúc phi tuyến phức tạp. Kết quả thực nghiệm trên tập dữ liệu *YelpHotelData* và *YelpResData* cho thấy rừng đa tạp quan điểm đạt độ chính xác trung bình 0,80, cao hơn rừng mật độ (0,78, +2%) và bằng hoặc cao hơn các rừng thành phần được báo cáo tại Hình 4.7 và Hình 4.8 (mục 4.7.3.1 và 4.7.3.2) và với tập dữ liệu *YelpHotelData* đạt 0,78 (K=3 cụm) và *YelpResData đạt 0,82* (K=4 cụm). Các số liệu này theo quy trình chia ngẫu nhiên theo dòng bộ năm của các bài báo đã công bố; khi chia theo đánh giá gốc (Phụ lục J, Bảng J.11, J.15), F_{MF} thấp hơn Random Forest khoảng 0,02–0,03 Balanced Accuracy do bước phân cụm đa tạp chỉ học trên mẫu con 5.000 dòng. Chương này cũng nhằm giải quyết các vấn đề hạn chế của không gian Euclidean tuyến tính trong việc mô hình hóa mối quan hệ phức tạp giữa các cụm quan điểm, cho thấy khai thác cấu trúc đa tạp phi tuyến giúp hiểu sâu hơn về bản chất dữ liệu và, theo số liệu đã công bố, cải thiện độ chính xác so với rừng mật độ trong khai phá quan điểm. Mô hình rừng đa tạp quan điểm đề xuất khai thác giả thuyết đa tạp trong dữ liệu quan điểm, một hướng nghiên cứu mới so với các phương pháp học quần thể truyền thống. So với các phương pháp học sâu như BERT [81], RoBERTa [83] và các mô hình LLM gần đây [127][128][135], rừng đa tạp quan điểm là một hướng tiếp cận khác, chú trọng khả năng diễn giải và chi phí tính toán thấp, phù hợp với môi trường tài nguyên hạn chế. Hạn chế chính của F_{MF} là ma trận quan hệ W có kích thước N×N, đòi hỏi lấy mẫu con khi dữ liệu lớn (Phụ lục G); hạn chế này cùng các kết quả của ba chương được tổng hợp tại phần Kết luận và hướng phát triển.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết luận

Khai phá quan điểm luôn là vấn đề đáng quan tâm và có nhiều ứng dụng trong kinh doanh, giải trí, chính phủ, giáo dục và xã hội… Nhiều nghiên cứu liên quan cũng đã đưa ra các giải pháp nhằm cải thiện độ chính xác dự đoán và khả năng giải thích của mô hình. Các phương pháp học quần thể cho kết quả khả quan, có hiệu suất dự đoán tốt hơn so với học máy riêng lẻ. Đồng thời, các phương pháp học quần thể có khả năng tổng quát hóa và cấu trúc linh hoạt. Hiện nay, các phương pháp khai phá quan điểm chưa quan tâm, tập trung vào phân loại quan điểm theo đa số/số đông và vẫn đang còn là một hướng mở. Bên cạnh đó, việc tiếp cận bộ phân loại cơ sở giống nhau như cây quyết định có khá nhiều ưu điểm trong việc mô phỏng cấu trúc phân cấp quan điểm.

Luận án "Khai phá quan điểm với các phương pháp quần thể" tập trung vào việc đề xuất và thử nghiệm các giải pháp nhằm cải thiện hiệu quả, hiệu suất, tăng cường khả năng diễn giải của mô hình khai phá quan điểm. Kết quả đạt được của luận án đã đề cập đến vấn đề có liên quan đến xây dựng tập dữ liệu quan điểm và đề xuất mô hình khai phá quan điểm mới, cụ thể là khai phá quan điểm với các phương pháp quần thể. Kết quả chính của luận án (các giá trị độ chính xác dưới đây là số liệu đã công bố, tính theo quy trình chia ngẫu nhiên theo dòng bộ năm; kết quả chia theo đánh giá gốc được tóm tắt ở cuối phần này), cụ thể là:

- Đề xuất mô hình khai phá quan điểm với rừng phân loại nhằm giải quyết vấn đề ảnh hưởng của kết hợp theo tầm quan trọng của các nhóm/lớp quan điểm khác nhau và diễn giải mô hình, cụ thể: độ chính xác (Accuracy) là 0,77 trên tập dữ liệu yelpHotelData và yelpResData. Diễn giải tốt qua tầm quan trọng đặc trưng và PDP.

- Đề xuất mô hình khai phá quan điểm với rừng mật độ để giải quyết các vấn đề ảnh hưởng kết hợp theo tầm quan trọng của cụm quan điểm khác nhau, cho thấy mô hình hóa cấu trúc dữ liệu giúp giải thích kết quả theo cụm quan điểm, cụ thể: độ chính xác trung bình (Average Accuracy) là 0,78 trên tập dữ liệu yelpHotelData và yelpResData, không cần GPU (ba mô hình rừng đề xuất dùng đặc trưng bộ năm đã mã hóa số làm đầu vào, trong khi các mô hình học sâu dùng trực tiếp văn bản đánh

giá gốc; các mô hình này được trình bày như mốc tham chiếu (xem §4.8 và Phụ lục J). Đồng thời diễn giải chỉ số SHAP đã xác định thực thể (Entity) là đặc trưng chủ đạo (trung bình |SHAP| ≈ 0,087, Hình 3.5).

- Đề xuất mô hình khai phá quan điểm với rừng đa tạp để giải quyết vấn đề ảnh hưởng của kết hợp theo mối quan hệ của các tập hợp/cụm quan điểm khác nhau bằng cách mô hình hóa mối quan hệ phức tạp giữa các cụm quan điểm, khai thác cấu trúc đa tạp phi tuyến để hiểu sâu hơn về bản chất của dữ liệu nhằm cải thiện độ chính xác và hiệu năng mô hình khai phá quan điểm, cụ thể: Mô hình xây ma trận quan hệ W, phân cụm DBSCAN trên không gian đa tạp ($d' < 5$) và tổng hợp qua Stacking. Kết quả độ chính xác trung bình (Average Accuracy) là 0,80.

- Đề xuất chuyển đổi tập dữ liệu quan điểm thành bộ năm quan điểm từ hai bộ dữ liệu chuẩn về nhà hàng và khách sạn (*yelpHotelData và yelpResData*) và hai tập dữ liệu chuẩn về nhà hàng và khách sạn (*tập dữ liệu VLSP-2018*) được sử dụng để thử nghiệm và đánh giá mô hình đề xuất.

- Ngoài ra, luận án còn đề xuất và xây dựng mô hình trích xuất bộ năm quan điểm có cấu trúc *(e, a, s, h, t)* bao gồm thực thể, khía cạnh, tình cảm, người giữ quan điểm và thời gian. Nghiên cứu mối quan hệ giữa các đặc trưng này để thiết lập biểu diễn dữ liệu phù hợp cho các phương pháp học quần thể.

Chuỗi tiến triển độ chính xác từ mô hình rừng phân loại quan điểm đến rừng mật độ quan điểm và rừng đa tạp quan điểm phản ánh việc khai thác cấu trúc dữ liệu ngày càng tinh tế, trong đó, trên cùng hai tập dữ liệu Yelp, độ chính xác trung bình theo số liệu công bố là 0,78 (rừng mật độ) và 0,80 (rừng đa tạp), đồng thời cả ba mô hình duy trì khả năng diễn giải qua cấu trúc cây, cụm và ma trận quan hệ.

Khi đánh giá lại theo quy trình chia theo đánh giá gốc (mọi dòng bộ năm của một đánh giá chỉ nằm trong một tập; Phụ lục J, Bảng J.11–J.17, 5 hạt giống × 10.000 đánh giá), Balanced Accuracy của mọi mô hình dùng bộ năm giảm về khoảng 0,51–0,57 trên cả hai tập dữ liệu: F_{CF} và F_{DF} tương đương Random Forest (chênh lệch không có ý nghĩa thống kê) và thấp hơn SVM tuyến tính khoảng 0,01; F_{MF} thấp hơn Random Forest khoảng 0,02–0,03 do bước phân cụm đa tạp chỉ học trên mẫu con. Các mô hình đọc văn bản gốc (BERT, RoBERTa) đạt 0,76–0,82, nhưng khi mọi mô hình chỉ dùng bộ năm, F_{CF} và F_{DF} tương đương BiLSTM-Attention và cao hơn về

trung bình so với LSTM, BERT, RoBERTa. Như vậy, đóng góp của ba mô hình rừng nằm chủ yếu ở cách biểu diễn bộ năm có cấu trúc và khả năng diễn giải; độ chính xác bị giới hạn bởi lượng thông tin của bốn thành phần (e, a, h, t).

Luận án đóng góp vào xu hướng nghiên cứu về học quần thể cho NLP [59] và XAI cho học máy [125]. So với các mô hình ABSA tiên tiến nhất dựa trên mạng nơ-ron đồ thị và kiến trúc tổng hợp ngữ pháp [136], kiến trúc Transformer như BERT, RoBERTa và các mô hình ngôn ngữ lớn (LLM) [81] [83] [127], các mô hình rừng học quần thể được đề xuất trong luận án có ba điểm khác biệt được thảo luận như sau:

Thứ nhất, về độ chính xác trên dữ liệu lớn có cấu trúc rõ ràng (structured/tabular), các mô hình dựa trên cây quyết định và rừng là lựa chọn phù hợp, đặc biệt khi dữ liệu được biểu diễn dưới dạng đặc trưng rời rạc như bộ năm quan điểm, kết quả này phù hợp với phát hiện gần đây cho thấy các mô hình dựa trên cây vẫn là lựa chọn mạnh trên dữ liệu dạng bảng [124]. Tuy nhiên, kết quả chia theo đánh giá gốc tại Phụ lục J cho thấy trên cùng bộ năm, các mô hình tuyến tính (SVM, Logistic Regression) có thể đạt kết quả tương đương hoặc nhỉnh hơn các mô hình rừng.

Thứ hai, về khả năng diễn giải, các mô hình rừng đề xuất có thế mạnh rõ rệt nhờ cấu trúc cây minh bạch kết hợp SHAP (trong Chương 3) và ma trận quan hệ đa tạp tường minh (trong Chương 4), trong khi các mô hình BERT/LLM thường được xem là hộp đen, đòi hỏi kỹ thuật diễn giải hậu kiểm phức tạp hơn [89][90]. Các mô hình rừng đề xuất cung cấp sự cân bằng giữa độ chính xác và khả năng giải thích, phù hợp với xu hướng mô hình AI có trách nhiệm (Responsible AI) [139] đang được cộng đồng khoa học và công nghiệp quan tâm.

Thứ ba, về chi phí tính toán và tài nguyên, các mô hình rừng đề xuất không yêu cầu GPU chuyên dụng hay tập dữ liệu huấn luyện quy mô lớn như các mô hình ngôn ngữ lớn, phù hợp với các bài toán ABSA quy mô vừa và nhỏ trong thực tế doanh nghiệp.

Tuy nhiên, luận án cũng nhận thức rõ hạn chế của hướng tiếp cận học quần thể so với học sâu là các mô hình rừng đề xuất xử lý đặc trưng đã được trích xuất sẵn bộ năm quan điểm, do đó phụ thuộc vào chất lượng của bước trích xuất ban đầu, trong

khi các mô hình end-to-end dựa trên Transformer có khả năng học biểu diễn ngữ nghĩa trực tiếp từ văn bản thô, nắm bắt tốt hơn các hiện tượng ngôn ngữ phức tạp như mỉa mai, phủ định hoặc quan điểm ẩn [131][132]. Đây là định hướng được đề cập tại Hướng phát triển (Mục 5.2), kết hợp biểu diễn ngữ nghĩa từ mô hình học sâu với cấu trúc cây/rừng để vừa giữ được khả năng diễn giải, vừa cải thiện chất lượng trích xuất đặc trưng đầu vào.

Bên cạnh hạn chế phụ thuộc vào bước trích xuất bộ năm, luận án nhận thức thêm năm giới hạn có tính hệ thống của hướng tiếp cận học quần thể dựa trên rừng trong bài toán khai phá quan điểm:

Thứ nhất, chuẩn hóa ngữ nghĩa và tính linh hoạt của ngôn ngữ tự nhiên. Việc ép ngôn ngữ tự do vào cấu trúc rời rạc *(e, a, s, h, t)* là sự đánh đổi có chủ ý: giúp mô hình rừng khai thác quan hệ đa chiều giữa các thành phần, nhưng đồng thời có thể làm mất thông tin ngữ cảnh phong phú vốn có trong câu tự nhiên. Đặc biệt, ngữ cảnh mơ hồ (holder không rõ), aspect xuất hiện không đúng nhãn/miền, hay ẩn dụ/đảo chiều quan điểm có thể khiến mô hình thiên về dự đoán theo hình thức hơn là ngữ nghĩa thực sự. Đây là đánh đổi mang tính nguyên lý giữa tính cấu trúc và tính linh hoạt trong xử lý ngôn ngữ.

Thứ hai, khả năng tổng quát hóa theo miền còn hạn chế. Các thực nghiệm trong luận án tập trung vào dữ liệu đánh giá thương mại như nhà hàng, khách sạn, thực phẩm, các lĩnh vực có mẫu biểu đạt tương đối quen thuộc với bộ từ vựng khía cạnh ổn định. Khi chuyển sang miền khác như y tế, giáo dục hay chính trị có cách diễn đạt quan điểm khác biệt căn bản nên bộ từ khía cạnh và phân phối nhãn có thể thay đổi đáng kể, dẫn đến giảm chất lượng mô hình. Thực nghiệm trên VLSP-2018 tiếng Việt tại Phụ lục F là bước đầu kiểm chứng tính tổng quát, nhưng cần mở rộng thêm sang các miền và ngôn ngữ đa dạng hơn.

Thứ ba, bài toán hiện tại giới hạn ở phân loại cực tính nhị phân (positive/negative), chưa nắm bắt được cường độ cảm xúc. Trên thực tế, sự khác biệt giữa "rất hài lòng" và "hài lòng", hay giữa "hơi thất vọng" và "rất tệ", mang giá trị thực tiễn cao trong các ứng dụng ra quyết định kinh doanh. Phân loại đa lớp như: very positive/positive/neutral/negative/very negative hoặc hồi quy cường độ là hướng mở

rộng tự nhiên, đặc biệt khi tập dữ liệu VLSP-2018 trong Phụ lục F đã hỗ trợ nhãn 3 lớp, tạo tiền đề cho bước phát triển này.

Thứ tư, độ phức tạp hệ thống do kiến trúc pipeline nhiều bước. Hệ thống gồm nhiều thành phần nối tiếp bao gồm tiền xử lý, phân tích phụ thuộc, trích xuất bộ năm, đến mô hình rừng, khiến lỗi từ bước trước có thể lan truyền và tích lũy tại bước sau, đồng thời gây khó khăn cho việc tối ưu end-to-end. Để giảm thiểu, luận án đã bổ sung ngưỡng tin cậy τ tại bước lọc bộ năm (Thuật toán 1, §1.6) nhằm chỉ truyền các bộ năm có độ tin cậy cao sang tầng phân loại. Tuy nhiên, việc tối ưu toàn cục cả pipeline vẫn là thách thức mở và hướng kết hợp với kiến trúc end-to-end được đề xuất tại §5.2 là giải pháp dài hạn.

Thứ năm, giải thích theo đặc trưng chưa phải là giải thích nhân quả. Các công cụ diễn giải được sử dụng trong luận án (tầm quan trọng đặc trưng, PDP, giá trị SHAP, chỉ số tương tác Shapley và ma trận quan hệ W) cho biết đặc trưng nào đóng góp nhiều vào dự đoán của mô hình và theo chiều nào, tức là giải thích hành vi của mô hình trên dữ liệu quan sát được, không khẳng định quan hệ nhân quả giữa các thành phần bộ năm và cực tính cảm xúc. Chẳng hạn, việc Entity có trung bình |SHAP| lớn nhất (Hình 3.5) cho thấy mô hình dựa nhiều vào thông tin thực thể, nhưng không chứng minh rằng bản thân thực thể là nguyên nhân khiến đánh giá tích cực hay tiêu cực; mối liên hệ này có thể đến từ các yếu tố gây nhiễu như chất lượng dịch vụ, phân khúc giá hay đặc điểm người đánh giá. Ngoài ra, khi các đặc trưng tương quan với nhau, giá trị SHAP có thể phân bổ đóng góp giữa chúng theo cách không phản ánh cơ chế thực. Vì vậy, các kết quả diễn giải trong luận án được hiểu là giải thích mô hình (model explanation), phục vụ kiểm tra tính hợp lý và phát hiện thiên lệch; suy luận nhân quả đòi hỏi thiết kế riêng (thực nghiệm can thiệp, mô hình nhân quả có cấu trúc hoặc phân tích phản thực tế) và được nêu tại hướng phát triển (§5.2).

Năm hạn chế trên là đặc điểm chung của các hệ thống ABSA dạng pipeline dựa trên học máy kinh điển, không đặc thù cho mô hình rừng. Trong phạm vi của luận án, các mô hình đề xuất (F_{CF}, F_{DF}, F_{MF}) cung cấp sự cân bằng giữa độ chính xác (tương đương các mô hình học máy dùng cùng bộ năm khi chia theo đánh giá gốc), khả năng diễn giải và chi phí tính toán (riêng F_{MF} có chi phí cao hơn do ma trận W), phù hợp với bối cảnh ứng dụng thực tế có tài nguyên hạn chế và yêu cầu tính minh bạch của

hệ thống AI [139], hướng đến triển khai trong các cơ quan, tổ chức, doanh nghiệp vừa và nhỏ phù hợp với bối cảnh tại Việt Nam.

5.2. Hướng phát triển

Ngoài các kết quả đạt được, luận án có thể tiếp tục mở rộng các hướng nghiên cứu sau:

Phát triển tập dữ liệu *yelpHotelData* và *yelpResData* bộ năm về quan điểm theo hướng đa miền mở rộng thành tập dữ liệu đa miền, đa ngôn ngữ, bổ sung tiếng Việt từ VLSP-2018 và các ngôn ngữ khác.

Xây dựng bộ năm quan điểm dựa trên Sentiment Treebank, phát triển các thuật toán phân cụm, dựng cây, dựng rừng khác nhau để khai phá quan điểm. Đồng thời phát triển mô hình khai phá quan điểm theo hướng diễn giải để tăng tính hiệu quả và độ tin cậy, chắc chắn của mô hình.

Kết hợp với biểu diễn ngữ nghĩa của các mô hình học sâu (BERT embedding, PhoBERT cho tiếng Việt) với cấu trúc rừng để cải thiện chất lượng trích xuất bộ năm và xử lý mỉa mai, phủ định, quan điểm ẩn.

Mở rộng sang phân tích quan điểm theo thời gian thực trên dữ liệu streaming và bài toán ABSA đa ngôn ngữ.

Kết hợp diễn giải mô hình với suy luận nhân quả (phân tích phản thực tế, mô hình nhân quả có cấu trúc) để phân biệt các yếu tố thực sự tác động đến cực tính cảm xúc với các tương quan trong dữ liệu quan sát.

Bên cạnh các hướng trên, luận án ghi nhận thêm một số hạn chế về mặt thực nghiệm và phương pháp luận cần được bổ sung ở các nghiên cứu tiếp theo:

Thứ nhất, về cách chia dữ liệu, việc chia ngẫu nhiên theo dòng bộ năm (thay vì theo đánh giá gốc hoặc theo thực thể) có thể khiến các dòng sinh ra từ cùng một đánh giá xuất hiện đồng thời ở tập huấn luyện và tập kiểm tra, dẫn đến các chỉ số đánh giá có phần lạc quan; phương án chia theo nhóm (group-based split theo đánh giá gốc) đã được áp dụng trong thực nghiệm so sánh thống nhất tại Phụ lục J (Bảng J.8) để đối chiếu; chia theo thực thể vẫn là hướng thử nghiệm tiếp theo. Kiểm tra có kiểm soát tại Phụ lục J (mục J.11, Bảng J.10) cho thấy, với cùng mô hình Random Forest và cùng 112.275 bộ năm, Accuracy thay đổi từ 0,611 (chia theo đánh giá gốc) đến 0,946 (chia theo dòng khi phần lớn dòng kiểm tra có dòng cùng đánh giá trong

tập huấn luyện); vì vậy các chỉ số tại Bảng 3.7- Bảng 3.9 và Bảng 4.4 cần được hiểu là kết quả theo quy trình chia theo dòng của các bài báo, còn kết quả theo quy trình chia theo đánh giá tại Phụ lục J (Bảng J.11) phản ánh sát hơn khả năng dự đoán cho đánh giá mới.

Thứ hai, đặc trưng định danh có lực lượng lớn (tên thực thể, người đánh giá) hiện được mã hóa số thứ tự; cách mã hóa target encoding (làm trơn Laplace, m = 10) đã được áp dụng trong các thực nghiệm bổ sung tại Phụ lục G - Phụ lục J; thực nghiệm so sánh ba cách mã hóa (số thứ tự, target encoding, one-hot) với F_{CF} tại Phụ lục I (mục I.3) cho thấy trên yelpHotelData, mã hóa target encoding cho Balanced Accuracy cao nhất (0,6303 so với 0,5765 của mã hóa số thứ tự và 0,5184 của mã hóa one-hot), còn trên VLSP-2018 ba cách mã hóa cho kết quả gần như nhau; việc kiểm tra tính ổn định của kết quả khi hoán vị bảng mã vẫn là hướng thử nghiệm cần thiết để loại trừ khả năng mô hình học một thứ tự nhân tạo không có ý nghĩa.

Thứ ba, về độ tin cậy thống kê của kết quả so sánh: do chênh lệch hiệu năng giữa ba mô hình rừng đề xuất chỉ ở mức 1–3%, luận án đã bổ sung kiểm định ý nghĩa thống kê (McNemar trên dự đoán cặp đôi và Wilcoxon signed-rank trên 5 lần chia dữ liệu với các hạt giống khác nhau), lặp lại thực nghiệm với 5 cặp hạt giống ngẫu nhiên khác nhau kèm độ lệch chuẩn, và thí nghiệm loại trừ (ablation) tách bạch đóng góp của không gian phân cụm (Euclidean/đa tạp) và phương pháp kết hợp K rừng (trung bình/Stacking), kết quả chi tiết tại Phụ lục H. Đồng thời, so sánh trực tiếp Ablation_Manifold_Avg (phân cụm đa tạp và kết hợp trung bình) với F_{MF} (phân cụm đa tạp và Stacking) tại Phụ lục H cho phép định lượng riêng phần đóng góp của cơ chế Stacking, làm rõ thêm câu hỏi tại mục 4.7.3.1 về việc RF2 đạt độ chính xác tương đương F_{MF}. Việc lựa chọn k-fold = 5 cho tập khách sạn và k-fold = 10 cho tập nhà hàng ở thực nghiệm minh họa tại Chương 4 hiện dựa trên kích thước mẫu và thời gian huấn luyện khả dụng của từng tập dữ liệu; cơ sở lựa chọn này cần được trình bày rõ hơn ở các nghiên cứu tiếp theo.

Thứ tư, về phương án cài đặt thực tế của ma trận quan hệ đa tạp $W \in \mathbb{R}^{N\times N}$ khi N ở quy mô lớn (ví dụ tập yelpResData), các phương án giảm chi phí bộ nhớ và thời gian (lấy mẫu con, biểu diễn thưa theo chỉ số nút lá, tính W cục bộ trong từng nút hoặc từng cụm) đã được phân tích tại §4.5.1 và các thực nghiệm tại Phụ lục G,

Phụ lục H dùng lấy mẫu con N = 5.000 cho bước tính W; phần phân tích độ phức tạp $\Theta(N \cdot \log N \cdot (nW + n \cdot d'))$ cũng cần được hiệu chỉnh cho tương thích với phương án cài đặt thực tế, tương tự đối với việc tính khoảng cách $dist_{\mathcal{M}}$ dùng cho DBSCAN trên đa tạp.

Thứ năm, entropy Gaussian được luận án sử dụng làm tiêu chí phân tách vốn được định nghĩa cho biến liên tục; khi áp dụng cho các đặc trưng định danh đã mã hóa số thứ tự, cách diễn giải ma trận hiệp phương sai $\Lambda(O)$ và cách xử lý trường hợp $|\Lambda(O)|$ tiến về 0 tại các nút lá có ít mẫu hoặc có cột hằng cần được làm rõ và kiểm chứng thêm bằng thực nghiệm. Tương tự, giả thuyết chiều nội tại đa tạp $d' < 5$ đã được ước lượng trực tiếp bằng hai phương pháp ước lượng độc lập MLE [77] và DANCo [209] (kèm một xấp xỉ tham khảo Local-PCA), cho kết quả $d' \approx 2{,}78$ (tập Hotel) và $d' \approx 3{,}66$ (tập Restaurant) trên không gian đặc trưng dùng xây ma trận W, ủng hộ giả thuyết đa tạp trên Hotel, còn trên tập dữ liệu Restaurant thì phương pháp DANCo cho $d' = 4$ bằng số chiều quan sát nên bằng chứng chưa nhất quán; xem chi tiết tại Phụ lục I.

Thứ sáu, về xử lý mất cân bằng nhãn: độ đo Recall trên tập kiểm tra của các mô hình rừng đề xuất hiện duy trì ở mức 0,62–0,76; các bảng kết quả chính của Chương 3, 4 chỉ báo cáo Accuracy, Precision, Recall và F1 theo số liệu đã công bố. Thực nghiệm bổ sung so sánh năm biến thể xử lý mất cân bằng lớp (Baseline, ThresholdOnly, ClassWeight, BalancedBoot, SMOTE) trên Kịch bản 1 (F_{DF}), thực hiện riêng trên tập Hotel và tập Restaurant, kèm đầy đủ F1-macro, Balanced Accuracy, MCC và PR-AUC bên cạnh Accuracy, đã được thực hiện và trình bày chi tiết tại Phụ lục G. Cuối cùng, việc đánh giá lại ba mô hình F_{CF}, F_{DF}, F_{MF} trên cùng một tập dữ liệu (N = 112.275), cùng tỉ lệ chia 60/20/20 và cùng quy trình (thay vì tỉ lệ 80/20 ở Chương 2, 3 và 60/20/20 ở Chương 4 trong các công bố) đã được thực hiện tại Phụ lục G, Phụ lục H và đối chiếu với số liệu đã công bố tại Phụ lục J; các hướng nghiên cứu còn mở rộng nêu trên sẽ được nghiên cứu sinh tiếp tục hoàn thiện ở các nghiên cứu tiếp theo.

DANH MỤC CÁC CÔNG TRÌNH KHOA HỌC ĐÃ CÔNG BỐ

TT	Tên công trình	Thể loại	Năm công bố
1	Bài báo số 1	Tạp chí Quốc tế International Journal of Advanced Computer Science and Applications (IJACSA)	2020
2	Bài báo số 2	Hội nghị Quốc tế The EAI International Conference on Context-Aware Systems and Applications (ICCASA)	2021
3	Bài báo số 3	Hội nghị Quốc tế The International Conference on Intelligent Information Technology (ICIIT)	2023
4	Bài báo số 4	Tạp chí Quốc tế Journal of EAI Endorsed Transactions on Context-aware Systems and Applications	2023
5	Bài báo số 5	Hội nghị Quốc tế The International Conference on Machine Learning and Soft Computing (ICMLSC)	2024
6	Bài báo số 6	Hội nghị Quốc tế The EAI International Conference on Context-Aware Systems and Applications (ICCASA)	2024
7	Bài báo số 7	Hội nghị Quốc tế The RIVF International Conference on Computing and Communication Technologies (RIVF), Ho Chi Minh City, Vietnam	2025
8	Bài báo số 8	Hội nghị Quốc tế The International Conference on Intelligent Information Technology (ICIIT) (chờ xuất bản)	2026
9	Bài báo số 9	Tạp chí Quốc tế EAI Endorsed Transactions on Scalable Information Systems (đã chấp nhận xuất bản)	2026

Trong Bảng danh mục công trình khoa học đã công bố, Bài báo số 4 và Bài báo số 6 cùng thuộc nhóm mô hình rừng học quần thể nhưng có điểm khác biệt là Bài báo số 4 (trong Chương 3) đề xuất rừng mật độ quan điểm sử dụng DBSCAN trên

khoảng cách Euclidean để phân cụm trước khi phân loại, trong khi Bài báo số 6 (Chương 4) đề xuất rừng đa tạp quan điểm sử dụng DBSCAN trên khoảng cách đa tạp $dist_{\mathcal{M}}$ được học từ ma trận quan hệ W, đồng thời tổng hợp kết quả qua Stacking meta-learner thay vì trung bình mật độ. Hình minh họa trong hai bài báo có cấu trúc tổng quan tương tự nhưng nội dung kỹ thuật tại bước phân cụm và bước tổng hợp là khác nhau, được trình bày chi tiết tại Chương 3 và Chương 4.

Bảng đối chiếu công trình – chương – nội dung sử dụng

TT	Công trình	Nội dung chính	Sử dụng trong luận án
1	Bài báo 1 (IJACSA, 2020)	Biểu diễn Doc2vec trên từ quan điểm, phân loại bằng Cây quyết định, Random Forest, Gradient Boosting trên Amazon Food Reviews (18.532 đánh giá).	Chương 1 (§1.1): bước khảo sát ban đầu, là cơ sở để chọn biểu diễn bộ năm thay cho véc-tơ nhúng Doc2vec; §2.7.1.1 (tập 18.532 đánh giá) và đoạn tổng kết cuối §2.7.3: số liệu tham chiếu 0,752 / 0,821 / 0,845. Doc2vec không được dùng trong ba mô hình đề xuất.
2	Bài báo 2 (ICCASA, 2021)	Rừng phân loại quan điểm trên ma trận bộ năm; 8.020 bộ năm trích từ 5.411 đánh giá Amazon; hai kịch bản thực nghiệm.	Chương 2 (§2.3–§2.7): Bảng 2.5, 2.8, 2.9 (Accuracy 0,805 và 0,82); đóng góp thứ nhất.
3	Bài báo 3 (ICIIT, 2023)	Rừng phân loại quan điểm trên TripAdvisor (45.706 bộ năm); tầm quan trọng và tương tác đặc trưng.	Chương 2 (§2.6–§2.7): Bảng 2.3, 2.6, 2.10–2.12; đóng góp thứ nhất.
4	Bài báo 4 (EAI Endorsed Trans. CASA, 2023)	Chuyển đổi yelpHotelData, yelpResData sang bộ năm; DBSCAN + K rừng, kết hợp bằng biểu quyết đa số (trong luận án và Phụ lục J: trung bình xác suất, công thức (3.18)); 7 cụm (Hotel), 9 cụm (Restaurant).	Chương 3 (quy trình dữ liệu, cấu trúc cụm), Bảng 4.5; Phụ lục J (Bảng J.4); đóng góp thứ hai và thứ tư.
5	Bài báo 5 (ICMLSC, 2024)	Rừng mật độ quan điểm kết hợp SHAP; kết quả trên yelpHotelData, yelpResData.	Chương 3: Bảng 3.5–3.9, §3.6 (SHAP); số liệu rừng mật độ tại Bảng 4.4; Phụ lục J; đóng góp thứ hai và thứ tư.

TT	Công trình	Nội dung chính	Sử dụng trong luận án
6	Bài báo 6 (ICCASA, 2024)	Rừng đa tạp quan điểm: ma trận quan hệ W từ rừng không giám sát, DBSCAN trên khoảng cách đa tạp, Stacking; 3 cụm (Hotel), 4 cụm (Restaurant).	Chương 4: Bảng 4.2–4.5, §4.7; số liệu rừng đa tạp tại Bảng 4.4; Phụ lục J; đóng góp thứ ba và thứ tư.
7	Bài báo 7 (RIVF, 2025)	Mô hình trích xuất bộ năm trên Twitter US Airline (3.200 mẫu gán nhãn).	Chương 1 (§1.6): phiên bản đầu của mô hình trích xuất; số liệu trong luận án theo bản mở rộng tại Bài báo 8; đóng góp thứ tư.
8	Bài báo 8 (ICIIT, 2026)	Khung xác suất trích xuất bộ năm $P(e\|x)P(a\|e,x)P(s\|e,a,x)P(h\|s,G)P(t\|x)$; 4.500 mẫu gán nhãn.	Chương 1 (§1.6): Bảng 1.5–1.7 (F1 0,78; 8,6 ms/tweet; 210 MB); đóng góp thứ tư.
9	Bài báo 9 (EAI Endorsed Trans. SIS, 2026, đã chấp nhận)	Công cụ gán nhãn bộ năm OpQuint; 500 mẫu cho mỗi miền (Twitter Airline, Amazon, Yelp Hotel, Yelp Restaurant).	Chương 1 (quy trình gán nhãn và kiểm định chất lượng bộ năm); đóng góp thứ tư.

Ghi chú: Bài báo 4 và Bài báo 6 có hình minh họa tổng quan tương tự nhưng khác nhau ở bước phân cụm (khoảng cách Euclidean/khoảng cách đa tạp) và bước tổng hợp K rừng (biểu quyết đa số /Stacking). Mô tả quy tắc nhãn 3★ trong Bài báo 4 và Bài báo 6 chưa thống nhất với mã nguồn thực nghiệm; luận án xác định lại theo mã nguồn (Bảng 3.3).

TÀI LIỆU THAM KHẢO

[1] B. Liu, Sentiment Analysis: Mining Sentiments, Opinions, and Emotions, 2nd ed. Cambridge University Press, 2020.

[2] H. Wang, Y. Lu, and C. X. Zhai, "Latent aspect rating analysis without aspect keyword supervision," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, San Diego, CA, USA, 2011, pp. 618–626, doi: 10.1145/2020408.2020505.

[3] A. Mukherjee, V. Venkataraman, B. Liu, and N. Glance, "What Yelp fake review filter might be doing," in *Proc. 7th Int. AAAI Conf. Weblogs Social Media (ICWSM)*, Cambridge, MA, USA, 2013, pp. 409–418, doi: 10.1609/icwsm.v7i1.14389.

[4] M. M. Truşcă and F. Frasincar, "Survey on aspect detection for aspect-based sentiment analysis," *Artif. Intell. Rev.*, vol. 56, pp. 3797-3846, 2023, doi: 10.1007/s10462-022-10252-y

[5] K. Cortis and B. Davis, "Over a decade of social opinion mining: A systematic review," *Artif. Intell. Rev.*, vol. 54, pp. 4873–4965, 2021, doi: 10.1007/s10462-021-10030-2.

[6] W. Zhang, X. Li, Y. Deng, L. Bing, W. Lam, "A survey on aspect-based sentiment analysis: Tasks, methods, and challenges," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 11, pp. 11019-11038, 2023, doi: 10.1109/TKDE.2022.3230975.

[7] G. Brauwers and F. Frasincar, "A survey on aspect-based sentiment classification," *ACM Comput. Surv.*, vol. 55, no. 4, Art. no. 65, 37 pp., 2023, doi: 10.1145/3503044.

[8] A. Onan and S. Korukoğlu, "Ensemble methods for opinion mining," in *Proc. 23rd IEEE Signal Process. Commun. Appl. Conf. (SIU)*, Malatya, Türkiye, pp. 212–215, 2015, doi: 10.1109/SIU.2015.7129796.

[9] A. Alslaity and R. Orji, "Machine learning techniques for emotion detection and sentiment analysis: Current state, challenges, and future directions," *Behav. Inf. Technol.*, vol. 43, no. 1, pp. 139–164, 2024, doi: 10.1080/0144929X.2022.2156387.

[10] M. M. Rahman and M. N. Islam, "Exploring the performance of ensemble machine learning classifiers for sentiment analysis of COVID-19 tweets," in *Proc. Int. Conf. Sentimental Analysis and Deep Learning (ICSADL)*, vol.1408, pp. 383-396, 2022, doi: 10.1007/978-981-16-5157-1_30.

[11] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retrieval*, vol. 2, nos. 1–2, pp. 1–135, 2008, doi: 10.1561/1500000011.

[12] L. Zhang and B. Liu, "Aspect and entity extraction for opinion mining," in *Data Mining and Knowledge Discovery for Big Data, Studies in Big Data*, pp.1-40, 2014, doi: 10.1007/978-3-642-40837-3_1.

[13] B. Efron, "Bootstrap methods: Another look at the jackknife," *Ann. Statist.*, vol. 7, no. 1, pp. 1-26, 1979, doi: 10.1214/aos/1176344552.

[14] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, pp. 1631–1642, 2013. [Online]. Available: https://aclanthology.org/D13-1170.

[15] N. Sharma and V. Jain, "Evaluation and summarization of student feedback using sentiment analysis," in *Proc. Advances in Intelligent Systems and Computing*, vol 1141, pp. 385-396, 2020, doi: 10.1007/978-981-15-3383-9_35.

[16] A. Tharwat, "Classification assessment methods," *Appl. Comput. Inform.*, vol. 17, no. 1, pp. 168-192, 2021, doi: 10.1016/j.aci.2018.08.003.

[17] G. Qiu, X. He, F. Zhang, Y. Shi, J. Bu, and C. Chen, "DASA: Dissatisfaction-oriented advertising based on sentiment analysis," *Expert Syst. Appl.*, vol. 37, no. 9, pp. 6182-6191, 2010, doi: 10.1016/j.eswa.2010.02.109.

[18] J. Jiao and Y. Zhou, "Sentiment polarity analysis based multi-dictionary," *Phys. Procedia*, vol. 22, pp. 590–596, 2011, doi: 10.1016/j.phpro.2011.11.091.

[19] K. Xu, S. S. Liao, J. Li, and Y. Song, "Mining comparative opinions from customer reviews for competitive intelligence," *Decis. Support Syst.*, vol. 50, no. 4, pp. 43-54, 2011, doi: 10.1016/j.dss.2010.08.021.

[20] A. Alshutayri and E. Atwell, "Exploring Twitter as a source of an Arabic dialect corpus," *Int. J. Comput. Linguist.*, vol. 8, no. 2, pp. 37-44, 2017.

[21] C. S. G. Khoo and S. B. Johnkhan, "Lexicon-based sentiment analysis: Comparative evaluation of six sentiment lexicons," *Journal of Information Science*, vol. 44, no. 4, pp. 491-511, 2017, doi: 10.1177/0165551517703514.

[22] M. Wook, N. A. M. Razali et al., "Opinion mining technique for developing student feedback analysis system using lexicon-based approach (OMFeedback)," *Educ. Inf. Technol.*, vol. 25, pp. 2549-2560, 2020, doi: 10.1007/s10639-019-10073-7.

[23] O. Grljević, Z. Bošnjak, and A. Kovacevic, "Opinion mining in higher education: A corpus-based approach," *Enterprise Information Systems*, vol. 16, no. 5, 2020, doi: 10.1080/17517575.2020.1773542.

[24] P. R. J. Dhanith and K. S. S. Prabha, “A critical empirical evaluation of deep learning models for solving aspect-based sentiment analysis,” *Artif. Intell. Rev.*, vol. 56, pp. 13127-13186, 2023, doi: 10.1007/s10462-023-10460-0.

[25] J. Y.-L. Chan, K. T. Bea, S. M. H. Leow, S. W. Phoong, and W. K. Cheng, “State of the art: A review of sentiment analysis based on sequential transfer learning*,” Artif. Intell. Rev.*, vol. 56, pp. 749-780, 2023, doi: 10.1007/s10462-022-10183-8.

[26] Z. Li, H. Lu, C. Zhao, L. Feng, G. Gu, and W. Chen, “Weakly supervised discriminate enhancement network for visual sentiment analysis,” *Artif. Intell. Rev.*, vol. 56, pp. 1763-1785, 2023, doi: 10.1007/s10462-022-10212-6.

[27] S. He and Y. Wang, “Evaluating new energy vehicles by picture fuzzy sets based on sentiment analysis from online reviews,” *Artif. Intell. Rev.*, vol. 56, pp. 2171-2192, 2023, doi: 10.1007/s10462-022-10217-1.

[28] W. Liao, J. Zhou, Y. Wang, and X. Zhang, “Fine-grained attention-based phrase-aware network for aspect-level sentiment analysis,” *Artif. Intell. Rev.*, vol. 55, pp. 3727-3746, 2022, doi: 10.1007/s10462-021-10080-6.

[29] W. Zhang, Y. Deng, X. Li, Y. Yuan et al., “Aspect sentiment quad prediction as paraphrase generation,” in *Proc. 2021 Conf. Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9209-9219, 2021, doi: 10.18653/v1/2021.emnlp-main.726.

[30] S. Kumar, M. Gahalawat, P. P. Roy, D. P. Dogra, and B.-G. Kim, “Exploring impact of age and gender on sentiment analysis using machine learning,” *Electronics*, vol. 9, no. 2, p. 374, 2020, doi: 10.3390/electronics9020374.

[31] J. Samuel, G. M. Ali, M. M. Rahman, E. Esawi, and Y. Samuel, “COVID-19 public sentiment insights and machine learning for tweets classification,” *Information*, vol. 11, no. 6, p. 314, 2020, doi: 10.3390/info11060314.

[32] S. Zahoor and R. Rohilla, “Twitter sentiment analysis using machine learning algorithms: A case study,” in *Proc. Int. Conf. Adv. Comput., Commun. Mater. (ICACCM)*, pp. 194-199, 2020, doi: 10.1109/ICACCM50413.2020.9213011.

[33] P. Mukherjee, Y. Badr, S. Doppalapudi, S. M. Srinivasan, R. S. Sangwan, and R. Sharma, “Effect of negation in sentences on sentiment analysis and polarity detection,” *Procedia Comput. Sci.*, vol. 185, pp. 370-379, 2021, doi: 10.1016/j.procs.2021.05.038.

[34] B. Noori, “Classification of customer reviews using machine learning algorithms,” *Appl. Artif. Intell.*, vol. 35, no. 8, pp. 567-588, 2021, doi: 10.1080/08839514.2021.1922843.

[35] S. A. Arif and T. B. Hossain, “Opinion mining of customer reviews using supervised learning algorithms,” in *Proc. 5th Int. Conf. Electrical Information and Communication Technology (EICT)*, pp. 1-6, 2021, doi: 10.1109/EICT54103.2021.9733435.

[36] Z. Li, Y. Chen, Y. LeCun, and F. T. Sommer, “Neural Manifold Clustering and Embedding,” in *Proc. 10th Int. Conf. Learn. Represent. (ICLR)*, 2022, doi: 10.48550/arXiv.2201.10000.

[37] C. Messaoudi, Z. Guessoum, and L. ben Romdhane, “A deep learning model for opinion mining in Twitter combining text and emojis,” *Procedia Comput. Sci.*, vol. 207, pp. 2528-2537, 2022, doi: 10.1016/j.procs.2022.09.321.

[38] B. AlBadani, R. Shi, and J. Dong, “A novel machine learning approach for sentiment analysis on Twitter incorporating the universal language model fine-tuning and SVM,” *Appl. Syst. Innov.*, vol. 5, no. 1, p. 13, 2022, doi: 10.3390/asi5010013.

[39] A. Adak, B. Pradhan, N. Shukla, and A. Alamri, “Unboxing deep learning model of food delivery service reviews using explainable artificial intelligence (XAI) technique,” *Foods*, vol. 11, no. 14, 2022, doi: 10.3390/foods11142019.

[40] J. Ahmed and M. Ahmed, “Classification, detection and sentiment analysis using machine learning over next generation communication platforms,” *Microprocess. Microsyst.*, vol. 98, 2023, doi: 10.1016/j.micpro.2023.104795.

[41] P. Gaur, S. Vashistha, and P. Jha, “Twitter sentiment analysis using Naive Bayes-based machine learning technique,” in *Advances in Intelligent Systems and Computing*, vol. 1432, Springer, 2023, doi: 10.1007/978-981-19-5443-6_27.

[42] Y. Qi and Z. Shabrina, “Sentiment analysis using Twitter data: A comparative application of lexicon- and machine-learning-based approach,” *Soc. Netw. Anal. Min.*, vol. 13, 2023, doi: 10.1007/s13278-023-01030-x.

[43] M. Haroon, Z. Alam, R. Kousar, J. Ahmad, and F. Nasim, “Sentiment Analysis of Customer Reviews on E-commerce Platforms: A Machine Learning Approach,” *Bull. Bus. Econ.*, vol. 13, no. 3, pp. 230–238, 2024, doi: 10.61506/01.00480.

[44] A. Koufakou, “Deep learning for opinion mining and topic classification of course reviews,” *Educ. Inf. Technol.*, vol. 29, pp. 2973-2997, 2023, doi:

[45] Z. Zhai, B. Liu, H. Xu, and P. Jia, “Clustering product features for opinion mining,” in *Proc. 4th ACM Int. Conf. Web Search and Data Mining (WSDM),* pp. 347-354, 2011, doi: 10.1145/1935826.1935884.

[46] Z. Zhai, B. Liu, L. Zhang, H. Xu, and P. Jia, “Identifying evaluative sentences in online discussions,” in *Proc. 25th AAAI Conf. Artificial Intelligence (AAAI)*, pp. 933-938, 2011, doi: 10.1609/aaai.v25i1.7966.

[47] N. A. M. Razali, N. A. Malizan, N. A. Hasbullah et al., “Opinion mining for national security: Techniques, domain applications, challenges and research opportunities,” *J. Big Data*, vol. 8, p.150, 2021, doi: 10.1186/s40537-021-00536-5.

[48] N. A. M. Razali et al., “Political security threat prediction framework using hybrid lexicon-based approach and machine learning technique,” *IEEE Access*, vol. 11, pp. 17151-17164, 2023, doi: 10.1109/ACCESS.2023.3246162.

[49] K. Dave, S. Lawrence, and D. M. Pennock, “Mining the peanut gallery: Opinion extraction and semantic classification of product reviews,” in *Proc. 12th Int. Conf. World Wide Web (WWW)*, pp. 519-528, 2003, doi: 10.1145/775152.775226.

[50] R. Alnashwan, A. P. O'Riordan, H. Sorensen, and C. Hoare, “Improving sentiment analysis through ensemble learning of meta-level features,” in *Proc. 2nd Int. Workshop Knowl. Discovery Web (KDWEB)*, *CEUR Workshop Proc.*, vol. 1748, 2016. [Online]. Available: https://ceur-ws.org/Vol-1748/paper-05.pdf.

[51] M. M. Fouad, T. F. Gharib, and A. S. Mashat, “Efficient Twitter sentiment analysis system with feature selection and classifier ensemble,” in *Proc. Int. Conf. Advanced Machine Learning Technologies and Applications (AMLTA)*, vol. 723, pp. 516-527, 2018, doi: 10.1007/978-3-319-74690-6_51.

[52] S. Sharma, S. Srivastava, A. Kumar et al., “Multi-class sentiment analysis comparison using SVM and BAGGING technique - an ensemble method,” in Proc. Int. Conf. Smart Computing and Electronic Enterprise (ICSCEE), pp. 1-6, 2018, doi: 10.1109/ICSCEE.2018.8538397.

[53] H.-Q. Nguyen and Q.-U. Nguyen, “An ensemble of shallow and deep learning algorithms for Vietnamese sentiment analysis,” in *Proc. 5th NAFOSTED Conf. Information and Computer Science (NICS)*, pp. 165-170, 2018, doi: 10.1109/NICS.2018.8606880.

[54] A. Alrehili and K. Albalawi, “Sentiment analysis of customer reviews using ensemble method,” in *Proc. Int. Conf. Computer and Information Sciences (ICCIS)*, pp.1-6, 2019, doi: 10.1109/ICCISci.2019.8716454.

[55] D. Tiwari and N. Singh, “Ensemble approach for Twitter sentiment analysis,” *Int. J. Inf. Technol. Comput. Sci. (IJITCS)*, vol. 11, no. 8, pp. 20-26, 2019, doi: 10.5815/ijitcs.2019.08.03.

[56] M. Khalid, I. Ashraf, A. Mehmood et al., “GBSVM: Sentiment classification from unstructured reviews using ensemble classifier,” Appl. Sci., vol. 10, no. 8, p. 2788, 2020, doi: 10.3390/app10082788.

[57] A. Mohammadi and A. Shaverizade, “Ensemble deep learning for aspect-based sentiment analysis,” *Int. J. Nonlinear Anal. Appl.*, vol. 12, pp. 29-38, 2021, doi: 10.22075/IJNAA.2021.4769.

[58] R. V. A. Ogutu, R. Rimiru, and C. Otieno, “Target sentiment analysis ensemble for product review classification,” *J. Inf. Technol. Res. (JITR)*, vol. 15, no. 1, pp. 1-13, 2022, doi: 10.4018/JITR.299382.

[59] M. S. Basarslan and F. Kayaalp, “Sentiment analysis with ensemble and machine learning methods in multi-domain datasets,” *Turk. J. Eng.*, vol. 7, no. 2, pp. 141–148, 2023, doi: 10.30931/tje.1265303.

[60] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Belmont, CA, USA: Wadsworth International Group, 1984.

[61] J. R. Quinlan, “Induction of decision trees,” Mach. Learn., vol. 1, pp. 81-106, 1986, doi: 10.1007/BF00116251.

[62] L. Breiman, “Bagging predictors,” Mach. Learn., vol. 24, pp. 123-140, 1996, doi: 10.1007/BF00058655.

[63] G. Kunapuli, *Ensemble Methods for Machine Learning*. Manning Publications, 2023.

[64] D. H. Wolpert, “Stacked generalization,” *Neural Netw.*, 5(2), pp.241-259, 1992, doi: 10.1016/S0893-6080(05)80023-1.

[65] L. Breiman, “Random forests,” *Mach. Learn.*, vol. 45, pp.5-32, 2001, doi: 10.1023/A:1010933404324.

[66] A. Criminisi, J. Shotton, and E. Konukoglu, “Decision Forests: A Unified Framework for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning,” *Found. Trends Comput. Graph. Vis.*, vol. 7, nos. 2–3, pp. 81–227, 2012, doi: 10.1561/0600000035.

[67] J. J. McAuley and J. Leskovec, “From amateurs to connoisseurs: Modeling the evolution of user expertise through online reviews,” in *Proc. 22nd Int. Conf. World Wide Web (WWW)*, pp. 897-908, 2013, doi: 10.1145/2488388.2488466.

[68] H. Wen and H. Hang, “Random Forest Density Estimation,” in *Proc. 39th Int. Conf. Mach. Learn. (ICML)*, vol. 162, 2022, pp. 23701–23722. [Online]. Available: https://proceedings.mlr.press/v162/wen22c.htm.

[69] J. Dong and Q. Qian, “A density-based random forest for imbalanced data classification,” *Future Internet*, vol. 14, no. 90, 2022, doi: 10.3390/fi14030090.

[70] H. Hang, Y. Cai, and H. Yang, “Density-based clustering with best-scored random forest,” arXiv:1906.05719, 2019, doi: 10.48550/arXiv.1906.10094.

[71] R. Perry, T. M. Tomita, J. Patsolic et al., "Manifold forests: Closing the gap on neural networks," arXiv:1907.05274, 2019.

[72] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in *Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining (KDD)*, pp. 226-231, 1996, doi: 10.5555/3001460.3001507.

[73] X. Zhang, “Gaussian Distribution,” in *Encyclopedia of Machine Learning*, Springer, 2011, pp. 425–428, doi: 10.1007/978-0-387-30164-8_323.

[74] C. Molnar, Interpretable Machine Learning: A Guide for Making Black Box Models Explainable, 2nd ed., 2023.

[75] S. M. Lundberg, G. Erion, H. Chen et al., “From local explanations to global understanding with explainable AI for trees,” *Nat. Mach. Intell.*, vol. 2, pp. 56-67, 2020, doi: 10.1038/s42256-019-0138-9.

[76] R. Marcinkevics and J. E. Vogt, “Interpretable and explainable machine learning: A methods-centric overview with concrete examples,” *WIREs Data Min. Knowl. Discov.*, vol. 13, no. 3, 2023, doi: 10.1002/widm.1493.

[77] E. Levina and P. J. Bickel, “Maximum likelihood estimation of intrinsic dimension,” in Proc. Adv. Neural Inf. Process. Syst. (NIPS), vol. 17, pp. 777–784, 2005.

[78] J. Barnes, L. Oberlaender, A. Kutuzov, E. Troiano, J. Buchmann, R. Agerri, L. Ovrelid, and E. Velldal, “SemEval-2022 task 10: Structured sentiment analysis,” in *Proc. 16th Int. Workshop Semantic Evaluation (SemEval-2022)*, pp. 1280-1295, 2022, doi: 10.18653/v1/2022.semeval-1.180.

[79] J. Cohen, “A coefficient of agreement for nominal scales,” *Educ. Psychol. Meas.*, vol. 20, no. 1, pp. 37-46, 1960, doi: 10.1177/001316446002000104.

[80] CrowdFlower, “Twitter US Airline Sentiment Dataset,” *Kaggle*, 2015. [Online]. Available: https://www.kaggle.com/crowdflower/twitter-airline-sentiment.

[81] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. 2019 Conf. North American Chapter Assoc. Comput. Linguist.: Human Language Technologies (NAACL-HLT)*, vol. 1, pp. 4171-4186, 2019, doi: 10.48550/arXiv.1810.04805.

[82] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, "spaCy: Industrial-strength natural language processing in Python," *Zenodo*, 2020, doi: 10.5281/zenodo.1212303.

[83] Y. Liu, M. Ott, N. Goyal et al., "RoBERTa: A robustly optimized BERT pretraining approach," arXiv:1907.11692 [cs.CL], 2019, doi: 10.48550/arXiv.1907.11692.

[84] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189-1232, 2001, doi: 10.1214/aos/1013203451.

[85] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735-1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[86] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, pp. 273-297, 1995, doi: 10.1007/BF00994018.

[87] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, pp. 79-86, 2002, doi: 10.3115/1118693.1118704.

[88] P. Zhou, W. Shi, J. Tian, Z. Qi, B. Li, H. Hao, and B. Xu, "Attention-based bidirectional long short-term memory networks for relation classification," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguist. (ACL)*, vol. 2, pp. 207-212, 2016, doi: 10.18653/v1/P16-2034.

[89] A. B. Arrieta, N. Diaz-Rodriguez, J. Del Ser et al., "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Inf. Fusion*, vol. 58, pp. 82-115, 2020, doi: 10.1016/j.inffus.2019.12.012.

[90] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nat. Mach. Intell.*, vol. 1, pp. 206-215, 2019, doi: 10.1038/s42256-019-0048-x.

[91] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor, *An Introduction to Statistical Learning: With Applications in R*, 2nd ed. Cham, Switzerland: Springer, 2021, doi: 10.1007/978-1-0716-1418-1.

[92] I. D. Mienye and N. Jere, "A Survey of Decision Trees: Concepts, Algorithms, and Applications," in *IEEE Access*, vol. 12, pp. 86716-86727, 2024, doi: 10.1109/ACCESS.2024.3416838.

[93] J. Kazmaier and J. H. van Vuuren, "The power of ensemble learning in sentiment analysis," *Expert Syst. Appl.*, vol. 187, p.115819, 2022, doi:10.1016/j.eswa.2021.115819

[94] R. E. Schapire, "The strength of weak learnability," *Machine Learn 5*, pp. 197–227, 1990, doi: 10.1007/BF00116037.

[95] L. Franceschi, M. Donini, V. Perrone, A. Klein, C. Archambeau, M. Seeger, M. Pontil, and P. Frasconi, “Hyperparameter Optimization in Machine Learning,” *Found. Trends Mach. Learn.*, vol. 18, no. 6, pp. 1054–1201, 2025, doi: 10.1561/2200000088.

[96] C. Sammut and G. I. Webb, *Encyclopedia of Machine Learning and Data Mining*, 2nd ed. Boston, MA, USA: Springer, 2017, doi: 10.1007/978-1-4899-7687-1.

[97] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, pp. 3111–3119, 2013, doi: 10.48550/arXiv.1310.4546.

[98] J. Pennington, R. Socher, and C. D. Manning, “GloVe: Global vectors for word representation,” in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, pp. 1532–1543, 2014, doi: 10.3115/v1/D14-1162.

[99] A. Conneau et al., “Unsupervised cross-lingual representation learning at scale,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440-8451, 2020, doi: 10.18653/v1/2020.acl-main.747.

[100] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter,” arXiv:1910.01108, 2020, doi: 10.48550/arXiv.1910.01108.

[101] P. He, X. Liu, J. Gao, and W. Chen, “DeBERTa: Decoding-enhanced BERT with disentangled attention,” in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021, doi: 10.48550/arXiv.2006.03654.

[102] Z. Yang et al., “XLNet: Generalized autoregressive pretraining for language understanding,” in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS),* 2019, doi: 10.48550/arXiv.1906.08237.

[103] A. Vaswani et al., “Attention Is All You Need,” in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, pp. 5998–6008, 2017, doi: 10.48550/arXiv.1706.03762.

[104] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” in *Proc. 2018 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Language Technol.*, vol. 1, pp. 2227–2237, 2018, doi: 10.18653/v1/N18-1202.

[105] W. X. Zhao, J. Jiang, H. Yan, and X. Li, “Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid,” in *Proc. 2010 Conf. Empirical Methods Natural Language Process. (EMNLP)*, pp. 56–65, 2010. [Online]. Available: https://aclanthology.org/D10-1006/.

[106] P. D. Turney, “Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews,” in *Proc. 40th Annu. Meeting Assoc. Comput. Linguist. (ACL)*, 2002, pp. 417–424, doi: 10.3115/1073083.1073153.

[107] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, “Predicting elections with Twitter: What 140 characters reveal about political sentiment,” in *Proc. Int. AAAI Conf. Web Soc. Media*, vol. 4, no. 1, pp. 178–185, 2010, doi: 10.1609/icwsm.v4i1.14009.

[108] D. Antypas, A. Preece, and J. Camacho-Collados, “Negativity spreads faster: A large-scale multilingual Twitter analysis on the role of sentiment in political communication,” *Online Social Networks and Media*, vol. 33, 2023, doi: 10.1016/j.osnem.2023.100242.

[109] M. S. Md Suhaimin, M. H. A. Hijazi, E. G. Moung, P. N. E. Nohuddin, S. Chua, and F. Coenen, “Social media sentiment analysis and opinion mining in public security: Taxonomy, trend analysis, issues and future directions,” *J. King Saud Univ. Comput. Inf. Sci.*, vol. 35, no. 9, Art. no. 101776, 2023, doi: 10.1016/j.jksuci.2023.101776.

[110] H. Smith and W. Cipolli, “The Instagram/Facebook ban on graphic self-harm imagery: A sentiment analysis and topic modeling approach,” *Policy Internet*, vol. 14, no. 1, pp. 170–185, 2022, doi: 10.1002/poi3.272.

[111] G. Yang, Y. Xu, and L. Tu, “An intelligent box office predictor based on aspect-level sentiment analysis of movie review,” *Wireless Netw.*, vol. 29, no. 7, pp. 3039–3049, 2023, doi: 10.1007/s11276-023-03378-6.

[112] A. M. Sarhan, H. Ayman, M. Wagdi, B. Ali, A. Adel, and R. Osama, “Integrating machine learning and sentiment analysis in movie recommendation systems,” *J. Electr. Syst. Inf. Technol.*, vol. 11, Art. no. 53, 2024, doi: 10.1186/s43067-024-00177-7.

[113] S. Özöğür Akyüz, P. Karadayı Ataş, and A. Benkhaldoun, “Predicting stock market by sentiment analysis and deep learning,” *Oper. Res. Decis.*, vol. 34, no. 2, pp. 85–107, 2024. DOI: 10.37190/ord240206.

[114] W. Yang, Y. Lin, H. Xue, and J. Wang, “Research on stock market sentiment analysis and prediction method based on convolutional neural network,” in *Proc. Int. Conf. Machine Learning and Neural Networks (MLNN),* pp. 91–96, 2025, doi: 10.1145/3747227.3747241.

[115] Y. Kryvenchuk and V. Shkliarov, “Sentiment-driven stock market prediction with CNN-LSTM,” in *Proc. 19th IEEE Int. Conf. Computer Science and Information Technologies (CSIT)*, 2024. doi: 10.1109/CSIT65290.2024.10982642.

[116] W. Liu and J. Wang, “Financial sentiment analysis using FinBERT with application in predicting stock movement,” arXiv:2306.02136, 2023, doi: 10.48550/arXiv.2306.02136.

[117] G. Wang, J. Sun, J. Ma, K. Xu, and J. Gu, “Sentiment classification: The contribution of ensemble learning,” *Decision Support Systems*, vol. 57, pp. 77–93, 2014, doi: 10.1016/j.dss.2013.08.002.

[118] Z. Karimi, “Manifold-based sparse representation for opinion mining,” *Scientific Reports*, vol. 13, Art. no. 15904, 2023, doi: 10.1038/s41598-023-43088-9.

[119] M. K. Chandan and S. Mandal, “A comprehensive survey on sentiment analysis: Framework, techniques, and applications,” *Comput. Sci. Rev.*, vol. 58, Art. no. 100777, 2025, doi: 10.1016/j.cosrev.2025.100777.

[120] I. Villanueva-Miranda, Y. Xie, and G. Xiao, “Sentiment analysis in public health: A systematic review of the current state, challenges, and future directions,” *Front. Public Health*, vol. 13, p. 1609749, 2025, doi: 10.3389/fpubh.2025.1609749.

[121] M. Ijaz, N. Anwar, M. Safran, S. Alfarhood, T. Sadad, and Imran, “Domain adaptive learning for multi-realm sentiment classification on big data,” *PLOS ONE*, vol. 19, no. 4, p. e0297028, 2024, doi: 10.1371/journal.pone.0297028.

[122] A. Hidri, S. A. Alsaif, M. Alahmari, E. AlShehri, and M. S. Hidri, “Opinion mining and analysis using hybrid deep neural networks,” *Technologies*, vol. 13, no. 5, p. 175, 2025, doi: 10.3390/technologies13050175.

[123] S. Bassan, G. Amir, M. Zehavi, and G. Katz, “What makes an ensemble (un)interpretable?” in *Proc. 42nd Int. Conf. Mach. Learn. (ICML)*, *Proc. Mach. Learn. Res. (PMLR),* vol. 267, pp. 3142–3201, 2025.

[124] L. Grinsztajn, E. Oyallon, and G. Varoquaux, “Why do tree-based models still outperform deep learning on tabular data?” in *Proc. 36th Conf. Neural Inf. Process. Syst. (NeurIPS), Datasets and Benchmarks Track*, 2022, doi: 10.48550/arXiv.2207.08815.

[125] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, pp. 4768–4774, 2017, doi: 10.48550/arXiv.1705.07874.

[126] Y. Ning, M. E. H. Ong, B. Chakraborty, B. A. Goldstein, D. S. W. Ting, R. Vaughan, and N. Liu, “Shapley variable importance cloud for interpretable machine learning,” *Patterns*, vol. 3, no. 4, p. 100452, 2022, doi: 10.1016/j.patter.2022.100452.

[127] T. Brown et al., “Language models are few-shot learners,” in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, pp. 1877–1901, 2020, doi: 10.5555/3495724.3495883.

[128] A. Chowdhery et al., “PaLM: Scaling language modeling with pathways,” *J. Mach. Learn. Res.*, vol. 24, Art. no. 240, pp. 1–113, 2023. [Online]. Available: http://jmlr.org/papers/v24/22-1144.html.

[129] P. Rahman, F. Daneshfar, and H. Parvin, “Multi-objective manifold representation for opinion mining,” *Expert Syst.*, 42: e70092, 2025, doi: 10.1111/exsy.70092.

[130] M. Tkachenko, M. Malyuk, A. Holmanyuk, and N. Liubimov, “Label Studio: Data labeling software,” 2020. [Online]. Available: https://github.com/HumanSignal/label-studio.

[131] H. Fei, F. Li, C. Li, S. Wu, J. Li, and D. Ji, “Inheriting the wisdom of predecessors: A multiplex cascade framework for unified aspect-based sentiment analysis,” in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, pp. 4121–4128, 2022, doi: 10.24963/ijcai.2022/572.

[132] L. Xu, L. Bing, W. Lu, and F. Huang, “Aspect sentiment classification with aspect-specific opinion spans,” in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP 2020)*, pp. 3561–3567, 2020, doi: 10.18653/v1/2020.emnlp-main.288.

[133] K. Dhamdhere, A. Agarwal, and M. Sundararajan, “The Shapley Taylor interaction index,” in *Proc. 37th Int. Conf. Mach. Learn. (ICML)*, vol. 119, pp. 2387–2396, 2020, doi: 10.48550/arXiv.1902.05622.

[134] W. Aljedaani et al., “Cluster-based ensemble learning model for improving sentiment classification of Arabic documents,” *Natural Language Engineering*, vol. 30, no. 5, pp. 1091–1129, 2024, doi: 10.1017/S135132492300027X.

[135] H. Touvron et al., “LLaMA: Open and efficient foundation language models,” arXiv:2302.13971, 2023, doi: 10.48550/arXiv.2302.13971.

[136] Q. Gu, Z. Wang, H. Zhang, S. Sui, and R. Wang, “Aspect-Level Sentiment Analysis Based on Syntax-Aware and Graph Convolutional Networks,” *Applied Sciences*, vol. 14, no. 2, p. 729, 2024, doi: 10.3390/app14020729.

[137] B. Liang, H. Su, L. Gui, E. Cambria, and R. Xu, “Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks,” *Knowledge-Based Systems*, vol. 235, p. 107643, 2022, doi: 10.1016/j.knosys.2021.107643.

[138] W. Zhang, X. Li, Y. Deng, L. Bing, and W. Lam, “Towards generative aspect-based sentiment analysis,” in *Proc. 59th ACL-IJCNLP*, pp. 504-510, 2021, doi: 10.18653/v1/2021.acl-short.64.

[139] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, “A survey on bias and fairness in machine learning,” *ACM Comput. Surv.*, vol. 54, no. 6, pp. 1-35, 2021, doi: 10.1145/3457607.

[140] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Hoboken, NJ, USA: Wiley-Interscience, 2006.

[141] A. Joseph and R. Priya, “Aspect-based sentiment analysis using deep learning and ensemble modelling: A systematic review,” in *Proc. Int. Conf. Intell. Comput. Commun. (ICICC 2025), Lecture Notes in Networks and Systems*, vol. 1839, Springer, Cham, pp. 366–381, 2026, doi: 10.1007/978-3-032-18349-1_37.

[142] W. Zhang, Y. Deng, B. Liu, S. Pan, and L. Bing, “Sentiment analysis in the era of large language models: A reality check,” in *Findings Assoc. Comput. Linguistics: NAACL*, pp. 3881–3906, 2024, doi: 10.18653/v1/2024.findings-naacl.246.

[143] T. Shaik, X. Tao, C. Dann, H. Xie, Y. Li, and L. Galligan, “Sentiment analysis and opinion mining on educational data: A survey,” *Natural Lang. Process. J.*, vol. 2, Art. no. 100003, 2023, doi: 10.1016/j.nlp.2022.100003.

[144] T. Tong and Z. Li, “Predicting learning achievement using ensemble learning with result explanation,” *PLOS ONE*, vol. 20, no. 1, Art. no. e0312124, 2025, doi: 10.1371/journal.pone.0312124.

[145] A. M. Salih, Z. Raisi-Estabragh, I. B. Galazzo et al., “A perspective on explainable artificial intelligence methods: SHAP and LIME,” *Adv. Intell. Syst.*, vol. 6, no. 9, Art. no. 2400304, 2024, doi: 10.1002/aisy.202400304.

[146] M. Wankhade, A. C. S. Rao, and C. Kulkarni, “A survey on sentiment analysis methods, applications, and challenges,” *Artif. Intell. Rev.*, vol. 55, no. 7, pp. 5731–5780, 2022, doi: 10.1007/s10462-022-10144-1.

[147] M. Bordoloi and S. K. Biswas, “Sentiment analysis: A survey on design framework, applications and future scopes,” *Artif. Intell. Rev.,* vol. 56, pp. 12701–12735, 2023, doi: 10.1007/s10462-023-10442-2.

[148] A. Nazir, Y. Rao, L. Wu, and L. Sun, “Issues and challenges of aspect-based sentiment analysis: A comprehensive survey,” *IEEE Trans. Affect. Comput.*, vol. 13, no. 2, pp. 845–863, 2022, doi: 10.1109/TAFFC.2020.2970399.

[149] I. D. Mienye and Y. Sun, “A survey of ensemble learning: Concepts, algorithms, applications, and prospects,” *IEEE Access*, vol. 10, pp. 99129-99149, 2022, doi: 10.1109/ACCESS.2022.3207287.

[150] R. Dwivedi et al., “Explainable AI (XAI): Core ideas, techniques, and solutions,” *ACM Comput. Surv.,* vol. 55, no. 9, pp. 1-33, 2023, doi: 10.1145/3561048.

[151] W. Saeed and C. Omlin, “Explainable AI (XAI): A systematic meta-survey of current challenges and future opportunities,” *Knowledge-Based Systems*, vol. 263, Art. no. 110273, 2023, doi: 10.1016/j.knosys.2023.110273.

[152] Y. C. Hua, P. Denny, K. Taskova, and J. Wicker, “A systematic review of aspect-based sentiment analysis (ABSA): domains, methods, and trends,” *Artif. Intell. Rev.*, vol. 57, Art. no. 369, 2024, doi: 10.1007/s10462-024-10906-z.

[153] V. G. Costa and C. E. Pedreira, “Recent advances in decision trees: an updated survey,” *Artif. Intell. Rev.*, vol. 56, no. 5, pp. 4765-4800, 2023, doi: 10.1007/s10462-022-10275-5.

[154] V. Borisov, T. Leemann, K. Sessler, J. Haug, M. Pawelczyk, and G. Kasneci, “Deep neural networks and tabular data: A survey,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 6, pp. 7499-7519, 2024, doi: 10.1109/TNNLS.2022.3229161.

[155] R. Shwartz-Ziv and A. Armon, “Tabular data: Deep learning is not all you need,” Inf. *Fusion*, vol. 81, no. C, pp. 84-90, 2022.

[156] A. A. Khan, O. Chaudhari, and R. Chandra, “A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation,” *Expert Syst. Appl.*, vol. 244, Art. no. 122778, 2024, doi: 10.1016/j.eswa.2023.122778.

[157] R. J. G. B. Campello, P. Kröger, J. Sander, and A. Zimek, “Density-based clustering,” *WIREs Data Min. Knowl. Discov.*, vol. 10, no. 2, Art. no. e1343, 2020, doi: 10.1002/widm.1343.

[158] P. Mahajan, S. Uddin, F. Hajati, and M. A. Moni, “Ensemble learning for disease prediction: A review,” *Healthcare*, vol. 11, no. 12, Art. no. 1808, 2023, doi: 10.3390/healthcare11121808.

[159] M. Meilă and H. Zhang, “Manifold learning: What, how, and why,” Annu. Rev. Stat. Appl., vol. 11, pp. 393-417, 2024, doi: 10.1146/annurev-statistics-040522-115238.

[160] N. Mughal, G. Mujtaba, S. Shaikh, A. Kumar, and S. M. Daudpota, “Comparative analysis of deep natural networks and large language models for aspect-based sentiment analysis,” *IEEE Access*, vol. 12, pp. 60943-60959, 2024, doi: 10.1109/ACCESS.2024.3386969.

[161] R. Patil, S. Boit, V. Gudivada, and J. Nandigam, “A survey of text representation and embedding techniques in NLP,” *IEEE Access*, vol. 11, pp. 36120-36146, 2023, doi: 10.1109/ACCESS.2023.3266377.

[162] N. Malik and M. Bilal, “Natural language processing for analyzing online customer reviews: a survey, taxonomy, and open research challenges,” *PeerJ Comput. Sci.*, vol. 10, Art. no. e2203, 2024, doi: 10.7717/peerj-cs.2203.

[163] K. L. Tan, C. P. Lee, and K. M. Lim, “A survey of sentiment analysis: Approaches, datasets, and future research,” *Appl. Sci.*, vol. 13, no. 7, Art. no. 4550, 2023, doi: 10.3390/app13074550.

[164] D. Tiwari, B. Nagpal, B. S. Bhati, A. Mishra, and M. Kumar, “A systematic review of social network sentiment analysis with comparative study of ensemble-based techniques,” *Artif. Intell. Rev.*, vol. 56, pp. 13407-13461, 2023, doi: 10.1007/s10462-023-10472-w.

[165] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, “Deep learning-based text classification: A comprehensive review,” *ACM Comput. Surv.*, vol. 54, no. 3, Art. no. 62, 40 pp., 2021, doi: 10.1145/3439726.

[166] F. M. Shiri, T. Perumal, N. Mustapha, and R. Mohamed, “A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU,” arXiv preprint arXiv:2305.17473, 2023, doi: 10.32604/jai.2024.054314.

[167] P. K. Jain, R. Pamula, and G. Srivastava, “A systematic literature review on machine learning applications for consumer sentiment analysis using online reviews,” *Comput. Sci. Rev.,* vol. 41, 2021, Art. no. 100413, doi: 10.1016/j.cosrev.2021.100413.

[168] K. Qiu and L. Zhang, “How online reviews affect purchase intention: A meta-analysis across contextual and cultural factors,” *Data Inf. Manage.*, vol. 8, no. 2, Art. no. 100058, 2024, doi: 10.1016/j.dim.2023.100058.

[169] S. Poria, D. Hazarika, N. Majumder, and R. Mihalcea, “Beneath the tip of the iceberg: Current challenges and new directions in sentiment analysis research,” *IEEE Trans. Affective Comput.*, vol. 14, no. 1, pp. 108-132, 2023, doi: 10.1109/TAFFC.2020.3038167.

[170] M. Treviso et al., “Efficient methods for natural language processing: A survey,” *Trans. Assoc. Comput. Linguistics*, vol. 11, pp. 826-860, 2023, doi: 10.48550/arXiv.2209.00099.

[171] Z. Fan, Z. Yu, K. Yang, W. Chen, X. Liu, G. Li, X. Yang, and C. L. P. Chen, "Diverse models, united goal: A comprehensive survey of ensemble learning," *CAAI Trans. Intell. Technol.*, vol. 10, pp. 959-982, 2025, doi: 10.1049/cit2.70030.

[172] Q. Alvi, S. F. Ali, S. B. Ahmed, N. A. Khan, M. Javed, and H. Nobanee, "On the frontiers of Twitter data and sentiment analysis in election prediction: a review," *PeerJ Comput. Sci.*, vol. 9, Art. no. e1517, 2023, doi: 10.7717/peerj-cs.1517.

[173] N. Das, B. Sadhukhan, R. Chatterjee, and S. Chakrabarti, "Integrating sentiment analysis with graph neural networks for enhanced stock prediction: A comprehensive survey," *Decis. Anal. J.*, vol. 10, Art. no. 100417, 2024, doi: 10.1016/j.dajour.2024.100417.

[174] S. K. Maurya, D. Singh, and A. K. Maurya, "Deceptive opinion spam detection approaches: a literature survey," *Appl. Intell.*, vol. 53, pp. 2189-2234, 2023, doi: 10.1007/s10489-022-03427-1.

[175] R. Gupta, V. Jindal, and I. Kashyap, "Recent state-of-the-art of fake review detection: A comprehensive review," *Knowl. Eng. Rev.*, vol. 39, Art. no. e8, 2024, doi: 10.1017/S0269888924000067.

[176] B. Ilyas and A. Sharif, "A systematic review of social media-based sentiment analysis in disaster risk management," *Int. J. Disaster Risk Reduct.*, vol. 123, Art. no. 105487, 2025, doi: 10.1016/j.ijdrr.2025.105487.

[177] H. Zhang, Y. N. Cheah, O. M. Alyasiri, and J. An, "Exploring aspect-based sentiment quadruple extraction with implicit aspects, opinions, and ChatGPT: a comprehensive survey," *Artif. Intell. Rev.*, vol. 57, no. 2, Art. no. 17, 2024, doi: 10.1007/s10462-023-10633-x.

[178] L. Dube and T. Verster, "Interpretability of the random forest model under class imbalance," *Data Sci. Finance Econ.*, vol. 4, no. 3, pp. 446-468, 2024, doi: 10.3934/DSFE.2024019.

[179] M. Haddouchi and A. Berrado, "A survey and taxonomy of methods interpreting random forest models," arXiv preprint arXiv:2407.12759, 2024, doi: 10.48550/arXiv.2407.12759.

[180] A. A. Bushra, D. Kim, Y. Kan, and G. Yi, "AutoSCAN: automatic detection of DBSCAN parameters and efficient clustering of data in overlapping density regions," *PeerJ Comput. Sci.*, vol. 10, Art. no. e1921, 2024, doi: 10.7717/peerj-cs.1921.

[181] M. Muschalik, F. Fumagalli, B. Hammer, and E. Hüllermeier, "Beyond TreeSHAP: Efficient computation of any-order Shapley interactions for tree ensembles," in *Proc.*

AAAI Conf. Artif. Intell., vol. 38, no. 13, pp. 14388-14396, 2024, doi: 10.1609/aaai.v38i13.29352.

[182] Z. Bi and P. Lafaye De Micheaux, "A survey and comparative evaluation of intrinsic dimension estimators under the manifold hypothesis," arXiv preprint arXiv:2509.15517, 2025, doi: 10.48550/arXiv.2509.15517.

[183] Y. Hou, J. Li, X. Fu, Z. He, A. Yan, X. Chen, and J. McAuley, "Bridging language and items for retrieval and recommendation," arXiv preprint arXiv:2403.03952, 2024, doi: 10.48550/arXiv.2403.03952.

[184] B. V. Sánchez Vinces, E. Schubert, A. Zimek, and R. L. F. Cordeiro, "A comparative evaluation of clustering-based outlier detection," *Data Min. Knowl. Discov.*, vol. 39, no. 2, Art. no. 13, 2025, doi: 10.1007/s10618-024-01086-z.

[185] T. H. Lee, A. Ullah, and R. Wang, "Bootstrap aggregating and random forest," in *Macroeconomic Forecasting in the Era of Big Data*, pp. 389-429, 2020, doi: 10.1007/978-3-030-31150-6_13.

[186] W. Chen, K. Yang, Z. Yu, Y. Shi, and C. L. P. Chen, "A survey on imbalanced learning: latest research, applications and future directions," *Artif. Intell. Rev*., vol. 57, Art. no. 137, 2024, doi: 10.1007/s10462-024-10759-6.

[187] K. Alahmadi, S. Alharbi, J. Chen, and X. Wang, "Generalizing sentiment analysis: A review of progress, challenges, and emerging directions," *Social Network Analysis and Mining*, vol. 15, Art. no. 45, 2025, doi: 10.1007/s13278-025-01461-8.

[188] D. V. Thin, D. N. Hao, and N. L.-T. Nguyen, "Vietnamese sentiment analysis: An overview and comparative study of fine-tuning pretrained language models," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 22, no. 6, Art. no. 166, 2023, doi: 10.1145/3589131.

[189] D. V. Thin, D. N. Hao, and N. L.-T. Nguyen, "A systematic literature review on Vietnamese aspect-based sentiment analysis," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 22, no. 8, Art. no. 177, 2023, doi: 10.1145/3610226.

[190] R. Das and T. D. Singh, "Multimodal sentiment analysis: A survey of methods, trends, and challenges," *ACM Comput. Surv.*, vol. 55, no. 13s, Art. no. 270, pp. 1-38, 2023, doi: 10.1145/3586075.

[191] S. U. S. Chebolu, F. Dernoncourt, N. Lipka, and T. Solorio, "A review of datasets for aspect-based sentiment analysis," in Proc. 13th Int. Joint Conf. Natural Language Processing and 3rd Conf. Asia-Pacific Chapter Assoc. Computational Linguistics

(IJCNLP-AACL 2023), Long Papers, pp. 611-628, 2023, doi: 10.18653/v1/2023.ijcnlp-main.41.

[192] A. Ameur, S. Hamdi, and S. Ben Yahia, "Sentiment analysis for hotel reviews: A systematic literature review," *ACM Comput. Surv.*, vol. 56, no. 2, Art. no. 51, pp. 1-38, 2023, doi: 10.1145/3605152.

[193] W. Yi, S. Bu, H.-H. Lee, and C.-H. Chan, "Comparative Analysis of Manifold Learning-Based Dimension Reduction Methods: A Mathematical Perspective," *Mathematics*, vol. 12, no. 15, Art. no. 2388, 2024, doi: 10.3390/math12152388.

[194] Y. Wang, J. Qian, M. Hassan, X. Zhang, T. Zhang, C. Yang, X. Zhou, and F. Jia, "Density peak clustering algorithms: A review on the decade 2014-2023," *Expert Syst. Appl.*, vol. 238, Art. no. 121860, 2024, doi: 10.1016/j.eswa.2023.121860.

[195] Y. Mao, Q. Liu, and Y. Zhang, "Sentiment analysis methods, applications, and challenges: A systematic literature review," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 36, no. 4, Art. no. 102048, 2024. doi: 10.1016/j.jksuci.2024.102048.

[196] J.-W. Bi, X.-E. Zhu, and T.-Y. Han, "Text analysis in tourism and hospitality: A comprehensive review*," J. Travel Res.*, 2024, doi: 10.1177/00472875241247318.

[197] H. T. M. Nguyen, H. V. Nguyen, Q. T. Ngo, L. X. Vu, V. M. Tran, B. X. Ngo, and C. A. Le, "VLSP shared task: Sentiment analysis," *J. Comput. Sci. Cybern.*, vol. 34, no. 4, pp. 283–294, 2018. [Online]. Available: https://vlsp.org.vn/resources.

[198] J. Cohen, "Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit," *Psychological Bulletin*, vol. 70, no. 4, pp. 213–220, 1968, doi: 10.1037/h0026256.

[199] J. L. Fleiss, "Measuring nominal scale agreement among many raters," *Psychological Bulletin*, vol. 76, no. 5, pp. 378–382, 1971, doi: 10.1037/h0031619.

[200] R. S. Kamath, M. R. Kaushik, and M. Ramakrishna, "Opinion mining using ensemble model for restaurant feedback analysis," *Discover Computing*, vol. 28, Art. no. 125, 2025, doi: 10.1007/s10791-025-09583-5.

[201] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manage.*, vol. 45, no. 4, pp. 427–437, 2009, doi: 10.1016/j.ipm.2009.03.002.

[202] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, 2006, doi: 10.1016/j.patrec.2005.10.010.

[203] D. R. Cox, “The regression analysis of binary sequences,” *J. Roy. Stat. Soc.,* Ser. B (Methodol.), vol. 20, no. 2, pp. 215–242, 1958, doi:10.1111/j.2517-6161.1958.tb00292.x.

[204] T. M. Mitchell, *Machine Learning*. New York, NY, USA: McGraw-Hill, 1997, ch. 6, “Bayesian Learning,” pp. 154–200.

[205] N. Hu, J. Zhang, and P. A. Pavlou, “Overcoming the J-shaped distribution of product reviews,” *Commun. ACM*, vol. 52, no. 10, pp. 144–147, 2009, doi: 10.1145/1562764.1562800.

[206] E. Bigne, C. Ruiz, and C. Perez-Cabañero, “Are customer star ratings and sentiments aligned? A deep learning study of the customer service experience in tourism destinations,” *Service Business*, vol. 17, pp. 281–314, 2023, doi: 10.1007/s11628-023-00524-0.

[207] F. Wilcoxon, “Individual comparisons by ranking methods,” *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945, doi: 10.2307/3001968.

[208] T. G. Dietterich, “Approximate statistical tests for comparing supervised classification learning algorithms,” *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998, doi: 10.1162/089976698300017197.

[209] C. Ceruti, S. Bassis, A. Rozza, G. Lombardi, E. Casiraghi, and P. Campadelli, “DANCo: An intrinsic dimensionality estimator exploiting angle and norm concentration,” *Pattern Recognition*, vol. 47, no. 8, pp. 2569–2581, 2014, doi: 10.1016/j.patcog.2014.02.013.

PHỤ LỤC

Phụ lục A

TẬP DỮ LIỆU YELPRESDATA.DB

1. Tổng quan tập dữ liệu

Tập dữ liệu *YelpResData.db* [3] bao gồm các đánh giá nhà hàng được thu thập từ Yelp, phản ánh quan điểm người dùng đối với chất lượng dịch vụ, món ăn và trải nghiệm tổng thể. Tập dữ liệu gồm ba bảng chính: review, restaurant và reviewer. Cấu trúc các bảng dữ liệu và phân bố điểm đánh giá của tập dữ liệu được trình bày tại Bảng A.1 - Bảng A.4.

2. Cấu trúc bảng restaurant review

Bảng A.1: Cấu trúc bảng đánh giá nhà hàng

STT	Tên thuộc tính	Kiểu dữ liệu	Mô tả
1	date	varchar2(30)	Ngày đăng đánh giá
2	reviewID	varchar2(30)	Mã định danh đánh giá
3	reviewerID	varchar2(30)	Mã người đánh giá
4	reviewContent	varchar2(800)	Nội dung văn bản đánh giá
5	rating	INT	Điểm đánh giá (1–5)
6	usefulCount	INT	Số lượt đánh giá “hữu ích”
7	coolCount	INT	Số lượt đánh giá “thú vị”
8	funnyCount	INT	Số lượt đánh giá “hài hước”
9	flagged	varchar2(3)	Trạng thái gắn cờ
10	restaurantID	varchar2(30)	Mã nhà hàng

- Tổng số bản ghi: 788.471
- Khoảng thời gian: từ 12/10/2004 đến 22/10/2012

Bảng A.2: Phân bố điểm đánh giá nhà hàng

Điểm	Số lượng	Tỷ lệ (%)
1	45.642	5,79
2	71.672	9,09
3	155.427	19,71
4	290.538	36,85

5	225.192	28,56

3. Cấu trúc bảng restaurant

Bảng A.3: Cấu trúc bảng restaurant

Nhóm thông tin	Thuộc tính
Định danh	restaurantID, name, location
Đánh giá	reviewCount, rating, filReviewCount
Phân loại	categories
Thông tin dịch vụ	PriceRange, TakesReservations, Delivery, Takeout
Tiện ích	WiFi, Parking, OutdoorSeating, WheelchairAccessible
Môi trường	Ambience, NoiseLevel
Khác	Alcohol, GoodFor, HasTV, Caters
Liên hệ	address, phoneNumber, webSite
Chính sách	GoodforKids, GoodforGroups, Attire

- Số bản ghi bảng restaurant: 242.652 (số mã nhà hàng khác nhau xuất hiện trong bảng review: 250.078)

4. Cấu trúc bảng restaurant reviewer

Bảng A.4: Cấu trúc bảng *restaurant reviewer*

Thuộc tính	Mô tả
reviewerID	Mã người đánh giá
name	Tên người dùng
location	Địa điểm
yelpJoinDate	Thời điểm tham gia Yelp
friendCount	Số bạn bè
reviewCount	Số bài đánh giá
usefulCount, coolCount, funnyCount	Thống kê phản hồi
complimentCount	Số lời khen
tipCount	Số mẹo

- Số bản ghi bảng reviewer: 16.941 (số người đánh giá khác nhau xuất hiện trong bảng review: 35.430)

Phụ lục B

TẬP DỮ LIỆU YELPHOTELDATA.DB

1. Tổng quan tập dữ liệu

Tập dữ liệu *YelpHotelData.db* [3] phản ánh các đánh giá của người dùng đối với khách sạn, với cấu trúc tương tự tập nhà hàng, cho phép so sánh và kiểm chứng tính ổn định của phương pháp đề xuất trên các miền dữ liệu khác nhau. Cấu trúc các bảng dữ liệu và phân bố điểm đánh giá được trình bày tại Bảng B.1-Bảng B.4.

2. Cấu trúc bảng hotel review

Bảng B.1: Cấu trúc bảng đánh giá khách sạn

STT	Tên thuộc tính	Kiểu dữ liệu	Mô tả
1	date	varchar2(30)	Ngày đánh giá
2	reviewID	varchar2(30)	Mã đánh giá
3	reviewerID	varchar2(30)	Mã người đánh giá
4	reviewContent	varchar2(800)	Nội dung đánh giá
5	rating	INT	Điểm đánh giá
6	usefulCount	INT	Số lượt hữu ích
7	coolCount	INT	Số lượt thú vị
8	funnyCount	INT	Số lượt hài hước
9	flagged	varchar2(3)	Trạng thái
10	hotelID	varchar2(30)	Mã khách sạn

- Tổng số bản ghi: 688.329
- Khoảng thời gian: 12/10/2004 – 21/10/2012

Bảng B.2: Phân bố điểm đánh giá khách sạn

Điểm	Số lượng	Tỷ lệ (%)
1	32.390	4,71
2	62.560	9,09
3	154.637	22,47
4	267.027	38,79
5	171.715	24,95

3. Cấu trúc bảng hotel

Bảng B.3: Cấu trúc bảng hotel

Nhóm	Thuộc tính
Định danh	hotelID, name, location
Đánh giá	reviewCount, rating, filReviewCount
Phân loại	categories
Dịch vụ	PriceRange, WiFi
Thanh toán	AcceptsCreditCards
Liên hệ	address, phoneNumber, webSite

- Số bản ghi bảng hotel: 283.086 (số mã khách sạn khác nhau xuất hiện trong bảng review: 283.291)

4. Cấu trúc bảng reviewer

Bảng B.4: Cấu trúc bảng *hotel reviewer*

Thuộc tính	Mô tả
reviewerID	Mã định danh duy nhất của người đánh giá
name	Tên người đánh giá
location	Địa điểm sinh sống của người đánh giá
yelpJoinDate	Thời điểm người dùng tham gia Yelp
friendCount	Số lượng bạn bè trên Yelp
reviewCount	Tổng số bài đánh giá đã đăng
firstCount	Số lượng đánh giá đầu tiên (first reviews)
usefulCount	Số lượt đánh giá được gắn nhãn “hữu ích”
coolCount	Số lượt đánh giá được gắn nhãn “thú vị”
funnyCount	Số lượt đánh giá được gắn nhãn “hài hước”
complimentCount	Tổng số lời khen nhận được
tipCount	Số lượng mẹo (tips) đã đăng
fanCount	Số lượng người theo dõi (fans)

- Số bản ghi bảng reviewer: 5.123 (số người đánh giá khác nhau xuất hiện trong bảng review: 5.132).

Phụ lục C

CHUYỂN ĐỔI DỮ LIỆU ĐÁNH GIÁ SANG CẤU TRÚC BỘ NĂM QUAN ĐIỂM

1. Chuyển đổi dữ liệu YelpResData.db

1.1 Cấu trúc dữ liệu đầu vào

Sau khi kết hợp các bảng review, restaurant và reviewer, mỗi bản ghi đánh giá được biểu diễn dưới dạng một dòng, trong đó mỗi dòng tương ứng với một đánh giá và mỗi cột biểu diễn một thuộc tính hoặc khía cạnh của nhà hàng tại Bảng C.1.

Xử lý giá trị thiếu (YelpResData): Trong quá trình kết hợp các bảng review, restaurant và reviewer, một số bản ghi có thể thiếu thông tin ở các cột khía cạnh. Chiến lược xử lý: (1) Bản ghi thiếu restaurantName hoặc reviewerName bị loại bỏ; (2) Các khía cạnh có giá trị null không được chuyển thành bộ năm; (3) Cột date thiếu được điền bằng ngày trung vị trong cùng nhà hàng. Sau xử lý, tổng số bộ năm hợp lệ là 14.791.500 mẫu.

Bảng C.1: Cấu trúc dữ liệu đầu vào – Restaurant

Thuộc tính	Mô tả	Vai trò
review_id	Mã định danh đánh giá	Định danh
reviewContent	Nội dung đánh giá	Nguồn trích xuất
restaurantName	Tên nhà hàng	Thực thể
reviewerName	Người đánh giá	Người giữ quan điểm
date	Ngày đánh giá	Thời gian
rating	Điểm đánh giá (1–5)	Tình cảm
GoodforKids	Phù hợp trẻ em	Khía cạnh
AcceptsCreditCards	Chấp nhận thẻ	Khía cạnh
Parking	Bãi đỗ xe	Khía cạnh
Attire	Trang phục	Khía cạnh
PriceRange	Mức giá	Khía cạnh
…	Các thuộc tính khác	Khía cạnh

1.2 Phương pháp chuyển đổi sang bộ năm quan điểm nhà hàng

Sử dụng hàm *pivot_longer()* trong gói *tidyr* của R để thực hiện phép chuẩn hóa dữ liệu từ dạng rộng sang dạng dài.

Đoạn mã chuyển đổi tại Hình C.1 có ý nghĩa như sau:

- Giữ nguyên các thuộc tính của quan điểm:
  - restaurantName → Thực thể (entity)
  - reviewerName → Người giữ quan điểm (holder)
  - date → Thời gian (time)
  - rating → dùng làm tình cảm tổng thể
- Biến đổi các cột khía cạnh thành các dòng quan điểm riêng biệt:
  - Tên cột → Khía cạnh (aspect)
  - Giá trị gán → Tình cảm (sentiment)

Sau chuyển đổi, mỗi bản ghi tương ứng với một quan điểm – khía cạnh cụ thể.

```
library(tidyr)

restaurant_aspect <- pivot_longer(
  restaurant,
  cols = -c(review_id, reviewContent, restaurantName,
            reviewerName, date, rating),
  names_to = "aspect",
  values_to = "sentiment"
)
```

Hình C.1: Đoạn mã chuyển đổi trên bộ dữ liệu yelpResData

1.3. Cấu trúc dữ liệu đầu ra

Dữ liệu sau chuyển đổi có dạng bộ năm quan điểm ở Bảng C.2.

Bảng C.2: Cấu trúc dữ liệu đầu ra – Restaurant

entity	aspect	holder	sentiment	time
Alinea	GoodforKids	Scott E.	5	09/22/2012
Alinea	AcceptsCreditCards	Scott E.	5	09/22/2012
Alinea	Parking	Scott E.	5	09/22/2012
Alinea	Attire	Scott E.	5	09/22/2012
…	…	…	…	…

2. Chuyển đổi dữ liệu YelpHotelData.db

2.1 Cấu trúc dữ liệu đầu vào

Xử lý giá trị thiếu (YelpHotelData): Áp dụng chiến lược tương tự YelpResData. Các bản ghi thiếu hotelName hoặc reviewerName bị loại bỏ; các khía cạnh null (WiFi, PriceRange, AcceptsCreditCards) bị bỏ qua khi pivot; cột date thiếu được điền bằng ngày trung vị trong cùng khách sạn. Sau xử lý, tổng số bộ năm quan điểm là 882.474 mẫu.

Bảng C.3: Cấu trúc dữ liệu đầu vào – Hotel

Thuộc tính	Mô tả	Vai trò
review_id	Mã đánh giá	Định danh
reviewContent	Nội dung đánh giá	Nguồn
hotelName	Tên khách sạn	Thực thể
reviewerName	Người đánh giá	Người giữ quan điểm
date	Ngày đánh giá	Thời gian
rating	Điểm đánh giá	Tình cảm
WiFi	Dịch vụ WiFi	Khía cạnh
PriceRange	Mức giá	Khía cạnh
Cleanliness	Vệ sinh	Khía cạnh
RoomService	Dịch vụ phòng	Khía cạnh
Location	Vị trí	Khía cạnh
…	Các khía cạnh khác	Khía cạnh

2.2 Phương pháp chuyển đổi sang bộ năm quan điểm khách sạn

Phương pháp chuyển đổi tương tự tập nhà hàng, sử dụng pivot_longer() để ánh xạ mỗi khía cạnh thành một quan điểm có đoạn mã ở Hình C.2.

```
hotel_aspect <- pivot_longer(
  hotel,
  cols = -c(review_id, reviewContent, hotelName,
            reviewerName, date, rating),
  names_to = "aspect",
  values_to = "sentiment"
)
```

Hình C.2: Đoạn mã chuyển đổi trên bộ dữ liệu yelpHotelData

2.3. Cấu trúc dữ liệu đầu ra quan điểm khách sạn

Dữ liệu sau chuyển đổi có dạng bộ năm quan điểm khách sạn ở Bảng C.4

Bảng C.4: Cấu trúc dữ liệu đầu ra – Hotel

entity	**aspect**	**holder**	**sentiment**	**time**
Tokyo Hotel	AcceptsCreditCards	Max C.	5	06/08/2011
Tokyo Hotel	PriceRange	Max C.	5	06/08/2011
Tokyo Hotel	WiFi	Max C.	5	06/08/2011
…	…	…	…	…

Phụ lục D

CHI TIẾT CÔNG CỤ VÀ THƯ VIỆN SỬ DỤNG

Bảng D.1: Danh sách chi tiết các thư viện R và công cụ sử dụng trong thực nghiệm

STT	Danh mục	Thư viện	Phiên bản	Mục đích sử dụng
MÔI TRƯỜNG PHÁT TRIỂN				
1	Môi trường	R	4.3.2	Ngôn ngữ lập trình chính cho toàn bộ thực nghiệm
2	Môi trường	RStudio	2023.12.1	Môi trường phát triển tích hợp để phát triển và gỡ lỗi mã nguồn
XỬ LÝ DỮ LIỆU				
3	Xử lý dữ liệu	dplyr	1.1.4	Thao tác và biến đổi dữ liệu dạng ngăn nắp
4	Xử lý dữ liệu	tidyr	1.3.0	Chuyển đổi cấu trúc dữ liệu (dạng rộng ↔ dạng dài)
5	Xử lý dữ liệu	data.table	1.14.10	Xử lý hiệu năng cao với tập dữ liệu lớn
6	Xử lý dữ liệu	readr	2.1.4	Đọc dữ liệu định dạng CSV và tập tin văn bản
XỬ LÝ VĂN BẢN				
7	Khai thác văn bản	tm	0.7-11	Khung làm việc xử lý văn bản: loại bỏ từ dừng, dấu câu, trích rút gốc từ
8	Khai thác văn bản	text2vec	0.6.3	Tạo ma trận TF-IDF và nhúng từ
9	Khai thác văn bản	tidytext	0.4.1	Phân tích văn bản theo phương pháp ngăn nắp
10	Khai thác văn bản	tokenizers	0.3.0	Tách từ cho văn bản tiếng Anh
11	Khai thác văn bản	SnowballC	0.7.1	Các thuật toán trích rút gốc từ (Porter, Snowball)
12	Khai thác văn bản	stringr	1.5.1	Xử lý và thao tác chuỗi ký tự
13	Khai thác văn bản	word2vec	0.3.4	Xây dựng vector nhúng từ Word2Vec
14	Khai thác văn bản	glove	0.2.0	Xây dựng vector nhúng từ GloVe
RỪNG PHÂN LOẠI (MÔ HÌNH ĐỀ XUẤT)				
15	Rừng ngẫu nhiên	randomForest	4.7-1.1	Cài đặt chính thức thuật toán [65]

16	Rừng ngẫu nhiên	ranger	0.16.0	Cài đặt Rừng ngẫu nhiên nhanh với xử lý song song
17	Rừng ngẫu nhiên	randomForest Explainer	0.10.1	Giải thích Rừng ngẫu nhiên: tầm quan trọng biến, tương tác
RỪNG MẬT ĐỘ (MÔ HÌNH ĐỀ XUẤT)				
18	Phân cụm	dbscan	1.1-12	Thuật toán DBSCAN cho phân cụm dựa trên mật độ
19	Phân cụm	fpc	2.2-11	Các phương pháp phân cụm linh hoạt và kiểm định cụm
20	Phân cụm	cluster	2.1.4	Phân tích cụm và phân cụm thứ bậc
21	Phân cụm	factoextra	1.0.7	Trực quan hóa kết quả phân cụm
22	Học quần thể	randomForest	4.7-1.1	Xây dựng Rừng ngẫu nhiên trong từng cụm
23	Học quần thể	rpart	4.1.23	Thuật toán CART cho cây quyết định
RỪNG ĐA TẠP (MÔ HÌNH ĐỀ XUẤT)				
24	Rừng đa tạp	tree	1.0-43	Xây dựng cây quyết định không giám sát để học cấu trúc đa tạp
25	Rừng đa tạp	randomForest	4.7-1.1	Huấn luyện rừng ngẫu nhiên cho từng cụm trên không gian đa tạp
26	Rừng đa tạp	rpart	4.1.23	Phân hoạch đệ quy với entropy dựa trên ma trận hiệp phương sai
27	Rừng đa tạp	dbscan	1.1-12	Phân cụm DBSCAN trên ma trận quan hệ đa tạp
28	Rừng đa tạp	caret	6.0-94	Huấn luyện và đánh giá mô hình với kiểm định chéo cho các mô hình chuyên biệt theo cụm
29	Rừng đa tạp	tidymodels	1.1.1	Khung mô hình hóa gọn gàng cho quy trình làm việc và công thức
TĂNG CƯỜNG GRADIENT				
30	Tăng cường Gradient	gbm	2.1.9	Máy tăng cường Gradient (Friedman 2001)
31	Tăng cường Gradient	xgboost	1.7.7.1	Tăng cường Gradient cực đại với hỗ trợ GPU
32	Tăng cường Gradient	lightgbm	3.3.5	LightGBM (Microsoft) cho tập dữ liệu lớn

MÁY VECTOR HỖ TRỢ				
33	SVM	e1071	1.7-14	Gói bao bọc LIBSVM cho SVM với hàm nhân RBF
34	SVM	kernlab	0.9-32	Các phương pháp học máy dựa trên hàm nhân
HỌC SÂU (LSTM, BiLSTM-Attention)				
35	Học sâu	keras	2.13.0	Giao diện R của API Keras cho học sâu
36	Học sâu	tensorflow	2.13.0	Nền tảng TensorFlow cho keras
37	Học sâu	reticulate	1.35.0	Giao diện R-Python cho các khung làm việc học sâu
TRANSFORMER (BERT, RoBERTa)				
38	Transformer	reticulate	1.35.0	Cầu nối R-Python để sử dụng Hugging Face transformers
39	Transformer	transformers (Python)	4.35.0	Hugging Face Transformers: các mô hình tiền huấn luyện BERT, RoBERTa
40	Transformer	torch (Python)	2.1.0	Nền tảng PyTorch cho các mô hình Transformer
HUẤN LUYỆN VÀ ĐÁNH GIÁ				
41	Huấn luyện	caret	6.0-94	Giao diện thống nhất cho huấn luyện và điều chỉnh siêu tham số
42	Huấn luyện	mlr3	0.17.0	Khung làm việc học máy hiện đại trong R
43	Huấn luyện	tidymodels	1.1.1	Hệ sinh thái mô hình hóa ngăn nắp
44	Đánh giá	yardstick	1.3.0	Các độ đo hiệu năng: độ chính xác, độ chuẩn xác, độ thu hồi, F1
45	Đánh giá	MLmetrics	1.1.3	Các độ đo học máy bổ sung
46	Lấy mẫu lại	rsample	1.2.0	Lấy mẫu phân tầng và kiểm định chéo
47	Lấy mẫu lại	boot	1.3-28	Các phương pháp bootstrap để ước lượng độ không đảm bảo
GIẢI THÍCH MÔ HÌNH				
48	SHAP	treeshap	0.3.1	Thuật toán TreeSHAP cho các mô hình dựa trên cây

49	SHAP	fastshap	0.1.0	Giá trị SHAP xấp xỉ nhanh cho mọi mô hình
50	SHAP	shapviz	0.9.0	Trực quan hóa SHAP: biểu đồ tóm tắt, biểu đồ phụ thuộc
51	Độc lập mô hình	iml	0.11.2	Khung làm việc học máy khả diễn giải
52	Độc lập mô hình	DALEX	2.4.3	Giải thích học máy mô tả
53	Độc lập mô hình	DALEXtra	2.3.0	Các mở rộng cho DALEX
54	Độc lập mô hình	lime	0.5.3	Giải thích cục bộ khả diễn giải độc lập mô hình
55	Độc lập mô hình	breakDown	0.2.1	Giải thích phân bổ biến
TRỰC QUAN HÓA				
56	Trực quan hóa	ggplot2	3.4.4	Ngữ pháp đồ họa cho các biểu đồ chất lượng xuất bản
57	Trực quan hóa	patchwork	1.2.0	Kết hợp nhiều biểu đồ ggplot2 thành bố cục phức tạp
58	Trực quan hóa	cowplot	1.1.2	Tạo lưới biểu đồ và chủ đề
59	Trực quan hóa	ggiraph	0.8.7	Đồ họa ggplot2 tương tác
60	Trực quan hóa mô hình	vip	0.4.1	Biểu đồ tầm quan trọng biến
61	Trực quan hóa mô hình	pdp	0.8.1	Biểu đồ phụ thuộc từng phần
62	Trực quan hóa mô hình	ALEPlot	1.1	Biểu đồ hiệu ứng cục bộ tích lũy

Ghi chú: toàn bộ mô hình rừng đề xuất, mô hình so sánh và thực nghiệm bổ sung được triển khai bằng R; các mô hình học sâu LSTM, BiLSTM-Attention trong thực nghiệm so sánh thống nhất (Phụ lục J) được xây dựng bằng gói torch cho R, còn BERT, RoBERTa, PhoBERT được tinh chỉnh bằng thư viện transformers (Python, nền PyTorch) gọi từ kịch bản R. Các thực nghiệm chính tại Chương 2–4 dùng R 4.3.2; các thực nghiệm bổ sung tại Phụ lục F–J dùng R 4.3.3 và 4.5.2 với cùng các gói. Mô hình trích xuất bộ năm tại mục 1.6 được triển khai bằng Python (mục 1.6.4.2).

Phụ lục E

MÔ TẢ CÁC TẬP DỮ LIỆU SỬ DỤNG

Phụ lục này mô tả chi tiết các tập dữ liệu được sử dụng trong các thực nghiệm của luận án, bao gồm: nguồn gốc, kích thước, định dạng, phân bố nhãn và lý do lựa chọn. Sáu tập dữ liệu được sử dụng cho đa dạng các kịch bản thực nghiệm: Thông tin chi tiết của các tập dữ liệu được tổng hợp tại Bảng E.1 - Bảng E.12.

E.1. yelpHotelData và yelpResData

Hai tập dữ liệu yelpHotelData và yelpResData là dữ liệu đánh giá của người dùng trên nền tảng Yelp.com về khách sạn (Hotel) và nhà hàng (Restaurant) do Mukherjee và cộng sự thu thập và công bố [3]. Các tập dữ liệu này được sử dụng chủ yếu cho mô hình rừng mật độ (Chương 3) và rừng đa tạp (Chương 4) vì có phân bố nhãn lệch về lớp positive ($\rho \approx 1{,}7–1{,}9$) và cấu trúc phân nhóm tự nhiên, đặc trưng phù hợp với thiết kế tích hợp DBSCAN của các mô hình.

Bảng E.1: Kết quả thống kê tập dữ liệu yelpHotelData và yelpResData

Thuộc tính	yelpHotelData	yelpResData
Lĩnh vực	Khách sạn (Hotel)	Nhà hàng (Restaurant)
Tổng số đánh giá	688.329	788.471
Số khách sạn/nhà hàng	283.086 (bảng hotel)	242.652 (bảng restaurant)
Số người dùng (reviewers)	5.132 (trong bảng review)	35.430 (trong bảng review)
Tỷ lệ mất cân bằng ρ(O)	≈ 1,7 (560.769/321.705 bộ năm)	≈ 1,9 (9.650.537/5.140.963 bộ năm)
Định dạng nhãn	Đa lớp (1-5 sao) → 2 lớp	Đa lớp (1-5 sao) → 2 lớp
Quy ước nhãn	1-3 sao = Negative; 4-5 sao = Positive	1-3 sao = Negative; 4-5 sao = Positive
Loại bỏ	Không (3 sao gộp vào Negative)	Không (3 sao gộp vào Negative)

Mỗi đánh giá được chuyển đổi thành bộ năm quan điểm x = (*e, a, s, h, t*) sau khi qua quy trình tiền xử lý (chi tiết tại Phụ lục C):

(1) *e (entity)*: tên của khách sạn/nhà hàng được đánh giá;

(2) *a (aspect):* khía cạnh được đề cập (GoodforKids, Parking, PriceRange, WiFi, ...);

(3) *s (sentiment):* cực tính cảm xúc (positive/negative);

(4) *h (holder):* người viết đánh giá (reviewerName);

(5) *t (time):* thời điểm đăng đánh giá.

E.2. Amazon Reviews

Tập dữ liệu Amazon Reviews được sử dụng từ phiên bản công bố của nghiên cứu [67], bao gồm hàng triệu đánh giá sản phẩm trên Amazon.com. Trong luận án, tập con Amazon Fine Food Reviews (568.454 đánh giá) được sử dụng để kiểm chứng tính tổng quát của mô hình rừng phân loại (Chương 2) trên dữ liệu thương mại điện tử với sự đa dạng về sản phẩm.

Bảng E.2: Kết quả thống kê tập dữ liệu Amazon Reviews

Thuộc tính	**Giá trị**
Nguồn	Stanford SNAP - Amazon Reviews [67]
Domain sử dụng	Fine Food (Amazon Food Reviews)
Tổng số đánh giá (đầy đủ)	568.454 (toàn bộ); 18.532 (Bài báo 1); 5.411 đánh giá → 8.020 bộ năm (F_{CF})
Số đặc trưng đánh giá	Rating (1-5 sao), Review text, Helpful votes, Verified purchase
Tỷ lệ mất cân bằng ρ(O)	≈ 5,4 (Mất cân bằng trung bình)
Quy ước nhãn	1-2 sao = Negative; 4-5 sao = Positive; 3 sao (loại bỏ)

E.3. TripAdvisor Reviews

Tập dữ liệu TripAdvisor được trích xuất từ các đánh giá về khách sạn trên nền tảng TripAdvisor. Tập dữ liệu này có cấu trúc đặc biệt với nhiều khía cạnh đánh giá riêng biệt (Value, Rooms, Location, Cleanliness, Check in/front desk, Service, Business service), cho phép kiểm chứng khả năng phân loại đa khía cạnh (multi-aspect ABSA) của mô hình rừng phân loại (Chương 2).

Bảng E.3: Kết quả thống kê tập dữ liệu TripAdvisor Reviews

Thuộc tính	Giá trị
Nguồn	TripAdvisor Hotels Dataset (Wang và cộng sự, KDD 2011 [2])
Tổng số đánh giá	108.891 (tập con sử dụng trong luận án, mục 2.7.1.2)
Số khách sạn	1.850
Số khía cạnh đánh giá riêng	7 (Value, Rooms, Location, Cleanliness, Check in/front desk, Service, Business service)
Định dạng đánh giá khía cạnh	1-5 sao trên mỗi khía cạnh
Số ngôn ngữ	Tiếng Anh

E.4. Tập dữ liệu VLSP-2018 (tiếng Việt)

Mục này mô tả chi tiết tập dữ liệu VLSP-2018 ABSA tiếng Việt, tập dữ liệu ABSA chuẩn cho tiếng Việt hiện được sử dụng để kiểm chứng khả năng tổng quát hóa của ba mô hình rừng đề xuất trên ngôn ngữ Việt. Mục này trình bày toàn bộ quy trình: nguồn gốc, thống kê tập dữ liệu, quy trình chú thích, cơ sở chú thích và khảo sát các tập dữ liệu ABSA tiếng Việt hiện có.

Tập dữ liệu VLSP-2018 ABSA được công bố tại [197] để phân tích tình cảm dựa trên khía cạnh (ABSA) tiếng Việt, nối tiếp tập dữ liệu VLSP-2016 chỉ thực hiện phân loại cực tính tổng thể. Tập dữ liệu cung cấp hai tập con đánh giá nhà hàng và

khách sạn tại Việt Nam, mỗi đánh giá được chú thích bộ ba [189] *{Entity#Attribute, Sentiment}* ở cấp tài liệu (document-level ABSA).

Bảng E.4: Thống kê tổng quan tập dữ liệu VLSP-2018 ABSA

Thuộc tính	**VLSP-2018 Restaurant**	**VLSP-2018 Hotel**
Lĩnh vực	Nhà hàng tại Việt Nam	Khách sạn tại Việt Nam
Số văn bản Train	2.961	3.000
Số văn bản Dev	1.290	2.000
Số văn bản Test	500	600
Tổng số văn bản	4.751	5.600
Số aspects (Train)	9.297	13.948
Số aspects (Dev)	3.443	7.111
Số aspects (Test)	2.419	2.584
Tổng aspects	15.159	23.643
Số cặp Entity#Attribute	12 cặp	34 cặp
Nhãn tình cảm	3 lớp: positive, neutral, negative	3 lớp
Nguồn dữ liệu	Foody.vn	Agoda.com/ Booking.com

Bảng E.5: Phân bố cảm xúc trên các tập dữ liệu

Tập dữ liệu	**Positive**	**Neutral**	**Negative**	**Tổng aspects**
Restaurant Train	7.071 (76,1%)	1.556 (16,7%)	670 (7,2%)	9.297
Restaurant Dev	2.520 (73,2%)	692 (20,1%)	231 (6,7%)	3.443
Restaurant Test	1.669 (69,0%)	591 (24,4%)	159 (6,6%)	2.419
Hotel Train	10.229 (73,3%)	570 (4,1%)	3.149 (22,6%)	13.948
Hotel Dev	5.077 (71,4%)	226 (3,2%)	1.808 (25,4%)	7.111
Hotel Test	1.806 (69,9%)	133 (5,1%)	645 (24,9%)	2.584

Bảng E.6: Top 5 cặp Entity#Attribute phổ biến nhất đối với tập Restaurant Train

Hạng	Entity#Attribute		Số lượng	Tỷ lệ
1	FOOD#QUALITY		2.658	28,6%
2	FOOD#STYLE&OPTIONS		1.746	18,8%
3	FOOD#PRICES		1.401	15,1%
4	RESTAURANT#GENERAL		855	9,2%
5	SERVICE#GENERAL		795	8,6%

Bảng E.7: Top 5 cặp Entity#Attribute phổ biến nhất đối với tập Hotel Train

Hạng	Entity#Attribute	Số lượng	Tỷ lệ
1	SERVICE#GENERAL	1.912	13,7%
2	HOTEL#GENERAL	1.282	9,2%
3	LOCATION#GENERAL	1.196	8,6%
4	HOTEL#COMFORT	1.139	8,2%
5	ROOMS#DESIGN&FEATURES	916	6,6%

Mỗi mẫu trong tập VLSP-2018 có định dạng:

Dòng 1: Ký hiệu "#N" với N là số thứ tự mẫu.

Dòng 2: Văn bản đánh giá tiếng Việt.

Dòng 3: Danh sách các bộ ba {Entity#Attribute, sentiment} cách nhau bởi dấu phẩy.

Ví dụ: cho đánh giá "Hương vị thơm ngon, ăn cay rất thích, nêm nếm vừa miệng", chú thích là {FOOD#QUALITY, positive}, {FOOD#STYLE&OPTIONS, positive}.

Quy trình chuyển bộ ba thành bộ năm (e, a, s, h, t)

VLSP-2018 cung cấp chú thích bộ ba *{Entity#Attribute, Sentiment}*. Để nhất quán với bài toán bộ năm *o = (e, a, s, h, t)* được định nghĩa theo Bing Liu [1], luận án thực hiện bước ánh xạ mở rộng theo quy trình 5 bước:

Bảng E.8: Quy trình ánh xạ từ bộ ba VLSP-2018 sang bộ năm (e, a, s, h, t)

Thành phần	Nguồn trong VLSP-2018	Phương pháp trích xuất	Ví dụ cụ thể
e (Entity)	Phần Entity trong cặp Entity#Attribute	Tách chuỗi theo ký tự '#', lấy phần trước	FOOD#QUALITY → e = FOOD
a (Attribute)	Toàn bộ cặp Entity#Attribute	Lấy nguyên cặp, mã hóa label encoding	FOOD#QUALITY → a = 3
s (Sentiment)	Nhãn Sentiment gốc	positive → 1; negative → 0; neutral → loại bỏ	positive → s = 1
h (Holder)	Không chú thích tường minh	Gán theo từ khóa 3 cấp (xem Bảng E.9)	“gia đình mình” → h = 2
t (Time)	Không có trường ngày đăng bài	Suy từ từ khóa mùa/tháng (Bảng E.10); không có → 4 (không rõ)	“nghỉ hè” → t = 1 (Hạ)

Bước 1-Tách Entity#Attribute thành (e, a): Mỗi cặp được tách bằng ký tự '#': phần trước là e, toàn bộ cặp là a. Bảng ánh xạ E→{0..4} và A→{0..11} xây dựng từ tập Train, với E là tập hợp thực thể, A là tập hợp khía cạnh.

Bước 2 - Xử lý nhãn s: Nhãn gốc 3 lớp → 2 lớp: positive → 1; negative → 0; neutral → loại bỏ. Lý do: nhất quán với quy tắc loại mức trung lập áp dụng cho Amazon Food Reviews và TripAdvisor (Bảng 3.3); xem thêm cơ sở tâm lý học người tiêu dùng [168].

Bước 3 - Gán Holder h: VLSP-2018 không chú thích Holder tường minh. Gán theo hệ thống từ khóa 3 cấp (tại Bảng E.9). Mặc định h = 0 khi không có dấu hiệu.

Bước 4 - Trích xuất t: Từ từ khóa mùa/tháng trong câu đánh giá → phân mùa: tháng 1–3 = 0 (Xuân), 4–6 = 1 (Hạ), 7–9 = 2 (Thu), 10–12 = 3 (Đông); không có từ khóa → 4 (không rõ), do VLSP-2018 không cung cấp ngày đăng bài.

Bước 5 - Chuẩn hóa Z-score: Tính μ và σ từ tập Train, áp dụng cho Dev và Test. Kết quả: $\hat{O} \in \mathbb{R}^{nx4}$.

Bảng E.9: Tiêu chí chú thích *h*

Giá trị h	Loại holder	Từ khóa nhận dạng (Cấp 1)	Ý nghĩa	Xu hướng pos
0	khach_le	"tôi", "mình", không đề cập người đi cùng	Khách lẻ	65–70%
1	doi_ban	"đi với bạn", "tụi mình", "hai đứa"	Đôi bạn	76–82%
2	gia_dinh	"cả nhà", "đưa con", "vợ chồng", "bọn trẻ"	Gia đình	80–85%
3	doanh_nhan	"tiếp khách", "đối tác", "họp", "công ty"	Doanh nhân	55–63%
4	phuot_thu	"du lịch", "ghé qua", "phượt", "khám phá"	Phượt thủ	78–84%

Hệ thống quyết định 3 cấp cho h:

Cấp 1 - Từ khóa rõ ràng: "gia đình mình", "hai đứa nhỏ" → h = 2.

Cấp 2 - Dấu hiệu gián tiếp: "tiệc sinh nhật", "họp mặt công ty" → suy luận.

Cấp 3 - Mặc định: không có dấu hiệu → h = 0 (khach_le).

Bảng E.10: Tiêu chí chú thích *t* theo mùa

Giá trị t	Mùa	Khoảng thời gian	Đặc trưng hành vi tiêu dùng	Xu hướng pos
0	Xuân	Tháng 1–3 (Tết, lễ hội)	Tụ họp gia đình, tiệc tất niên, đầu năm nhiều kỳ vọng	72–76%
1	Hạ	Tháng 4–6 (nghỉ hè)	Cao điểm du lịch, nhiều khách nhất, trải nghiệm mới	78–84%
2	Thu	Tháng 7–9 (mưa nhiều miền Nam)	Trung bình, ít du lịch hơn, khách quen địa phương	70–74%
3	Đông	Tháng 10–12 (cuối năm)	Vắng khách nhất, tiêu chuẩn so sánh cao hơn	63–68%

Hệ thống quyết định 3 cấp cho t:

Cấp 1 - Từ khóa mùa: "hè", "nghỉ hè", "mùa đông", "tết" → gán trực tiếp.

Cấp 2 - Từ khóa tháng: “tháng 5”, “tháng 12” → suy luận mùa.

Cấp 3 - Không có dấu hiệu: t = 4 (không rõ), do VLSP-2018 không cung cấp ngày đăng bài.

Bảng E.11: Độ đo đánh giá chú thích và ngưỡng mục tiêu

Thành phần	**Loại biến**	**Độ đo IAA**	**Ngưỡng mục tiêu**
h (Holder)	Danh nghĩa (5 lớp)	Cohen's Kappa (κ) [79]	$\kappa \geq 0{,}80$
t (Time/Mùa)	Thứ tự (4 mùa)	Weighted Kappa (κ_w) [198]	$\kappa_w \geq 0{,}75$
s (Sentiment)	Danh nghĩa (2 lớp)	Cohen's Kappa (κ) [79]	$\kappa \geq 0{,}80$
e#a (Entity#Attr)	Danh nghĩa (12/34 lớp)	Fleiss' Kappa [199]	$\kappa \geq 0{,}80$

E.5. Tổng hợp các tập dữ liệu

Bảng E.12: Bảng tổng hợp các tập dữ liệu dùng trong thực nghiệm

Tập dữ liệu	**Lĩnh vực**	**Ngôn ngữ**	**Kích thước**	**ρ(O)**	**Sử dụng thực nghiệm**
yelpHotelData	Khách sạn (Yelp)	Tiếng Anh	688.329 reviews	≈ 1,7	Chương 3, 4
yelpResData	Nhà hàng (Yelp)	Tiếng Anh	788.471 reviews	≈ 1,9	Chương 3, 4
Amazon	Thương mại điện tử	Tiếng Anh	568.454 reviews	≈ 5,4	Chương 2
TripAdvisor	Khách sạn	Tiếng Anh	108.891 reviews	≈ 4	Chương 2
VLSP-2018 Restaurant	Nhà hàng VN	Tiếng Việt	4.751 mẫu / 12 aspects	≈ 10,5	Phụ lục F

VLSP-2018 Hotel	Khách sạn VN	Tiếng Việt	5.600 mẫu / 34 aspects	≈ 3,2	Phụ lục F

Bốn tập dữ liệu tiếng Anh và 2 tập tiếng Việt được sử dụng tổng cộng trong các kịch bản thực nghiệm của luận án, đảm bảo đa dạng về: (1) lĩnh vực (khách sạn, nhà hàng, sản phẩm thực phẩm); (2) ngôn ngữ (tiếng Anh, tiếng Việt); (3) mức độ mất cân bằng (từ $\rho \approx 1{,}7$ đến mất cân bằng nghiêm trọng $\rho \approx 10{,}5$); (4) cấp độ chú thích (cấp tài liệu, cấp câu, cấp khía cạnh). Sự đa dạng này cho phép đánh giá toàn diện tính tổng quát hóa của các mô hình rừng đề xuất trên các điều kiện thực tế khác nhau.

Phụ lục F
KẾT QUẢ THỰC NGHIỆM TRÊN VLSP-2018

Phụ lục trình bày kết quả thực nghiệm của ba mô hình rừng đề xuất (F_{CF}, F_{DF}, F_{MF}) và năm mô hình cơ sở (Logistic Regression, SVM tuyến tính, Naive Bayes, Random Forest, Gradient Boosting), cùng mô hình tham chiếu PhoBERT (mục F.6), trên dữ liệu gốc VLSP-2018 SA (Restaurant, Hotel) [197]. Toàn bộ thực nghiệm được thực hiện bằng kịch bản R, dùng chung các hàm phân cụm, huấn luyện và đánh giá với Phụ lục H và thực nghiệm so sánh thống nhất, nhằm bảo đảm quy trình đồng nhất giữa dữ liệu tiếng Anh và tiếng Việt.

F.1. Dữ liệu và quy trình xây dựng bộ năm quan điểm

Dữ liệu gốc gồm các tập dữ liệu train, dev và test (được định dạng CSV, mỗi cột ứng với một cặp Entity#Attribute, mã 0 = không đề cập, 1 = positive, 2 = negative, 3 = neutral). Mỗi cặp (đánh giá, khía cạnh) có nhãn được chuyển thành một bộ năm o = (e, a, s, h, t) theo quy tắc sau:

- e: phần Entity của cặp Entity#Attribute (ví dụ FOOD); a: toàn bộ cặp Entity#Attribute (ví dụ FOOD#QUALITY);

- s: positive → 1, negative → 0; nhãn neutral được loại bỏ, cùng quy tắc với Amazon Food Reviews và TripAdvisor (Bảng 3.3);

- h: suy tự động bằng so khớp từ khóa trong câu đánh giá theo Bảng E.9 (gia đình, đôi bạn, doanh nhân, phượt thủ); không có dấu hiệu → h = 0 (khách lẻ);

- t: suy tự động bằng so khớp từ khóa mùa/tháng trong câu theo Bảng E.10 (cấp 1–2); VLSP-2018 không cung cấp ngày đăng bài nên khi không có từ khóa → t = 4 (không rõ).

Số liệu chi tiết Bảng F.1, Bảng F.2 cho thấy phần lớn bộ năm nhận giá trị mặc định: 92,9% (Restaurant) và 74,3% (Hotel) có h = khách lẻ; 96,6% (Restaurant) và 97,2% (Hotel) có t = không rõ. Như vậy thông tin thực sự phân biệt trong véc-tơ đặc trưng chủ yếu đến từ hai thành phần e, a, đây là giới hạn cốt lõi của việc áp dụng biểu diễn bộ năm cho VLSP-2018, vốn không chú thích Holder và Time.

F.2. Mã hóa đặc trưng và quy trình thực nghiệm

- Mã hóa: e, a, h, t được mã hóa target encoding (làm trơn Laplace, m = 10, chỉ ước lượng trên tập Train) và chuẩn hóa Z-score theo tập Train (Phụ lục G–J).

- Chia tập: dùng đúng các tập Train/Dev/Test chính thức của VLSP-2018; Dev dùng để chọn ngưỡng quyết định (tối đa Balanced Accuracy) cho mọi mô hình và làm tập xác thực cho bộ kết hợp Stacking của F_{MF}.

- F_{CF}: một rừng ngẫu nhiên 300 cây với Bootstrap cân bằng lớp (BalancedBoot). F_{DF}: DBSCAN Euclidean với eps thích ứng theo phân vị khoảng cách k-láng giềng, mỗi cụm một rừng BalancedBoot, kết hợp bằng trung bình xác suất. F_{MF}: ma trận quan hệ W từ rừng không giám sát (thật/giả) trên mẫu con 5.000 bộ năm, DBSCAN trên khoảng cách đa tạp $dist_{\mathcal{M}} = \overline{1 - W}$ mỗi cụm một rừng BalancedBoot, kết hợp bằng Stacking (meta-learner rừng ngẫu nhiên trên tập Dev).

- Mô hình cơ sở: Logistic Regression [203] (glm, trọng số lớp nghịch tần suất), SVM tuyến tính [86] (e1071, hàm nhân tuyến tính, trọng số lớp), Naive Bayes [204] (e1071), Random Forest [65] (randomForest, 300 cây, Bootstrap mặc định) và Gradient Boosting [84] (gbm, 300 cây, độ sâu tương tác 3, hàm mất mát Bernoulli). Năm mô hình này dùng cùng véc-tơ bộ năm, cùng cách chia và cùng quy tắc chọn ngưỡng như ba mô hình rừng đề xuất.

- Về lựa chọn mô hình cơ sở: bộ mô hình cơ sở trên VLSP-2018 bao gồm đủ ba mô hình học máy dùng trên dữ liệu Yelp tại Chương 3, Chương 4 (Random Forest, Gradient Boosting, SVM), bổ sung hai mô hình tuyến tính, xác suất thông dụng cho phân loại văn bản (Logistic Regression, Naive Bayes). SVM tuyến tính là thuật toán SVM [86] với hàm nhân tuyến tính; SVM tại Chương 3, Chương 4 dùng cấu hình của các bài báo đã công bố (gói e1071, Bảng D.1), còn tại phụ lục này và trong thực nghiệm so sánh thống nhất (Phụ lục J, Bảng J.8) dùng hàm nhân tuyến tính vì phù hợp với véc-tơ đặc trưng ít chiều và có chi phí huấn luyện thấp. Nhóm mô hình học sâu, Transformer dùng văn bản gốc (LSTM, BiLSTM-Attention, BERT, RoBERTa trên Yelp) tương ứng với PhoBERT trên tiếng Việt (mục F.6), vì BERT, RoBERTa bản gốc được tiền huấn luyện cho tiếng Anh. Như vậy, trên cả hai ngôn ngữ, các mô hình dùng bộ năm được so sánh với nhau trong cùng điều kiện, còn các mô hình dùng văn bản gốc được trình bày như mốc tham chiếu.

- Độ đo: Accuracy, F1-macro, F1-weighted, F1-pos, F1-neg, Balanced Accuracy (BA), MCC, ROC-AUC; kiểm định McNemar từng cặp với hiệu chỉnh Holm. Thời gian là thời gian huấn luyện trên CPU.

- Khi một cụm chỉ gồm các điểm trùng nhau (mọi đặc trưng là hằng số), rừng ngẫu nhiên không thể tách nút; cụm đó được dự đoán bằng tỷ lệ positive của cụm (VLSP-2018 chỉ có khoảng 120 tổ hợp (e, a, h, t) khác nhau trên tập Restaurant).

F.3. Kết quả trên VLSP-2018 Restaurant (ρ ≈ 10,5)

Số liệu chi tiết tại Bảng F.3, Bảng F.4 (trình bày ở cuối Phụ lục F).

F.4. Kết quả trên VLSP-2018 Hotel (ρ ≈ 3,2)

Số liệu chi tiết tại Bảng F.5, Bảng F.6 (trình bày ở cuối Phụ lục F).

F.5. Thảo luận

Trên VLSP-2018 Restaurant (ρ ≈ 10,5), mọi mô hình chỉ đạt Balanced Accuracy 0,49–0,66 và ROC-AUC 0,50–0,69, tức khả năng phân biệt hai lớp rất hạn chế. F_{DF} đạt BA cao nhất (0,6596) và MCC cao nhất (0,1799); F_{CF} đạt MCC 0,1602, cao hơn năm mô hình cơ sở (MCC từ -0,0042 đến 0,1438). Random Forest và F_{MF} không học được lớp negative (BA 0,4988 và 0,4919, ROC-AUC 0,5032 và 0,5150): Random Forest dùng Bootstrap mặc định nên thiên về lớp đa số khi ρ ≈ 11, còn bộ kết hợp Stacking của F_{MF} huấn luyện trên Dev với số mẫu negative nhỏ (231) cũng thiên hẳn về lớp positive; điều này cho thấy vai trò của cơ chế Bootstrap cân bằng lớp mà F_{CF}, F_{DF} sử dụng. Ở ngưỡng 0,5 (Bảng F.4), SVM tuyến tính, Random Forest và F_{DF} dự đoán gần như toàn bộ là positive (Accuracy 0,9103–0,9130, gần bằng tỷ lệ lớp đa số, F1-neg = 0), cho thấy Accuracy không phải độ đo phù hợp trên tập mất cân bằng mạnh này.

Trên VLSP-2018 Hotel (ρ ≈ 3,2), các mô hình đạt BA 0,69–0,73 và ROC-AUC 0,71–0,77. Logistic Regression đạt BA cao nhất (0,7300), tiếp theo là Gradient Boosting (0,7270), F_{DF} (0,7239) và F_{MF} (0,7171); Random Forest đạt Accuracy cao nhất (0,7666) nhưng BA thấp nhất (0,6906) do thiên về lớp positive. F_{DF} và F_{MF} có Accuracy, F1-macro cao nhất trong ba mô hình rừng đề xuất (0,7503/0,7031 và 0,7556/0,7029). Kiểm định McNemar (hiệu chỉnh Holm trên 28 cặp) cho thấy F_{DF} và F_{MF} đúng nhiều hơn có ý nghĩa so với F_{CF}, SVM tuyến tính và Naive Bayes ($p < 0,001$), nhưng không khác biệt có ý nghĩa so với Logistic Regression, Random Forest và Gradient Boosting (p sau hiệu chỉnh từ 0,19 đến 1,0) cũng như giữa F_{DF} với F_{MF} ($p = 1,0$).

Kết quả trên dữ liệu gốc cho thấy: (i) trên VLSP-2018, cực tính cảm xúc được quyết định chủ yếu bởi nội dung câu đánh giá, trong khi véc-tơ bộ năm chỉ mang thông tin về khía cạnh (e, a) và hai thành phần h, t phần lớn nhận giá trị mặc định (Bảng F.2); vì vậy cả ba mô hình rừng lẫn năm mô hình cơ sở đều bị giới hạn ở mức phân biệt thấp; (ii) trong giới hạn đó, trên tập Hotel cơ chế phân cụm của F_{DF} và F_{MF} giúp cải thiện Accuracy, F1-macro so với F_{CF} (trên tập Restaurant, F_{DF} chỉ cải thiện Balanced Accuracy và MCC), phù hợp với nhận định của luận án về vai trò của phân cụm theo mật độ; (iii) trên tập Hotel, F_{DF} và F_{MF} không khác biệt có ý nghĩa thống kê so với Logistic Regression, Random Forest và Gradient Boosting (kiểm định McNemar–Holm), tức trên đặc trưng bộ năm của VLSP-2018, các mô hình rừng đề xuất đạt mức tương đương các mô hình học máy thông dụng. Kết quả này củng cố hạn chế đã nêu tại §5.1 (phụ thuộc vào chất lượng trích xuất bộ năm) và hướng phát triển tại §5.2 (kết hợp biểu diễn ngữ nghĩa PhoBERT với cấu trúc rừng).

F.6. Mô hình tham chiếu PhoBERT

PhoBERT (checkpoint vinai/phobert-base) được tinh chỉnh cho bài toán phân loại nhị phân trên cùng các cặp (đánh giá, khía cạnh), cùng nhãn và cùng tập Train/Dev/Test chính thức như Bảng F.1. Khác với tám mô hình tại Bảng F.3 - F.6 dùng véc-tơ bộ năm (e, a, h, t), đầu vào của PhoBERT là văn bản "khía cạnh + câu đánh giá" (ví dụ: "khía cạnh food quality . <câu đánh giá>"). Vì vậy PhoBERT được trình bày như mốc tham chiếu cho mức thông tin chứa trong văn bản gốc, không nằm trong cùng nhóm so sánh với các mô hình dùng bộ năm.

Cấu hình: độ dài chuỗi tối đa 128, kích thước batch 16, 2 epoch, tốc độ học 2×10^{-5} (AdamW), dừng sớm theo tổn thất trên Dev (patience = 1), hạt giống 2026; ngưỡng quyết định chọn trên Dev theo Balanced Accuracy như các mô hình khác. Huấn luyện trên CPU 8 luồng (PyTorch 2.2.0, không dùng GPU), bộ nhớ đỉnh của tiến trình khoảng 3,96 GB. Tổn thất (train/Dev) sau 2 epoch: Restaurant 0,2068/0,1890; Hotel 0,2687/0,3631.

Số liệu chi tiết tại Bảng F.7 (trình bày ở cuối Phụ lục F).

Khi dùng văn bản gốc, PhoBERT đạt Balanced Accuracy 0,7906 và ROC-AUC 0,8603 trên Restaurant, 0,8323 và 0,9160 trên Hotel. Kết quả này cho thấy phần lớn thông tin về cực tính cảm xúc của VLSP-2018 nằm trong nội dung câu đánh giá,

nhất quán với nhận định (i) tại mục F.5 về giới hạn của véc-tơ bộ năm khi hai thành phần h, t phần lớn nhận giá trị mặc định. Về chi phí, thời gian huấn luyện PhoBERT trên CPU là khoảng 5,5–6,5 giờ mỗi tập, trong khi các mô hình dùng bộ năm cần từ dưới 1 giây đến khoảng 5 phút (Bảng F.3- Bảng J.5). Hai đặc điểm này là cơ sở cho hướng phát triển tại §5.2: kết hợp biểu diễn ngữ nghĩa từ PhoBERT với cấu trúc rừng để giữ khả năng diễn giải và chi phí suy luận thấp của các mô hình đề xuất.

F.7. Mục đích, nguồn dữ liệu, tái lập và giới hạn

- *Mục đích:* phụ lục này kiểm tra khả năng áp dụng của quy trình bộ năm và ba mô hình rừng đề xuất trên một ngôn ngữ khác (tiếng Việt), với dữ liệu có chú thích khía cạnh công khai của cộng đồng. Đây là thực nghiệm bổ sung, không thay thế các kết quả trên dữ liệu tiếng Anh tại Chương 2, Chương 3, Chương 4 và không nhằm so sánh với bảng xếp hạng của VLSP-2018.

- *Nguồn của dữ liệu bộ năm*: từ sáu tệp gốc chính thức của VLSP-2018 SA [197] (Restaurant, Hotel × Train, Dev, Test). Mỗi dòng ứng với một cặp (đánh giá, khía cạnh) có nhãn positive hoặc negative, gồm các cột: mã đánh giá, tập chia (train/dev/test), e, a, s, h, t và văn bản đánh giá gốc; số dòng của từng tập khớp với cột "Số bộ năm dùng" tại Bảng F.1.

- *Cách xây dựng và cách suy ra h, t:* toàn bộ quá trình là tự động, không có bước gán nhãn thủ công. Ba thành phần e, a, s lấy trực tiếp từ chú thích chính thức của VLSP-2018 (tách cặp Entity#Attribute; nhãn positive, negative; loại nhãn neutral). Hai thành phần h, t được suy bằng so khớp từ khóa (biểu thức chính quy, không phân biệt chữ hoa, chữ thường) trên văn bản đánh giá theo bộ từ khóa của Bảng E.9, Bảng E.10, có bổ sung một số biến thể thường gặp (ví dụ "bố mẹ", "ông bà", "bạn bè", "người yêu", "đồng nghiệp", "công tác", "giáng sinh", "trung thu"). Khi một câu chứa từ khóa của nhiều nhóm, h được gán theo thứ tự ưu tiên gia đình, doanh nhân, đôi bạn, phượt thủ; không khớp từ khóa nào thì h = 0 (khách lẻ) và t = 4 (không rõ). Do dùng so khớp từ khóa nên kết quả có nhiễu (ví dụ cụm "dịch vụ du lịch" khiến h nhận giá trị phượt thủ) và chưa có chú thích chuẩn để đo độ chính xác của bước suy diễn; các ngưỡng đồng thuận tại Bảng E.11 là ngưỡng mục tiêu cho chú thích thủ công trong nghiên cứu tiếp theo, chưa áp dụng trong phụ lục này.

- *Khả năng dùng chung cho mô hình cổ điển và hiện đại:* mỗi dòng giữ đồng thời bộ năm và văn bản gốc với cùng mã đánh giá và cùng tập chia chính thức, nên cùng một tệp được dùng cho cả tám mô hình trên đặc trưng bộ năm (Bảng F.3–F.6) lẫn mô hình dùng văn bản PhoBERT (Bảng F.7, đầu vào "khía cạnh + câu đánh giá"). Nhờ vậy mọi mô hình được đánh giá trên đúng cùng các cặp (đánh giá, khía cạnh) và cùng tập Test; tệp có thể dùng lại cho các mô hình khác (LSTM, BiLSTM-Attention hoặc mô hình ngôn ngữ lớn) mà không phải xây dựng lại dữ liệu.

- *Lý do PhoBERT không được so sánh trực tiếp*: PhoBERT dùng toàn bộ văn bản và tri thức tiền huấn luyện từ kho văn bản tiếng Việt lớn, trong khi tám mô hình tại Bảng F.3–F.6 chỉ dùng véc-tơ bốn đặc trưng (e, a, h, t); hai nhóm khác nhau về không gian đầu vào, về lượng thông tin được tiếp cận và về chi phí tính toán (khoảng 5,5–6,5 giờ CPU mỗi tập so với từ dưới 1 giây đến khoảng 5 phút). Vì vậy, giống cách trình bày BERT, RoBERTa tại Chương 3, Chương 4, PhoBERT được dùng làm mốc tham chiếu cho lượng thông tin chứa trong văn bản; phép so sánh có kiểm soát chỉ thực hiện giữa các mô hình cùng đầu vào bộ năm (Bảng F.3–F.6 và kiểm định McNemar).

- *Tái lập:* một kịch bản thực nghiệm trên phần mềm R (với hạt giống 2026, 300 cây cho các mô hình rừng) đọc sáu tệp gốc, sinh hai tệp bộ năm nói trên, huấn luyện tám mô hình dùng bộ năm và ghi các bảng kết quả, kết quả tại ngưỡng 0,5, thống kê thành phần h, t và kiểm định McNemar. PhoBERT được tinh chỉnh bằng thư viện transformers (nền tảng PyTorch), gọi từ kịch bản trên phần mềm R.

Bài toán tại phụ lục này là phân loại nhị phân trên từng cặp (đánh giá, khía cạnh) đã biết khía cạnh, khác định dạng gốc của VLSP-2018 Shared Task (phát hiện khía cạnh và cực tính) nên kết quả không so sánh trực tiếp với bảng xếp hạng chính thức của VLSP-2018.

BẢNG SỐ LIỆU CỦA PHỤ LỤC F

Bảng F.1: Số cặp (đánh giá, khía cạnh) và số bộ năm sau khi loại nhãn neutral

Tập	Tổng số cặp	Positive	Neutral	Negative	Số bộ năm dùng	ρ (pos/neg)
Restaurant – train	9.297	7.071	1.556	670	7.741	10,55
Restaurant – dev	3.443	2.520	692	231	2.751	10,91
Restaurant – test	2.419	1.669	591	159	1.828	10,50
Hotel – train	13.948	10.229	570	3.149	13.378	3,25
Hotel – dev	7.111	5.077	226	1.808	6.885	2,81
Hotel – test	2.584	1.806	133	645	2.451	2,80

Bảng F.2: Phân bố thành phần h và t suy từ từ khóa trên tập Train

Tập	h: khách lẻ	h: đôi bạn	h: gia đình	h: doanh nhân	h: phượt thủ	t: Xuân	t: Hạ	t: Thu	t: Đông	không rõ
Restaurant	7.192	213	182	70	84	83	158	5	14	7.481
Hotel	9.941	206	2.413	183	635	142	176	18	40	13.002

Bảng F.3: Kết quả trên tập Test VLSP-2018 Restaurant (1.828 bộ năm; ngưỡng tối ưu BA trên Dev)

Mô hình	Accuracy	F1-macro	F1-weighted	F1-pos	F1-neg	BA	MCC	ROC-AUC	Ngưỡng	Thời gian (s)
Logistic Regression [203]	0,4469	0,3908	0,5436	0,5757	0,2058	0,6175	0,1357	0,6713	0,70	0,0
SVM (tuyến tính) [86]	0,4551	0,3965	0,5519	0,5847	0,2083	0,6220	0,1404	0,6777	0,95	3,9

Mô hình	Accuracy	F1-macro	F1-weighted	F1-pos	F1-neg	BA	MCC	ROC-AUC	Ngưỡng	Thời gian (s)
Naive Bayes [204]	0,6668	0,5079	0,7389	0,7876	0,2281	0,6212	0,1438	0,6190	0,94	0,3
Random Forest [65]	0,8900	0,4900	0,8631	0,9417	0,0383	0,4988	-0,0042	0,5032	0,85	1,3
Gradient Boosting [84]	0,4759	0,4088	0,5734	0,6080	0,2096	0,6220	0,1391	0,6660	0,95	1,9
Rừng phân loại (F_{CF})	0,6406	0,5001	0,7190	0,7651	0,2352	0,6382	0,1602	0,6850	0,37	0,3
Rừng mật độ (F_{DF})	0,5602	0,4635	0,6516	0,6912	0,2357	0,6596	0,1799	0,6871	0,59	0,9
Rừng đa tạp (F_{MF})	0,8463	0,4913	0,8423	0,9162	0,0664	0,4919	-0,0171	0,5150	0,95	289,0

Bảng F.4: Đối chiếu tại ngưỡng cố định 0,5 – VLSP-2018 Restaurant

Mô hình	Accuracy	F1-macro	F1-pos	F1-neg	BA	MCC
Logistic Regression [203]	0,6422	0,5044	0,7658	0,2431	0,6504	0,1742
SVM (tuyến tính) [86]	0,9130	0,4773	0,9545	0,0000	0,5000	0,0000
Naive Bayes [204]	0,8217	0,5213	0,9005	0,1421	0,5268	0,0463
Random Forest [65]	0,9103	0,4765	0,9530	0,0000	0,4985	-0,0162
Gradient Boosting [84]	0,9054	0,4809	0,9503	0,0114	0,4987	-0,0082

Mô hình	Accuracy	F1-macro	F1-pos	F1-neg	BA	MCC
Rừng phân loại (F_{CF})	0,6247	0,4937	0,7513	0,2361	0,6437	0,1650
Rừng mật độ (F_{DF})	0,9130	0,4773	0,9545	0,0000	0,5000	0,0000
Rừng đa tạp (FMF)	0,9054	0,4809	0,9503	0,0114	0,4987	-0,0082

Ghi chú: tại ngưỡng 0,5, rừng mật độ (F_{DF}) và SVM tuyến tính cùng dự đoán toàn bộ bộ năm là positive nên có chỉ số trùng nhau; rừng đa tạp (F_{MF}) và Gradient Boosting cùng dự đoán 16 bộ năm là negative (1 dự đoán đúng) nhưng ở các bộ năm khác nhau (hai mô hình có 32 bộ năm dự đoán khác nhau), nên ma trận nhầm lẫn và các chỉ số trùng nhau. Kết quả đã được kiểm tra lại bằng cách chạy lại toàn bộ kịch bản với cùng hạt giống 2026.

Bảng F.5: Kết quả trên tập Test VLSP-2018 Hotel (2.451 bộ năm; ngưỡng tối ưu BA trên Dev)

Mô hình	Accuracy	F1-macro	F1-weighted	F1-pos	F1-neg	BA	MCC	ROC-AUC	Ngưỡng	Thời gian (s)
Logistic Regression [203]	0,7475	0,7041	0,7578	0,8174	0,5909	0,7300	0,4235	0,7666	0,43	0,0
SVM (tuyến tính) [68]	0,6879	0,6614	0,7063	0,7561	0,5666	0,7159	0,3818	0,7658	0,80	20,8
Naive Bayes [204]	0,6867	0,6594	0,7050	0,7557	0,5631	0,7126	0,3761	0,7567	0,86	0,7
Random Forest [65]	0,7666	0,6939	0,7646	0,8431	0,5446	0,6906	0,3881	0,7063	0,94	1,6

Mô hình	Accuracy	F1-macro	F1-weighted	F1-pos	F1-neg	BA	MCC	ROC-AUC	Ngưỡng	Thời gian (s)
Gradient Boosting [84]	0,7475	0,7028	0,7574	0,8180	0,5876	0,7270	0,4193	0,7663	0,69	1,3
Rừng phân loại (F_{CF})	0,6879	0,6612	0,7062	0,7563	0,5661	0,7154	0,3810	0,7544	0,67	1,0
Rừng mật độ (F_{DF})	0,7503	0,7031	0,7592	0,8215	0,5848	0,7239	0,4168	0,7503	0,53	1,6
Rừng đa tạp (F_{MF})	0,7556	0,7029	0,7622	0,8280	0,5779	0,7171	0,4112	0,7495	0,91	277,9

Bảng F.6: Đối chiếu tại ngưỡng cố định 0,5 – VLSP-2018 Hotel

Mô hình	Accuracy	F1-macro	F1-pos	F1-neg	BA	MCC
Logistic Regression [203]	0,7050	0,6704	0,7772	0,5637	0,7111	0,3777
SVM (tuyến tính) [86]	0,7870	0,6892	0,8636	0,5149	0,6721	0,3981
Naive Bayes [204]	0,7401	0,6740	0,8208	0,5271	0,6791	0,3489
Random Forest [65]	0,7891	0,7018	0,8631	0,5404	0,6869	0,4148
Gradient Boosting [84]	0,7895	0,7022	0,8634	0,5409	0,6872	0,4158
Rừng phân loại (F_{CF})	0,6928	0,6645	0,7619	0,5670	0,7158	0,3823
Rừng mật độ (F_{DF})	0,7882	0,7010	0,8625	0,5395	0,6864	0,4130

Mô hình	Accuracy	F1-macro	F1-pos	F1-neg	BA	MCC
Rừng đa tạp (F_{MF})	0,7813	0,6658	0,8623	0,4693	0,6483	0,3668

Bảng F.7: Kết quả PhoBERT trên tập Test VLSP-2018 (Restaurant: 1.828 bộ năm; Hotel: 2.451 bộ năm)

Tập	Ngưỡng	Accuracy	F1-macro	F1-weighted	F1-pos	F1-neg	BA	MCC	ROC-AUC	Thời gian (s)
Restaurant	Tối ưu BA trên Dev	0,7735	0,6226	0,8197	0,8613	0,3839	0,7906	0,3646	0,8603	23.447,9
Restaurant	0,5	0,9190	0,6390	–	0,9570	0,3211	0,6029	–	0,8603	–
Hotel	Tối ưu BA trên Dev	0,8233	0,7944	0,8309	0,8716	0,7172	0,8323	0,6091	0,9160	19.747,1
Hotel	0,5	0,8633	0,8077	–	0,9111	0,7043	0,7847	–	0,9160	–

Ghi chú: ROC-AUC không phụ thuộc ngưỡng nên giống nhau ở hai dòng của cùng một tập; "–": độ đo không được xuất ở lần chạy tại ngưỡng 0,5. Mốc "luôn dự đoán positive" có Accuracy 0,9130 (Restaurant) và 0,7368 (Hotel).

Phụ lục G

SO SÁNH BA MÔ HÌNH F_{CF}, F_{DF}, F_{MF} TRÊN CÙNG NĂM KỸ THUẬT XỬ LÝ MẤT CÂN BẰNG LỚP

Phụ lục này so sánh năm kỹ thuật xử lý mất cân bằng lớp trên cả ba mô hình đề xuất của luận án - F_{CF} (Chương 2), F_{DF} (Chương 3) và F_{MF} (Chương 4) nhằm làm rõ hai vấn đề: (i) "Ba mô hình F_{CF}, F_{DF}, F_{MF} được đánh giá lại trên cùng một tập dữ liệu, cùng tỉ lệ chia và cùng quy trình chia dữ liệu để việc so sánh được đồng nhất"; (ii) "Recall được duy trì ở mức 0,62–0,76 xuyên suốt ba mô hình. So sánh thêm với các kỹ thuật quen thuộc (class weights, balanced random forest, SMOTE, điều chỉnh ngưỡng quyết định) và báo cáo kèm F1-macro, PR-AUC hoặc MCC bên cạnh Accuracy."

G.1. Thiết lập thực nghiệm

Cả ba mô hình được huấn luyện và đánh giá trên cùng một tập dữ liệu, cùng tỉ lệ chia (60% Train/ 20% Validation/ 20% Test, phân chia ngẫu nhiên với set.seed(101)) và cùng cỡ mẫu N = 112.275 (giống Phụ lục J), đáp ứng trực tiếp yêu cầu (i) nêu trên. Kiến trúc từng mô hình giữ nguyên như đã trình bày trong luận án:

- Mô hình F_{CF}: một rừng ngẫu nhiên duy nhất, huấn luyện trên toàn bộ tập Train (không phân cụm).

- Mô hình F_{DF}: DBSCAN Euclidean trên không gian đặc trưng của tập Train, sau đó K rừng ngẫu nhiên riêng biệt theo cụm, kết hợp bằng trung bình xác suất qua K rừng (density averaging).

- Mô hình F_{MF}: DBSCAN trên không gian đa tạp (khoảng cách suy ra từ một rừng không giám sát phân biệt dữ liệu thật/giả), sau đó K rừng riêng biệt theo cụm, kết hợp bằng Stacking qua một meta-learner huấn luyện trên Validation.

Riêng với F_{MF}, ma trận khoảng cách đa tạp W có kích thước N×N; ở N = 112.275, ma trận này đòi hỏi xấp xỉ 100 GB bộ nhớ, không khả thi trên máy tính thông thường - đúng như hạn chế đã được luận án tự nêu tại mục 4.5.1 (hướng mở rộng quy mô), trong đó có đề xuất giải pháp lấy mẫu con (subsampling) trên tập huấn luyện trước khi xây W. Thực nghiệm này áp dụng cụ thể giải pháp đó: riêng bước phân cụm không gian đa tạp và huấn luyện K rừng cơ sở của F_{MF} dùng một tập con ngẫu nhiên phân tầng theo lớp gồm 5.000 dòng trích từ tập Train

(N_MF_SUBSAMPLE = 5.000); tập Validation và Test vẫn dùng đầy đủ cho cả ba mô hình để việc đánh giá cuối cùng công bằng và có thể so sánh trực tiếp giữa F_{CF}, F_{DF}, F_{MF}.

Năm biến thể xử lý mất cân bằng lớp được áp dụng khi huấn luyện mỗi rừng cơ sở (một rừng duy nhất với F_{CF}; từng rừng theo cụm với F_{DF}/F_{MF}): Baseline (bootstrap mặc định, ngưỡng cố định 0,5), ThresholdOnly (bootstrap mặc định, ngưỡng hiệu chỉnh trên Validation theo Balanced Accuracy), ClassWeight, BalancedBoot (bootstrap cân bằng lớp qua sampsize/strata) và SMOTE (oversampling SMOTE bằng gói smotefamily, K=5 láng giềng, trên tập huấn luyện của mỗi rừng cơ sở). Riêng với F_{MF}, năm biến thể này chỉ áp dụng cho K rừng cơ sở; meta-learner của Stacking giữ nguyên mặc định ở mọi biến thể, nhằm tách bạch rõ đóng góp của việc xử lý mất cân bằng ở tầng rừng cơ sở, tránh nhầm lẫn với hiệu ứng của Stacking. Việc hiệu chỉnh ngưỡng luôn áp dụng trên xác suất đầu ra cuối cùng của từng mô hình (sau trung bình K rừng với F_{CF}/F_{DF}, sau meta-learner với F_{MF}), đảm bảo nhất quán giữa ba mô hình. Độ đo báo cáo cho mỗi tổ hợp (mô hình, biến thể): Accuracy, Precision, Recall, Specificity, F1, F1-macro, Balanced Accuracy, MCC và PR-AUC (diện tích dưới đường cong Precision-Recall, xấp xỉ hình thang trên lưới ngưỡng 0,01 - 0,99 của tập Test, không phụ thuộc ngưỡng được chọn). Toàn bộ tiền xử lý, mã hóa đặc trưng và phân cụm DBSCAN được giữ nguyên giữa các biến thể, nên khác biệt giữa các hàng trong Bảng G.1, Bảng G.2 chỉ đến từ cách xử lý mất cân bằng lớp.

Đây là thực nghiệm bổ sung, không thay thế số liệu ba mô hình đã báo cáo chính thức trong luận án (Chương 2, Chương 3 và Chương 4) hay tại Phụ lục J.

G.2. Kết quả trên tập Hotel

Số liệu chi tiết tại Bảng G.1 (trình bày ở cuối Phụ lục G).

G.3. Kết quả trên tập Restaurant

Số liệu chi tiết tại Bảng G.2 (trình bày ở cuối Phụ lục G).

Lưu ý về độ đo PR-AUC: chỉ số này được tính bằng quy tắc hình thang trên lưới ngưỡng 0,01 - 0,99. Với F_{CF} (một rừng duy nhất), nhiều quan sát có xác suất dự đoán bằng 1, nên ngay tại ngưỡng 0,99 Recall vẫn còn cao và đoạn đường Precision–Recall ứng với Recall thấp không được tính vào diện tích; đây nhiều khả năng là lý do PR-AUC của F_{CF} (và một phần F_{MF}) thấp hơn tỷ lệ lớp positive dù Balanced Accuracy tương đương F_{DF}. Vì vậy PR-AUC chỉ dùng để so sánh các biến thể của

cùng một mô hình; so sánh giữa các mô hình dựa trên Balanced Accuracy, F1-macro và MCC.

G.4. Thảo luận

Kết quả với Baseline (không xử lý mất cân bằng) tái hiện nhất quán ở cả ba mô hình đề xuất trên cả hai tập dữ liệu: Recall rất cao (0,7719-0,9837) nhưng Specificity rất thấp (0,0746- 0,4950), kéo theo Balanced Accuracy thấp hơn đa số các biến thể có xử lý mất cân bằng (mức giảm rõ rệt nhất ở F_{DF} trên Hotel). Điều này cho thấy khi không xử lý mất cân bằng, cả ba mô hình đều thiên mạnh về lớp đa số. Khoảng Recall 0,62 - 0,76 trong số liệu đã công bố (Chương 3, 4) là một điểm vận hành giữa Recall và Specificity; trong các thực nghiệm chạy lại, cơ chế cân bằng lớp (Balanced Bootstrap kết hợp hiệu chỉnh ngưỡng, mục J.1-J.6) đưa Recall và Specificity về mức cân bằng hơn: nếu bỏ hoàn toàn cơ chế cân bằng (biến thể Baseline), Recall tăng lên 0,77-0,98 nhưng Specificity sụt giảm nghiêm trọng.

Trên cả hai tập dữ liệu, BalancedBoot đạt hoặc gần đạt Balanced Accuracy cao nhất ở đa số tổ hợp mô hình: cả ba mô hình trên Hotel, và F_{CF} trên Restaurant. Với F_{DF} và F_{MF} trên Restaurant, ThresholdOnly và ClassWeight nhỉnh hơn BalancedBoot nhưng chênh lệch nhỏ (0,0010 điểm Balanced Accuracy với F_{DF} và 0,0084 điểm với F_{MF}) - xấp xỉ biên độ biến động giữa các lần chạy randomForest. Nhìn chung, không có biến thể nào trong bốn kỹ thuật xử lý mất cân bằng (ThresholdOnly, ClassWeight, BalancedBoot, SMOTE) vượt trội rõ rệt và nhất quán so với ba biến thể còn lại ở cả ba mô hình, củng cố thêm cho lựa chọn Balanced Bootstrap kết hợp hiệu chỉnh ngưỡng được dùng trong các lần chạy lại Kịch bản 1 tại Phụ lục J (mục J.1–J.6).

So sánh giữa ba mô hình ở cùng điều kiện dữ liệu cho thấy F_{MF} có các độ đo tổng hợp (Balanced Accuracy, F1-macro, MCC) thấp hơn F_{CF} và F_{DF} ở mọi biến thể có xử lý mất cân bằng (riêng biến thể Baseline trên Hotel, F_{MF} cao hơn F_{DF} do F_{DF}-Baseline thiên lệch mạnh về lớp đa số; riêng PR-AUC của F_{MF} thấp hơn F_{DF} nhưng cao hơn F_{CF}), trên cả hai tập dữ liệu. Đây là hệ quả trực tiếp và đã được dự liệu trước của việc phải lấy mẫu con xuống N = 5.000 cho bước phân cụm không gian đa tạp và huấn luyện K rừng cơ sở của F_{MF} (do giới hạn bộ nhớ $O(N^2)$ của ma trận W), khiến F_{MF} trong thực nghiệm này học trên lượng dữ liệu gán nhãn ít hơn đáng kể so với F_{CF}/F_{DF} (vốn dùng đủ N = 112.275). Do đó, chênh lệch này không nên diễn giải là

F_{MF} kém ưu việt hơn về mặt kiến trúc, mà phản ánh đúng giới hạn khả năng mở rộng quy mô đã nêu tại mục 4.5.1 của luận án; khi có hạ tầng tính toán đủ lớn để xây W ở quy mô đầy đủ (hoặc dùng biểu diễn W thưa/cục bộ theo cụm như luận án đề xuất), khoảng cách hiệu năng này được kỳ vọng thu hẹp lại.

Việc hiệu chỉnh ngưỡng trên Validation (ThresholdOnly trở lên) mang lại cải thiện Balanced Accuracy so với Baseline ở cả ba mô hình (rõ rệt nhất ở F_{DF} trên Hotel; F1-macro không tăng nhất quán), xác nhận rằng phần lớn lợi ích quan sát được đến từ việc xử lý mất cân bằng lớp một cách có hệ thống, phù hợp với đóng góp mà luận án đã công bố, chứ không phải ngẫu nhiên theo từng mô hình riêng lẻ.

Kết quả của F_{DF} với biến thể BalancedBoot (tập Hotel: Accuracy 0,6527/BA 0,6355; tập Restaurant: Accuracy 0,6145/BA 0,6083) gần khớp với số liệu Kịch bản 1 cải tiến tại Phụ lục J (tập Hotel: Accuracy 0,6558/BA 0,6389; tập Restaurant: Accuracy 0,6170/BA 0,6084); chênh lệch dưới 0,5 điểm phần trăm là biến động ngẫu nhiên giữa hai lần chạy độc lập của randomForest (cùng phương pháp, cùng dữ liệu), xác nhận tính lặp lại của kết quả. Trên Restaurant, PR-AUC của năm biến thể với F_{DF} gần như không đổi (0,7488–0,7543) trong khi Accuracy, Recall, Specificity thay đổi rõ, cho thấy lợi ích của các kỹ thuật xử lý mất cân bằng chủ yếu đến từ dịch chuyển điểm vận hành (ngưỡng); trên Hotel, PR-AUC tách biệt hơn (0,7709–0,7799) và BalancedBoot cao nhất, cho thấy cơ chế lấy mẫu cân bằng cải thiện cả năng lực phân biệt lớp.

G.5. Kết luận

Thực nghiệm bổ sung này làm rõ đồng thời hai vấn đề nêu ở đầu phụ lục: (i) ba mô hình F_{CF}, F_{DF}, F_{MF} đã được đánh giá lại trên cùng một tập dữ liệu, cùng tỉ lệ chia 60/20/20 và cùng quy trình chia dữ liệu (riêng F_{MF} có thêm bước lấy mẫu con bắt buộc do giới hạn bộ nhớ, đã công khai và giải thích rõ trong mục G.1); (ii) khi không xử lý mất cân bằng (Baseline), cả ba mô hình đều thiên về lớp đa số với Specificity rất thấp, còn bốn kỹ thuật thay thế thông dụng (ThresholdOnly, ClassWeight, BalancedBoot, SMOTE) đều cải thiện rõ rệt Specificity và nhìn chung cải thiện Balanced Accuracy so với không xử lý (Baseline) ở cả ba mô hình (F1-macro và MCC không cải thiện nhất quán), mà không có kỹ thuật nào vượt trội tuyệt đối so với Balanced Bootstrap được dùng trong các lần chạy lại tại Phụ lục J.

BẢNG SỐ LIỆU CỦA PHỤ LỤC G

Bảng G.1: So sánh 3 mô hình × 5 biến thể xử lý mất cân bằng lớp — Hotel

Mô hình	Biến thể	Accuracy	Precision	Recall	Specificity	F1	F1-macro	BalancedAccuracy	MCC	PR-AUC
F_{CF}	Baseline	0,6724	0,7315	0,7719	0,4950	0,7512	0,6359	0,6334	0,2736	0,5374
F_{CF}	ThresholdOnly	0,6411	0,7575	0,6469	0,6308	0,6979	0,6280	0,6389	0,2677	0,5374
F_{CF}	ClassWeight	0,6444	0,7535	0,6612	0,6143	0,7043	0,6291	0,6378	0,2665	0,5366
F_{CF}	BalancedBoot	0,6433	0,7562	0,6542	0,6240	0,7015	0,6293	0,6391	0,2685	0,5496
F_{CF}	SMOTE	0,6451	0,7416	0,6844	0,5749	0,7119	0,6249	0,6297	0,2531	0,5363
F_{DF}	Baseline	0,6570	0,6546	0,9837	0,0746	0,7861	0,4607	0,5292	0,1479	0,7761
F_{DF}	ThresholdOnly	0,6427	0,7501	0,6631	0,6061	0,7040	0,6267	0,6346	0,2607	0,7761
F_{DF}	ClassWeight	0,6172	0,7572	0,5925	0,6613	0,6648	0,6093	0,6269	0,2435	0,7709
F_{DF}	BalancedBoot	0,6527	0,7448	0,6966	0,5744	0,7199	0,6315	0,6355	0,2654	0,7799
F_{DF}	SMOTE	0,6291	0,7499	0,6318	0,6244	0,6858	0,6166	0,6281	0,2466	0,7730
F_{MF}	Baseline	0,6547	0,6940	0,8245	0,3520	0,7537	0,5882	0,5882	0,1985	0,5786
F_{MF}	ThresholdOnly	0,6200	0,7161	0,6741	0,5236	0,6945	0,5960	0,5989	0,1939	0,5786
F_{MF}	ClassWeight	0,6171	0,7154	0,6680	0,5262	0,6909	0,5939	0,5971	0,1901	0,5750
F_{MF}	BalancedBoot	0,6204	0,7174	0,6723	0,5278	0,6941	0,5969	0,6000	0,1960	0,6021
F_{MF}	SMOTE	0,6100	0,7078	0,6662	0,5097	0,6864	0,5854	0,5880	0,1725	0,5963

***Ghi chú**: Các hàng tô đậm (BalancedBoot) là biến thể đạt Balanced Accuracy cao nhất trong 5 biến thể, nhất quán ở cả ba mô hình trên tập Hotel.*

Bảng G.2: So sánh 3 mô hình × 5 biến thể xử lý mất cân bằng lớp — Restaurant

Mô hình	Biến thể	Accuracy	Precision	Recall	Specificity	F1	F1-macro	BalancedAccuracy	MCC	PR-AUC
F_{CF}	Baseline	0,6544	0,7150	0,7815	0,4163	0,7468	0,6014	0,5989	0,2082	0,5802
F_{CF}	ThresholdOnly	0,6025	0,7427	0,5973	0,6124	0,6621	0,5898	0,6048	0,2000	0,5802
F_{CF}	ClassWeight	0,6192	0,7324	0,6555	0,5513	0,6918	0,5969	0,6034	0,1998	0,5744
F_{CF}	BalancedBoot	0,6148	0,7395	0,6317	0,5831	0,6814	0,5972	0,6074	0,2059	0,5939
F_{CF}	SMOTE	0,6110	0,7328	0,6350	0,5662	0,6804	0,5918	0,6006	0,1933	0,5802
F_{DF}	Baseline	0,6604	0,7123	0,8039	0,3916	0,7553	0,6003	0,5978	0,2112	0,7542
F_{DF}	ThresholdOnly	0,6179	0,7404	0,6375	0,5811	0,6851	0,5997	0,6093	0,2099	0,7542
F_{DF}	ClassWeight	0,6089	0,7441	0,6099	0,6071	0,6703	0,5948	0,6085	0,2072	0,7488
F_{DF}	BalancedBoot	0,6145	0,7408	0,6288	0,5878	0,6802	0,5975	0,6083	0,2075	0,7543
F_{DF}	SMOTE	0,6149	0,7400	0,6310	0,5847	0,6812	0,5975	0,6078	0,2068	0,7526
F_{MF}	Baseline	0,6453	0,6881	0,8342	0,2915	0,7541	0,5590	0,5628	0,1471	0,6719
F_{MF}	ThresholdOnly	0,5934	0,7098	0,6366	0,5124	0,6712	0,5692	0,5745	0,1440	0,6719
F_{MF}	ClassWeight	0,6072	0,7060	0,6813	0,4684	0,6934	0,5735	0,5748	0,1476	0,6436
F_{MF}	BalancedBoot	0,5830	0,7044	0,6211	0,5116	0,6602	0,5604	0,5664	0,1279	0,6748
F_{MF}	SMOTE	0,5923	0,7077	0,6383	0,5061	0,6712	0,5674	0,5722	0,1398	0,6759

Ghi chú: Các hàng tô đậm là biến thể đạt Balanced Accuracy cao nhất theo từng mô hình trên tập Restaurant: BalancedBoot (F_{CF}), ThresholdOnly (F_{DF}) và ClassWeight (F_{MF}), khác biệt giữa các biến thể tốt nhất khá nhỏ (dưới 0,003 điểm Balanced Accuracy so với biến thể đứng thứ hai).

Phụ lục H

ABLATION KIẾN TRÚC (2×2) VÀ KIỂM ĐỊNH Ý NGHĨA THỐNG KÊ GIỮA BA MÔ HÌNH F_{CF}, F_{DF}, F_{MF}

Phụ lục này bổ sung hai nội dung mà Phụ lục G, Phụ lục J chưa giải quyết bằng thực nghiệm mới: (i) kiểm định ý nghĩa thống kê (McNemar, Wilcoxon) cho khác biệt hiệu năng giữa các mô hình, thay vì chỉ so sánh điểm số trung bình; (ii) thí nghiệm loại trừ (ablation) tách bạch đóng góp riêng của từng thành phần kiến trúc - không gian phân cụm (Euclidean so với đa tạp) và phương pháp kết hợp K rừng (trung bình so với Stacking) đã bị nhập chung trong F_{DF} và F_{MF} khiến chưa thể quy kết rõ khác biệt hiệu năng đến từ thành phần nào.

H.1. Thiết kế ablation 2×2

Mô hình F_{DF} (Chương 3) kết hợp đồng thời hai lựa chọn kiến trúc: phân cụm trên không gian Euclidean và kết hợp K rừng bằng trung bình xác suất (density averaging). F_{MF} (Chương 4) kết hợp đồng thời hai lựa chọn còn lại: phân cụm trên không gian đa tạp (qua ma trận quan hệ W) và kết hợp bằng Stacking. Do hai thành phần luôn đi cùng nhau trong thiết kế gốc, chênh lệch hiệu năng quan sát được giữa F_{DF} và F_{MF} không thể quy kết riêng cho không gian phân cụm hay cho phương pháp kết hợp. Để tách bạch, thực nghiệm bổ sung hai ô còn thiếu của bảng 2×2:

- *Ablation_Euclid_Stacking*: phân cụm Euclidean (tương tự như F_{DF}) nhưng kết hợp K rừng bằng Stacking (tương tự như F_{MF}).
- *Ablation_Manifold_Avg*: phân cụm trên không gian đa tạp W (tương tự như F_{MF}) nhưng kết hợp K rừng bằng trung bình xác suất (tương tự như F_{DF}).

	Kết hợp: TRUNG BÌNH	**Kết hợp: STACKING**
Phân cụm: EUCLIDEAN	F_{DF} (đã có, Chương 3)	Ablation_Euclid_Stacking (mới)
Phân cụm: ĐA TẠP (W)	Ablation_Manifold_Avg (mới)	F_{MF} (đã có, Chương 4)

Hình H.1: Thiết kế ablation 2×2 theo không gian phân cụm và cách kết hợp K rừng

Mô hình FCF (không phân cụm, Chương 2) đóng vai trò mốc tham chiếu "không có cụm" cho cả bốn ô trên. Với mỗi cặp so sánh có chung một thành phần (ví dụ F_{DF} và *Ablation_Euclid_Stacking* cùng phân cụm Euclidean, chỉ khác cách kết

hợp), chênh lệch hiệu năng quan sát được có thể quy trực tiếp cho đúng một thành phần kiến trúc khác biệt duy nhất.

Toàn bộ 5 biến thể (*F_{CF}, F_{DF}, F_{MF}, Ablation_Euclid_Stacking, Ablation_Manifold_Avg*) dùng chung một kỹ thuật xử lý mất cân bằng lớp cố định là Balanced Bootstrap (kết quả tốt nhất/gần tốt nhất theo Phụ lục G) nhằm cô lập hoàn toàn hiệu ứng kiến trúc, không lẫn với hiệu ứng xử lý mất cân bằng lớp. Toàn bộ quy trình (lấy mẫu N = 112.275, chia Train/Validation/Test theo tỉ lệ 60/20/20, hai bước phân cụm, huấn luyện cả 5 biến thể) được lặp lại 5 lần (5 lần chia dữ liệu) với 5 cặp hạt giống ngẫu nhiên độc lập (lấy mẫu: 2026–2030; chia tập: 101–105), báo cáo trung bình ± độ lệch chuẩn qua 5 lần lặp cho từng độ đo.

H.2. Hai phương pháp kiểm định ý nghĩa thống kê

Kiểm định McNemar được áp dụng trên dự đoán cặp đôi (paired) của cùng một tập Test cố định (lần lặp #1), theo bảng tương ứng 2×2 giữa đúng/sai của hai mô hình trên từng quan sát, phù hợp khi so sánh hai bộ phân loại trên cùng một tập kiểm thử [208]. Gọi b là số quan sát mô hình A dự đoán đúng còn mô hình B dự đoán sai, c là số quan sát A sai còn B đúng; các quan sát cả hai cùng đúng hoặc cùng sai không ảnh hưởng đến kiểm định. Thống kê kiểm định (có hiệu chỉnh liên tục, như hàm mcnemar.test của R) là $\chi^2 = (|b - c| - 1)^2 / (b + c)$, tuân theo phân phối chi bình phương 1 bậc tự do dưới giả thuyết H_0: hai mô hình có cùng tỉ lệ dự đoán đúng (khi đó $b \approx c$); p-value = $P(\chi^2_1 > \chi^2)$. Vì Accuracy(A) - Accuracy(B) = $(b - c)/n$, kiểm định này đánh giá chênh lệch Accuracy của hai mô hình trên cùng tập kiểm tra.

Kiểm định Wilcoxon signed-rank (kiểm định hạng có dấu Wilcoxon, ghép cặp, hai phía) [207] là kiểm định phi tham số, không giả định chênh lệch tuân theo phân phối chuẩn, được áp dụng trên 5 giá trị Balanced Accuracy ứng với 5 lần chia dữ liệu ngẫu nhiên (5 hạt giống khác nhau) của mỗi mô hình. Do các lần chia cùng lấy từ một tập dữ liệu nên không hoàn toàn độc lập và có thể làm kiểm định lạc quan hơn [208]; vì vậy kết quả Wilcoxon được dùng để đánh giá tính nhất quán chiều khác biệt, bổ trợ cho kiểm định McNemar. Với mỗi cặp mô hình A, B, chênh lệch Balanced Accuracy ở từng lần lặp được xếp hạng theo giá trị tuyệt đối; thống kê V (do hàm wilcox.test của phần mềm R trả về) là tổng hạng của các lần lặp mà A cao hơn B. Với n = 5, V nhận giá trị từ 0 đến 15: V = 15 khi A cao hơn B ở cả 5/5 lần lặp, V = 0 khi B cao hơn A ở cả 5/5 lần lặp, còn V gần giá trị kỳ vọng 7,5 khi thắng thua xen kẽ, không có chiều khác biệt nhất quán. Ở hai giá trị cực trị (V = 0 hoặc V = 15), p-value

chính xác hai phía là 0,0625 ($2/2^5$), là giá trị nhỏ nhất có thể đạt được khi n = 5; giá trị 0,05906 tại Bảng H.5–H.6 là p-value xấp xỉ chuẩn (có hiệu chỉnh liên tục) do hàm wilcox.test của phần mềm R trả về khi không dùng phép tính chính xác. Do đó ngưỡng $\alpha = 0,05$ quy ước không thể đạt được kể cả khi một mô hình cao hơn ở cả 5/5 lần lặp; đây là hạn chế do cỡ mẫu nhỏ (N_REPEATS = 5), không phải bằng chứng cho thấy hai mô hình không khác nhau.

H.3. Bảng tổng hợp mean ± SD qua 5 lần lặp

Số liệu chi tiết tại Bảng H.1, Bảng H.2 (trình bày ở cuối Phụ lục H).

Lưu ý về PR-AUC: chỉ số này được tính bằng quy tắc hình thang trên lưới ngưỡng 0,01–0,99. Với F_{CF} (một rừng duy nhất), nhiều quan sát có xác suất dự đoán bằng 1, nên ngay tại ngưỡng 0,99 Recall vẫn còn cao và đoạn đường Precision–Recall ứng với Recall thấp không được tính vào diện tích; đây nhiều khả năng là lý do PR-AUC của F_{CF} (và một phần F_{MF}) thấp hơn tỷ lệ lớp positive dù Balanced Accuracy tương đương F_{DF}. Vì vậy PR-AUC chỉ dùng để so sánh các biến thể của cùng một mô hình; so sánh giữa các mô hình dựa trên Balanced Accuracy, F1-macro và MCC.

H.4. Kiểm định McNemar (dự đoán cặp đôi, tập Test cố định)

Số liệu chi tiết tại Bảng H.3, Bảng H.4 (trình bày ở cuối Phụ lục H).

H.5. Kiểm định Wilcoxon signed-rank (5 lần lặp, Balanced Accuracy)

Số liệu chi tiết tại Bảng H.5, Bảng H.6 (trình bày ở cuối Phụ lục H)

H.6. Thảo luận

Kiểm định McNemar (cỡ mẫu bằng số quan sát trong tập Test, rất lớn) cho thấy khác biệt có ý nghĩa thống kê ($p < 0,05$, phần lớn $p < 10^{-19}$) ở 6/7 cặp so sánh trên Hotel và 7/7 cặp trên Restaurant, xác nhận rằng chênh lệch điểm số trung bình đã báo cáo trong luận án và các phụ lục trước (G, H, I) không phải do dao động ngẫu nhiên của một lần chia dữ liệu. Riêng cặp F_{DF} so sánh *Ablation_Euclid_Stacking* trên Hotel không đạt ý nghĩa ($chi^2 = 0,0969$; $p = 0,7556$): khi giữ nguyên không gian phân cụm Euclidean, số quan sát chỉ F_{DF} dự đoán đúng và số quan sát chỉ bản Stacking dự đoán đúng gần bằng nhau, nên tổng số dự đoán đúng của hai mô hình không khác biệt có ý nghĩa. Điều này không có nghĩa hai mô hình mắc lỗi trên cùng các quan sát: Stacking dịch điểm cân bằng sang lớp positive (Bảng H.1: Recall tăng từ 0,6864 lên 0,7806, Specificity giảm từ 0,5857 xuống 0,4481), tức đổi thêm lỗi ở lớp negative lấy thêm dự đoán đúng ở lớp positive với số lượng gần tương đương. Kết luận "chưa đủ

bằng chứng" cũng không khẳng định hai mô hình tương đương: trên Restaurant, chính cặp này khác biệt có ý nghĩa Bảng H.4).

Ngược lại, mọi cặp so sánh có khác biệt về không gian phân cụm (Euclidean so với đa tạp, ở cùng một phương pháp kết hợp) đều đạt ý nghĩa thống kê rất mạnh qua McNemar (F_{DF} so sánh *Ablation_Manifold_Avg và Ablation_Euclid_Stacking* so sánh với F_{MF} đều có $p < 10^{-14}$ ở cả hai tập; cặp F_{DF} so sánh F_{MF}, khác đồng thời cả hai thành phần, cũng có $p < 10^{-23}$), cho thấy lựa chọn không gian phân cụm là thành phần kiến trúc có ảnh hưởng mạnh và nhất quán hơn đến tập dự đoán, so với lựa chọn phương pháp kết hợp K rừng.

Kiểm định Wilcoxon trên 5 lần chia dữ liệu cho kết quả cùng chiều với kiểm định McNemar: ở 6 cặp so sánh còn lại (trừ cặp $F_{CF} – F_{DF}$), một mô hình đạt Balanced Accuracy cao hơn ở cả 5/5 lần lặp tại cả Hotel và Restaurant, tương ứng với giá trị cực trị của thống kê V (V = 0 hoặc V = 15; xem H.2). Tuy p-value (0,05906) chưa đạt ngưỡng 0,05, đây là p-value nhỏ nhất có thể đạt được khi n = 5, nên kết quả này được xem là bằng chứng về tính nhất quán chiều khác biệt qua các lần chia dữ liệu độc lập, chưa phải bằng chứng có ý nghĩa thống kê. Riêng cặp $F_{CF} – F_{DF}$ không nhất quán chiều khác biệt giữa các lần lặp (V = 7 ở Hotel và V = 5 ở Restaurant, gần giá trị kỳ vọng 7,5), phù hợp với việc đây là cặp có chênh lệch điểm số nhỏ nhất trong toàn bộ các cặp so sánh (xem Bảng H.1–H.2).

H.7. Kết luận

Thực nghiệm ablation 2×2 kết hợp hai phép kiểm định thống kê độc lập (McNemar trên dự đoán cặp đôi, Wilcoxon trên nhiều lần lặp) cho phép kết luận: (i) theo kiểm định McNemar, phần lớn khác biệt hiệu năng giữa ba mô hình đề xuất của luận án có ý nghĩa thống kê, không phải dao động ngẫu nhiên, và kiểm định Wilcoxon cho thấy chiều khác biệt nhất quán qua 5 lần lặp (trừ cặp F_{CF} - F_{DF}); (ii) trong hai thành phần kiến trúc cấu thành F_{MF} (không gian phân cụm đa tạp và Stacking), thành phần không gian phân cụm đóng vai trò quyết định hơn đến sự khác biệt trong tập dự đoán, còn Stacking chủ yếu điều chỉnh lại điểm cân bằng Recall/Specificity. Đây là bằng chứng thực nghiệm trực tiếp cho việc kiểm định ý nghĩa thống kê và tách bạch đóng góp của các thành phần DBSCAN, W và Stacking.

BẢNG SỐ LIỆU CỦA PHỤ LỤC H

Bảng H.1: Ablation 2×2 — tập Hotel (mean ± SD, 5 lần lặp)

Mô hình	Accuracy	Precision	Recall	Specificity	F1	F1-macro	BalancedAcc.	MCC	PR-AUC
F_{CF} (không phân cụm)	0,6465 ± 0,0085	0,7454 ± 0,0069	0,6758 ± 0,0245	0,5953 ± 0,0223	0,7087 ± 0,0119	0,6294 ± 0,0058	0,6355 ± 0,0043	0,2642 ± 0,0091	0,5503 ± 0,0034
F_{DF} (Euclid + TB)	0,6499 ± 0,0082	0,7439 ± 0,0042	0,6864 ± 0,0208	0,5857 ± 0,0200	0,7139 ± 0,0113	0,6312 ± 0,0062	0,6360 ± 0,0056	0,2660 ± 0,0115	0,7776 ± 0,0047
F_{MF} (Đa tạp + Stacking)	0,6147 ± 0,0062	0,7127 ± 0,0068	0,6615 ± 0,0066	0,5326 ± 0,0085	0,6862 ± 0,0062	0,5936 ± 0,0058	0,5971 ± 0,0061	0,1899 ± 0,0118	0,6302 ± 0,0410
Ablation: Euclid + Stacking	0,6598 ± 0,0053	0,7128 ± 0,0073	0,7806 ± 0,0277	0,4481 ± 0,0353	0,7448 ± 0,0090	0,6166 ± 0,0050	0,6144 ± 0,0056	0,2400 ± 0,0096	0,7547 ± 0,0030
Ablation: Đa tạp + TB	0,6142 ± 0,0120	0,7399 ± 0,0081	0,6081 ± 0,0375	0,6247 ± 0,0353	0,6669 ± 0,0211	0,6036 ± 0,0080	0,6164 ± 0,0046	0,2247 ± 0,0089	0,7515 ± 0,0075

Ghi chú: Hàng tô đậm: F_{DF} (mô hình chính thức Chương 3) và Ablation_Manifold_Avg (ô ablation đối trọng cùng không gian đa tạp nhưng kết hợp bằng trung bình).

Bảng H.2: Ablation 2×2 — tập Restaurant (mean ± SD, 5 lần lặp)

Mô hình	Accuracy	Precision	Recall	Specificity	F1	F1-macro	BalancedAcc.	MCC	PR-AUC
F_{CF} (không phân cụm)	0,6202 ± 0,0036	0,7334 ± 0,0057	0,6570 ± 0,0158	0,5510 ± 0,0215	0,6930 ± 0,0070	0,5974 ± 0,0024	0,6040 ± 0,0037	0,2010 ± 0,0062	0,5935 ± 0,0042

Mô hình	Accuracy	Precision	Recall	Specificity	F1	F1-macro	BalancedAcc.	MCC	PR-AUC
F_{DF} (Euclid + TB)	0,6165 ± 0,0120	0,7370 ± 0,0016	0,6411 ± 0,0318	0,5698 ± 0,0267	0,6854 ± 0,0183	0,5966 ± 0,0070	0,6055 ± 0,0040	0,2031 ± 0,0086	0,7573 ± 0,0035
F_{MF} (Đa tạp + Stacking)	0,5937 ± 0,0059	0,7036 ± 0,0024	0,6523 ± 0,0139	0,4836 ± 0,0132	0,6769 ± 0,0078	0,5647 ± 0,0044	0,5679 ± 0,0040	0,1324 ± 0,0082	0,6714 ± 0,0470
Ablation: Euclid + Stacking	0,6364 ± 0,0216	0,7042 ± 0,0121	0,7668 ± 0,0868	0,3913 ± 0,1008	0,7317 ± 0,0345	0,5770 ± 0,0065	0,5790 ± 0,0071	0,1696 ± 0,0031	0,7356 ± 0,0040
Ablation: Đa tạp + TB	0,6017 ± 0,0231	0,7252 ± 0,0088	0,6289 ± 0,0752	0,5497 ± 0,0775	0,6714 ± 0,0393	0,5795 ± 0,0096	0,5893 ± 0,0056	0,1733 ± 0,0100	0,7305 ± 0,0059

Ghi chú: Cấu hình, seed và quy trình lặp giống hệt Bảng H.1 chỉ khác tập dữ liệu (Restaurant, N = 112.275).

Bảng H.3: Kiểm định McNemar — Hotel

Cặp so sánh (McNemar, lần lặp #1)	chi²	p-value	Kết luận (α=0,05)
F_{CF} vs F_{DF}	26,1406	3,174e-07	Có ý nghĩa
F_{CF} vs F_{MF}	48,6990	2,984e-12	Có ý nghĩa
F_{DF} vs F_{MF}	114,8741	8,386e-27	Có ý nghĩa
F_{DF} vs Ablation_Euclid_Stacking	0,0969	0,7556	Chưa đủ bằng chứng
F_{MF} vs Ablation_Manifold_Avg	3,9467	0,04696	Có ý nghĩa (biên)
Ablation_Euclid_Stacking vs F_{MF}	111,8068	3,939e-26	Có ý nghĩa

Cặp so sánh (McNemar, lần lặp #1)	chi²	p-value	Kết luận (α=0,05)
F_{DF} vs Ablation_Manifold_Avg	98,9783	2,553e-23	Có ý nghĩa

Bảng H.4: Kiểm định McNemar — Restaurant

Cặp so sánh (McNemar, lần lặp #1)	chi²	p-value	Kết luận (α=0,05)
F_{CF} vs F_{DF}	10,6152	1,122e-03	Có ý nghĩa
F_{CF} vs F_{MF}	53,7380	2,291e-13	Có ý nghĩa
F_{DF} vs F_{MF}	102,4501	4,424e-24	Có ý nghĩa
F_{DF} vs Ablation_Euclid_Stacking	80,9070	2,366e-19	Có ý nghĩa
F_{MF} vs Ablation_Manifold_Avg	10,2399	1,374e-03	Có ý nghĩa
Ablation_Euclid_Stacking vs F_{MF}	315,5833	1,327e-70	Có ý nghĩa
F_{DF} vs Ablation_Manifold_Avg	63,3225	1,755e-15	Có ý nghĩa

Ghi chú: Hàng tô đậm (tập Hotel, F_{DF} so sánh Ablation_Euclid_Stacking): cặp so sánh duy nhất không đạt ý nghĩa thống kê trong cả hai bảng (ở Restaurant, cặp này có khác biệt có ý nghĩa, p = 2,366e-19), khi cố định không gian phân cụm ở Euclidean, số quan sát chỉ một trong hai mô hình dự đoán đúng ở hai chiều gần cân bằng, nên tổng số dự đoán đúng của F_{DF} và bản Stacking không khác biệt có ý nghĩa ở tập Hotel; Stacking chủ yếu dịch điểm cân bằng Recall/Specificity (Bảng H.1) chứ không làm thay đổi đáng kể độ chính xác tổng thể.

Bảng H.5: Kiểm định Wilcoxon — Hotel

Cặp so sánh (Wilcoxon, 5 lần lặp)	V	p-value	Kết luận (α=0,05)
F_{CF} vs F_{DF}	7	1,0000	Chưa đủ bằng chứng
F_{CF} vs F_{MF}	15	0,05906	Chưa đủ (biên, min-p n=5)
F_{DF} vs F_{MF}	15	0,05906	Chưa đủ (biên, min-p n=5)
F_{DF} vs Ablation_Euclid_Stacking	15	0,05906	Chưa đủ (biên, min-p n=5)
F_{MF} vs Ablation_Manifold_Avg	0	0,05906	Chưa đủ (biên, min-p n=5)
Ablation_Euclid_Stacking vs F_{MF}	15	0,05906	Chưa đủ (biên, min-p n=5)
F_{DF} vs Ablation_Manifold_Avg	15	0,05906	Chưa đủ (biên, min-p n=5)

Bảng H.6: Kiểm định Wilcoxon — Restaurant

Cặp so sánh (Wilcoxon, 5 lần lặp)	V	p-value	Kết luận (α=0,05)
F_{CF} vs F_{DF}	5	0,5896	Chưa đủ bằng chứng
F_{CF} vs F_{MF}	15	0,05906	Chưa đủ (biên, min-p n=5)
F_{DF} vs F_{MF}	15	0,05906	Chưa đủ (biên, min-p n=5)
F_{DF} vs Ablation_Euclid_Stacking	15	0,05906	Chưa đủ (biên, min-p n=5)

Cặp so sánh (Wilcoxon, 5 lần lặp)	V	p-value	Kết luận (α=0,05)
F_{MF} vs Ablation_Manifold_Avg	0	0,05906	Chưa đủ (biên, min-p n=5)
Ablation_Euclid_Stacking vs F_{MF}	15	0,05906	Chưa đủ (biên, min-p n=5)
F_{DF} vs Ablation_Manifold_Avg	15	0,05906	Chưa đủ (biên, min-p n=5)

Phụ lục I

THÍ NGHIỆM LOẠI TRỪ ĐẶC TRƯNG (h/e), ƯỚC LƯỢNG CHIỀU NỘI TẠI CỦA KHÔNG GIAN ĐA TẠP VÀ SO SÁNH CÁCH MÃ HÓA ĐẶC TRƯNG

Phụ lục này làm rõ hai nội dung: (1) mức độ đóng góp riêng của đặc trưng Holder (h - định danh người viết đánh giá) và Entity (e - định danh doanh nghiệp) trong bộ dữ liệu Phụ lục C, ngoài Aspect và Time; và (2) ước lượng trực tiếp chiều nội tại (intrinsic dimension) của không gian đặc trưng dùng để xây ma trận quan hệ đa tạp W ở Chương 4, thay vì chỉ suy luận gián tiếp qua tham số DBSCAN như trước; ngoài ra, mục I.3 đánh giá cách mã hóa các đặc trưng định danh (so sánh mã hóa số thứ tự, target encoding và one-hot).

I.1. Ablation đặc trưng h/e

Số liệu chi tiết tại Bảng I.1, Bảng I.2 (trình bày ở cuối Phụ lục I).

So với biến thể đầy đủ, loại bỏ Entity (e) làm giảm Accuracy nhiều hơn loại bỏ Holder (h) trên Hotel (− 0,0652 so với − 0,0288 điểm), trong khi trên Restaurant, loại Holder làm giảm Accuracy nhiều hơn (− 0,0711 so với − 0,0281 điểm) nhưng loại Entity lại làm giảm Balanced Accuracy nhiều hơn (− 0,0369 so với − 0,0283 điểm). Nói cách khác, mức độ đóng góp tương đối của h và e là phụ thuộc bộ dữ liệu chứ không có một thành phần nào luôn chiếm ưu thế tuyệt đối ở cả hai tập; cả hai đều mang tín hiệu dự đoán bổ sung đáng kể ngoài Aspect và Time. Khi loại bỏ đồng thời cả h và e, Balanced Accuracy sụt về xấp xỉ mức ngẫu nhiên/đa số (0,5119 ở tập Hotel; 0,5006 ở tập Restaurant, tức gần như 0,5 tuyệt đối) và MCC gần bằng 0 (0,0260 và 0,0010), khẳng định Aspect và Time đơn thuần gần như không đủ để phân biệt hai lớp quan điểm, cả Holder lẫn Entity đều là thành phần đặc trưng cần thiết, không chỉ mang tính "nhiễu định danh" như có thể đặt vấn đề.

I.2. Ước lượng chiều nội tại của không gian đa tạp

Không gian đặc trưng 4 chiều (Entity_te, Aspect_te, Holder_te, Time_id đã chuẩn hóa), đúng không gian dùng để xây ma trận quan hệ đa tạp W ở Chương 4 được ước lượng chiều nội tại bằng hai phương pháp độc lập: MLE (Maximum Likelihood, Levina–Bickel [77]) và DANCo [209], trên 5 lần lấy mẫu con ngẫu nhiên (5.000 quan

sát/lần, k = 10 láng giềng gần nhất). Một xấp xỉ cục bộ của CA-PCA (Local-PCA: với mỗi điểm, PCA trên k láng giềng gần nhất, đếm số thành phần cần giữ để đạt ≥ 90% phương sai, sau đó lấy trung vị/trung bình) được bổ sung tham khảo do CA-PCA nguyên bản không có sẵn trong các gói R hiện hành.

Số liệu chi tiết tại Bảng I.3, Bảng I.4 (trình bày ở cuối Phụ lục I).

Phương pháp MLE cho d′ nhỏ hơn số chiều quan sát D = 4 một cách nhất quán qua toàn bộ 5 lần lấy mẫu độc lập ở cả hai tập dữ liệu; DANCo cho d′ = 3 (tập Hotel) và d′ = 4 = D (tập Restaurant): trung bình d′ ≈ 2,78 (tập Hotel) và d′ ≈ 3,66 (tập Restaurant) khi gộp MLE và DANCo. Kết quả này là bằng chứng thực nghiệm trực tiếp (không chỉ suy luận gián tiếp qua tham số DBSCAN) ủng hộ giả thuyết đa tạp trên tập Hotel, nơi cả hai phương pháp cho d′ < D; trên tập Restaurant, phương pháp MLE cho d′ < D nhưng DANCo cho d′ = D = 4, nên bằng chứng ở tập này chưa nhất quán giữa hai phương pháp. Do không gian quan sát chỉ có D = 4 chiều, điều kiện d′ < 5 nêu ở Chương 4 luôn được thỏa; điều có ý nghĩa là d′ nhỏ hơn D. Ước lượng Local-PCA (2,62 và 3,01) thấp hơn DANCo (và thấp hơn MLE trên tập Restaurant; trên tập Hotel xấp xỉ MLE là 2,56), phù hợp với đặc điểm đã biết của các phương pháp local-PCA/CA-PCA là có xu hướng đánh giá thấp chiều nội tại khi mật độ điểm không đều hoặc khi có nhiều điểm gần trùng lặp (do một số đặc trưng được mã hóa mục tiêu - target encoding, chỉ nhận rất ít giá trị phân biệt, ví dụ Aspect_te), nên được xem là tham khảo bổ sung, còn hai phương pháp MLE và DANCo là hai bằng chứng chính được khuyến nghị trích dẫn.

I.3. So sánh các cách mã hóa đặc trưng định danh

Nội dung này đánh giá cách mã hóa các đặc trưng định danh của bộ năm. Ba cách mã hóa được so sánh trên cùng mô hình rừng phân loại F_{CF} (randomForest 300 cây, Bootstrap cân bằng lớp qua phương pháp sampsize/strata, ngưỡng quyết định chọn trên tập xác thực theo Balanced Accuracy) và lặp với ba hạt giống 2026, 2027, 2028: (1) mã hóa số thứ tự (ordinal/label encoding), là cách dùng trong các bài báo đã công bố và công thức (3.2); (2) mã hóa mục tiêu- target encoding làm trơn Laplace (m = 10), chỉ ước lượng trên tập huấn luyện, là cách dùng tại Phụ lục G–J; (3) mã hóa one-hot, giữ đầy đủ với đặc trưng ít giá trị, còn với đặc trưng nhiều giá trị chỉ giữ 30 giá trị phổ biến nhất trên tập huấn luyện và gộp phần còn lại vào một cột "khác". Giới

hạn này là cần thiết vì trên yelpHotelData, tập huấn luyện có 39.925 thực thể, 3 khía cạnh, 2.585 người đánh giá và 3.969 giá trị thời gian khác nhau, one-hot đầy đủ sẽ tạo hơn 46.000 cột. Dữ liệu gồm tập yelpHotelData (lấy mẫu N = 112.275 như Phụ lục G, chia 60/20/20 thành 67.365/22.455/22.455 bộ năm) và VLSP-2018 Restaurant, Hotel (bộ năm từ dữ liệu gốc, các tập Train/Dev/Test chính thức như Phụ lục F).

Số liệu chi tiết tại Bảng J.15 (trình bày ở cuối Phụ lục I).

Trên yelpHotelData, nơi các đặc trưng định danh có lực lượng rất lớn, target encoding cho kết quả tốt nhất ở cả bốn độ đo (Balanced Accuracy 0,6303, MCC 0,2517) và có thời gian huấn luyện thấp nhất; mã hóa số thứ tự cho kết quả thấp hơn (Balanced Accuracy 0,5765) do gán một thứ tự nhân tạo cho các giá trị định danh; one-hot rút gọn cho kết quả thấp nhất (Balanced Accuracy 0,5184) vì phần lớn thực thể và người đánh giá bị gộp vào cột "khác" nên mất gần hết thông tin, đồng thời thời gian huấn luyện tăng khoảng 23–41 lần do véc-tơ đặc trưng có 96 cột. Trên VLSP-2018, nơi các đặc trưng có ít giá trị (Bảng F.1, F.2), ba cách mã hóa cho kết quả gần như nhau (chênh lệch Balanced Accuracy khoảng 0,03 trên tập Restaurant và dưới 0,001 trên tập Hotel); mã hóa one-hot kém ổn định trên tập Restaurant (độ lệch chuẩn Accuracy 0,1628) do ngưỡng chọn trên tập Dev dao động giữa các lần lặp. Kết quả cho thấy: (i) với dữ liệu có đặc trưng định danh lực lượng lớn như tập Yelp thì mã hõa target encoding là lựa chọn phù hợp và đã được áp dụng cho các thực nghiệm bổ sung tại Phụ lục G–J; (ii) số liệu của các bảng chính tại Chương 3, Chương 4 dùng mã hóa số thứ tự theo các bài báo đã công bố được giữ nguyên và cách mã hóa này là một hạn chế đã được nêu tại §5.1.

I.4. Thảo luận chung

Các thực nghiệm của Phụ lục I bổ khuyết trực tiếp hai câu hỏi còn để ngỏ về vai trò của h, e và về giả thuyết đa tạp (cùng với Phụ lục G, Phụ lục H, Phụ lục J). Kết quả không làm thay đổi kiến trúc hay số liệu chính thức đã báo cáo trong luận án, mà cung cấp thêm bằng chứng thực nghiệm định lượng cho hai giả định đã được luận án nêu nhưng chưa kiểm chứng trực tiếp bằng thực nghiệm mới: vai trò của định danh Holder/Entity trong Phụ lục C và giả thuyết đa tạp làm nền tảng cho Chương 4.

I.5. Kết luận

Kết quả phụ lục cho thấy: (i) cả hai đặc trưng định danh Holder và Entity trong Phụ lục C đều đóng góp tín hiệu dự đoán thực chất, mức độ đóng góp tương đối phụ thuộc bộ dữ liệu và (ii) chiều nội tại của không gian đặc trưng dùng xây dựng đa tạp ở Chương 4 nhỏ hơn số chiều quan sát, được xác nhận bằng phương pháp MLE ở cả hai tập và bằng DANCo trên tập Hotel, củng cố tính hợp lý của giả thuyết đa tạp mà mô hình F_{MF} dựa vào; (iii) với đặc trưng định danh có lực lượng lớn, mã hõa target encoding cho kết quả tốt hơn mã hóa số thứ tự và one-hot, còn với đặc trưng ít giá trị (VLSP-2018) ba cách mã hóa cho kết quả tương đương.

BẢNG SỐ LIỆU CỦA PHỤ LỤC I

Bảng I.1: Ablation đặc trưng — Hotel (mean ± SD, 3 lần lặp, một rừng ngẫu nhiên duy nhất, BalancedBoot)

Biến thể đặc trưng	Số ĐT	Accuracy	Precision	Recall	Specificity	F1	F1-macro	BalancedAcc.	MCC
Đầy đủ (e, a, h, t)	4	0,6449 ± 0,0067	0,7493 ± 0,0101	0,6643 ± 0,0241	0,6113 ± 0,0265	0,7040 ± 0,0106	0,6300 ± 0,0041	0,6378 ± 0,0038	0,2676 ± 0,0069
Loại bỏ Holder (h)	3	0,6161 ± 0,0159	0,7128 ± 0,0065	0,6644 ± 0,0577	0,5308 ± 0,0592	0,6867 ± 0,0294	0,5937 ± 0,0054	0,5976 ± 0,0031	0,1922 ± 0,0068
Loại bỏ Entity (e)	3	0,5797 ± 0,0165	0,7163 ± 0,0073	0,5623 ± 0,0566	0,6093 ± 0,0577	0,6288 ± 0,0328	0,5708 ± 0,0099	0,5858 ± 0,0067	0,1658 ± 0,0131
Chỉ Aspect + Time	2	0,4524 ± 0,0235	0,6570 ± 0,0080	0,2935 ± 0,0853	0,7303 ± 0,0851	0,4002 ± 0,0841	0,4459 ± 0,0333	0,5119 ± 0,0021	0,0260 ± 0,0055

Bảng I.2: Ablation đặc trưng — Restaurant (mean ± SD, 3 lần lặp)

Biến thể đặc trưng	Số ĐT	Accuracy	Precision	Recall	Specificity	F1	F1-macro	BalancedAcc.	MCC
Đầy đủ (e, a, h, t)	4	0,6202 ± 0,0036	0,7354 ± 0,0048	0,6533 ± 0,0130	0,5580 ± 0,0134	0,6919 ± 0,0052	0,5984 ± 0,0021	0,6057 ± 0,0007	0,2039 ± 0,0016
Loại bỏ Holder (h)	3	0,5491 ± 0,0143	0,7343 ± 0,0063	0,4852 ± 0,0365	0,6695 ± 0,0313	0,5837 ± 0,0252	0,5456 ± 0,0123	0,5774 ± 0,0049	0,1490 ± 0,0087
Loại bỏ Entity (e)	3	0,5921 ± 0,0050	0,7052 ± 0,0064	0,6452 ± 0,0223	0,4925 ± 0,0264	0,6736 ± 0,0092	0,5647 ± 0,0016	0,5688 ± 0,0024	0,1337 ± 0,0035
Chỉ Aspect + Time	2	0,4509 ± 0,0260	0,6529 ± 0,0039	0,3387 ± 0,0725	0,6626 ± 0,0646	0,4431 ± 0,0645	0,4492 ± 0,0271	0,5006 ± 0,0010	0,0010 ± 0,0096

Ghi chú: e = Entity_te, a = Aspect_te, h = Holder_te, t = Time_id (đã mã hóa/chuẩn hóa như Chương 2). Biến thể "Chỉ Aspect + Time" loại bỏ đồng thời cả h và e, dùng làm mốc dưới để đánh giá mức độ mô hình phụ thuộc vào định danh thực thể/người viết.

Bảng I.3: Ước lượng chiều nội tại d′ — Hotel (D = 4 chiều quan sát)

Lần lặp subsample	MLE	DANCo	Local-PCA (trung vị)	Local-PCA (trung bình)
1	2,5771	3	3	2,6230
2	2,5547	3	3	2,6210
3	2,5766	3	3	2,6246
4	2,5411	3	3	2,5980
5	2,5697	3	3	2,6172
Trung bình	2,5638	3,0000	3	2,6168

Bảng I.4: Ước lượng chiều nội tại d′ — Restaurant (D = 4 chiều quan sát)

Lần lặp subsample	MLE	DANCo	Local-PCA (trung vị)	Local-PCA (trung bình)
1	3,3411	4	3	3,0182
2	3,2904	4	3	2,9934
3	3,3295	4	3	3,0102
4	3,3166	4	3	3,0028
5	3,3229	4	3	3,0056
Trung bình	3,3201	4,0000	3	3,0060

Bảng I.5: So sánh ba cách mã hóa đặc trưng định danh với F_{CF} trên tập kiểm tra (trung bình ± độ lệch chuẩn qua 3 hạt giống)

Tập dữ liệu	Cách mã hóa	Số cột	Accuracy	F1-macro	Balanced Accuracy	MCC	Thời gian (s)
yelpHotelData	Số thứ tự	4	0,5500 ± 0,0159	0,5479 ± 0,0139	0,5765 ± 0,0018	0,1501 ± 0,0008	19,5
yelpHotelData	Target encoding	4	0,6276 ± 0,0048	0,6181 ± 0,0030	0,6303 ± 0,0004	0,2517 ± 0,0012	11,0
yelpHotelData	One-hot (rút gọn)	96	0,4128 ± 0,0005	0,3701 ± 0,0009	0,5184 ± 0,0002	0,0572 ± 0,0006	452,7
VLSP-2018 Restaurant	Số thứ tự	4	0,6331 ± 0,0033	0,4994 ± 0,0002	0,6492 ± 0,0069	0,1720 ± 0,0074	0,3
VLSP-2018 Restaurant	Target encoding	4	0,6435 ± 0,0026	0,5018 ± 0,0014	0,6388 ± 0,0013	0,1612 ± 0,0015	0,2
VLSP-2018 Restaurant	One-hot	28	0,5035 ± 0,1628	0,4187 ± 0,0905	0,6181 ± 0,0127	0,1409 ± 0,0170	0,6
VLSP-2018 Hotel	Số thứ tự	4	0,6897 ± 0,0072	0,6624 ± 0,0054	0,7158 ± 0,0020	0,3819 ± 0,0044	0,8
VLSP-2018 Hotel	Target encoding	4	0,6879 ± 0,0008	0,6613 ± 0,0006	0,7158 ± 0,0003	0,3815 ± 0,0005	0,8
VLSP-2018 Hotel	One-hot	48	0,6897 ± 0,0006	0,6627 ± 0,0005	0,7166 ± 0,0004	0,3832 ± 0,0008	4,0

Ghi chú: Thời gian là thời gian huấn luyện trung bình trên CPU (giây); "Số cột" là số chiều của véc-tơ đặc trưng sau mã hóa bốn thành phần e, a, h, t.

Phụ lục J
KẾT QUẢ THỰC NGHIỆM BỔ SUNG

Phụ lục này tập hợp các thực nghiệm bổ sung nhằm đối chiếu với số liệu đã công bố, gồm ba phần: (1) mục J.1–J.6 thực nghiệm Kịch bản 1 của mô hình rừng mật độ với cơ chế cân bằng lớp và tối ưu ngưỡng quyết định; (2) mục J.7–J.10 thực nghiệm ba mô hình rừng trên cỡ mẫu đầy đủ và so sánh thống nhất 10 mô hình theo cùng một quy trình; (3) mục J.11–J.12 phân tích ảnh hưởng của cách chia dữ liệu đến chỉ số đánh giá và lặp lại phép so sánh thống nhất với 5 hạt giống × 10.000 đánh giá trên yelpHotelData và yelpResData, kèm bản cải tiến của mô hình rừng đề xuất.

Mục J.1–J.6 trình bày một thực nghiệm bổ sung cho Kịch bản 1 (mục 3.7.3.1 của luận án) trên hai tập dữ liệu Hotel và Restaurant, nhằm kiểm tra độ vững của phương pháp trước hiện tượng mất cân bằng lớp và cải tiến các mô hình đề xuất. Toàn bộ nội dung, phương pháp luận và số liệu Kịch bản 1 đã trình bày ở Chương 3, phụ lục bổ sung với hai cải tiến kỹ thuật nhằm kiểm soát mất cân bằng lớp.

J.1. Mục đích và bối cảnh

Kịch bản 1 sử dụng thuật toán DBSCAN để phân cụm tập Train không nhãn thành K cụm, huấn luyện K rừng ngẫu nhiên giám sát trên K cụm này, sau đó kết hợp kết quả dự đoán của K rừng theo phương pháp trung bình mật độ xác suất (density averaging) như mô tả tại mục 3.7.3.1 của luận án và hai cải tiến bổ sung là:

(1) Bootstrap cân bằng lớp trong từng rừng con, thông qua tham số sampsize/strata của hàm randomForest(), mỗi cây được huấn luyện trên mẫu bootstrap có số lượng bằng nhau giữa hai lớp.

(2) Tối ưu ngưỡng quyết định trên một tập Validation độc lập (không dùng để huấn luyện hay đánh giá cuối), bằng tìm kiếm lưới trong khoảng 0,05–0,95 để tối đa hóa Balanced Accuracy = 0,5 × (Recall + Specificity), thay vì áp ngưỡng cố định 0,5.

J.2. Thiết lập thực nghiệm

Cỡ mẫu: N = 112.275 bộ năm quan điểm cho cả hai tập Hotel và Restaurant, giống với cỡ mẫu đã sử dụng trong luận án, nhằm đảm bảo khả năng đối chiếu. Tách dữ liệu theo tỷ lệ 60% Train / 20% Validation / 20% Test thay vì 80/20 nhằm có một tập Validation độc lập phục vụ việc tối ưu ngưỡng mà không rò rỉ thông tin từ tập

Test. Các đặc trưng Entity/Aspect/Holder được mã hóa bằng target encoding (làm trơn Laplace, m = 10) và tham số mã hóa được ước lượng trên tập Train và áp dụng lại cho Validation/Test.

Cơ chế kết hợp K rừng: trung bình xác suất dự đoán lớp "positive" của K rừng (density averaging), đúng theo phương pháp chính thức của Kịch bản 1 tại mục 3.7.3.1 của luận án, không thay đổi so với công trình đã công bố.

J.3. Kết quả trên tập Test

Số liệu chi tiết tại Bảng J.1, Bảng J.2, Bảng J.3(trình bày ở cuối Phụ lục J).

J.4. Đối chiếu với Bài báo số 4, Bài báo số 5 và Bài báo số 6

Số liệu chi tiết tại Bảng J.4, Bảng J.5, Bảng J.6(trình bày ở cuối Phụ lục J).

J.5. Thảo luận về khác biệt số liệu

Điểm đáng chú ý nhất trong đối chiếu trên là chỉ số F1 của mục J.1–J.6 trên tập Hotel (0,7223) rất gần với F1 của Bài báo số 4 (0,72), trong khi Accuracy chênh lệch rất lớn (0,6558 so với 0,92). Vì F1 phản ánh trực tiếp năng lực phân loại trên lớp "positive" (điều hòa giữa Precision và Recall), sự tương đồng này cho thấy năng lực phân loại thực sự của Kịch bản 1 là tương đương giữa hai lần thực nghiệm; phần chênh lệch lớn về Accuracy chủ yếu bắt nguồn từ cách xử lý mất cân bằng lớp và lựa chọn ngưỡng quyết định, chứ không phải từ một mô hình yếu hơn.

Ba yếu tố kỹ thuật giải thích phần lớn khác biệt còn lại giữa các công bố và mục J.1–J.6 như sau:

Thứ nhất, cơ chế kết hợp K rừng khác nhau (trung bình mật độ xác suất trong luận án và mục J.1–J.6, majority vote trong Bài báo số 4, stacking trong Bài báo số 6) tạo ra phân phối xác suất dự đoán khác nhau, dẫn đến độ nhạy khác nhau khi áp một ngưỡng quyết định cố định.

Thứ hai, Bài báo số 4, Bài báo số 5 và Bài báo số 6 đều không áp dụng cơ chế cân bằng bootstrap và không báo cáo Balanced Accuracy hay Specificity, nên không thể loại trừ khả năng Accuracy cao của các công bố này phần nào phản ánh xu hướng mô hình thiên về lớp đa số. Ngoài ra, Bài báo số 4 báo cáo Recall 0,60 (Hotel) và 0,59 (Restaurant) cùng Accuracy 0,91–0,92; với lớp thiểu số chiếm khoảng 35–37% mẫu, Recall khoảng 0,6 ở bất kỳ lớp nào chỉ cho phép Accuracy tối đa khoảng 0,85, nên hai chỉ số này khó đồng thời đúng trên cùng một tập kiểm tra và cần được đối

chiếu thận trọng (xem mục J.11 về ảnh hưởng của cách chia dữ liệu). Hiện tượng thiên lệch tương tự cũng từng được ghi nhận với chính Kịch bản 1 trước khi áp dụng hai cải tiến của Phụ lục này (Balanced Accuracy ban đầu chỉ đạt 0,4978–0,6081 khi dùng bootstrap mặc định và ngưỡng 0,5), cũng như với mô hình cơ sở SVM tuyến tính (và F_{DF}) tại ngưỡng 0,5 trên tập VLSP-2018 Restaurant tại Phụ lục F (Bảng F.4, F1-neg = 0,0000 khi tỉ lệ mất cân bằng $\rho \approx 10,5$).

Thứ ba, tỷ lệ tách dữ liệu khác nhau (80/20 ở Bài báo số 4 so với 60/20/20 ở mục J.1–J.6) làm giảm tỷ lệ dữ liệu huấn luyện từ 80% xuống 60% tổng số mẫu (tức khoảng một phần tư lượng dữ liệu huấn luyện) để dành cho tập Validation phục vụ tối ưu ngưỡng; đây có thể là một phần nguyên nhân phụ, nhưng không phải nguyên nhân chính vì F1 giữa hai lần thực nghiệm gần như không đổi.

J.6. Kết luận và khuyến nghị

Nội dung Kịch bản 1 trình bày tại Chương 3 (mục 3.7.3.1) của luận án, cũng như số liệu đã công bố tại Bài báo số 4 và Bài báo số 5, được giữ nguyên như đã công bố; Phụ lục này không thay thế mà chỉ bổ sung một phép đối chứng có kiểm soát mất cân bằng lớp.

Kết quả đối chứng cho thấy: khi kiểm soát mất cân bằng lớp bằng bootstrap cân bằng và tối ưu ngưỡng trên tập Validation độc lập, Kịch bản 1 vẫn giữ được năng lực phân loại tương đương với số liệu đã công bố (F1 ≈ Bài báo số 4), đồng thời đạt được chỉ số Balanced Accuracy trung thực hơn (0,6389 trên Hotel, 0,6084 trên Restaurant), phản ánh sát hơn hiệu năng thực tế của mô hình trên cả hai lớp positive và negative.

Các bảng kết quả chính của luận án (Bảng 3.5–3.9, Bảng 4.2–4.4) trình bày số liệu đã công bố tại Bài báo 5 và Bài báo 6, được giữ nguyên để bảo đảm tính truy xuất nguồn. Trong quá trình hoàn thiện luận án, nghiên cứu sinh đã chạy lại ba mô hình rừng đề xuất và các mô hình tham chiếu theo một quy trình thống nhất (cùng dữ liệu, cùng cách chia, cùng cơ chế chọn ngưỡng và cùng bộ độ đo). Mục J.7–J.10 tổng hợp các kết quả chạy lại đó để người đọc đối chiếu.

J.7. Kết quả chạy lại trên cỡ mẫu đầy đủ (N = 112.275, Phụ lục H và mục J.1–J.6)

Số liệu chi tiết tại Bảng J.7 (trình bày ở cuối Phụ lục J).

J.8. So sánh thống nhất 10 mô hình (yelpHotelData, 3.000 đánh giá, 1 lần chia)

Thực nghiệm dùng 3.000 đánh giá lấy ngẫu nhiên (hạt giống 2026), chia theo đánh giá thành 60/20/20; tập kiểm tra gồm 600 đánh giá (63,7% positive). Ngưỡng quyết định của mọi mô hình được chọn trên tập xác thực theo Balanced Accuracy (BA). Cách 1: mỗi mô hình dùng đầu vào tự nhiên của nó (mô hình rừng và học máy dùng bộ năm; LSTM, BiLSTM-Attention, BERT, RoBERTa dùng văn bản đánh giá gốc). Cách 2: mọi mô hình chỉ dùng bộ năm (e, a, h, t); các mô hình văn bản đọc bộ năm dưới dạng chuỗi. Bảng J.8 được giữ nguyên để đối chiếu; kết quả lặp lại với 5 hạt giống và 10.000 đánh giá được trình bày tại mục J.12 (Bảng J.11, Bảng J.12).

Số liệu chi tiết tại Bảng J.8, Bảng J.9 (trình bày ở cuối Phụ lục J).

J.9. Các yếu tố tạo nên khác biệt so với số liệu công bố

- Cơ chế kết hợp K rừng khác nhau giữa các công bố (biểu quyết đa số ở Bài báo 4, Stacking ở Bài báo 6) và các lần chạy lại (trung bình xác suất cho rừng mật độ, Stacking cho rừng đa tạp) - xem Bảng J.4.

- Xử lý mất cân bằng lớp và ngưỡng quyết định: các lần chạy lại dùng Bootstrap cân bằng lớp và chọn ngưỡng theo Balanced Accuracy trên tập xác thực; các công bố dùng ngưỡng 0,5 và không cân bằng lớp. Khi dữ liệu có khoảng 63–65% mẫu positive, Accuracy chịu ảnh hưởng mạnh của ngưỡng.

- Cách chia dữ liệu: các công bố chia ngẫu nhiên theo dòng bộ năm (80/20 hoặc 60/20/20); thực nghiệm J.8 chia theo đánh giá gốc để các dòng của cùng một đánh giá không xuất hiện đồng thời ở tập huấn luyện và tập kiểm tra.

- Quy tắc nhãn: các lần chạy lại gộp mức 3★ vào negative (theo mã nguồn, Bảng 3.3); mô tả nhãn trong Bài báo 4 và Bài báo 6 khác nhau (xem ghi chú dưới Bảng đối chiếu công trình – chương – nội dung sử dụng).

- Cỡ mẫu và lấy mẫu con: rừng đa tạp cần lấy mẫu con 5.000 dòng để xây ma trận W do giới hạn bộ nhớ (Phụ lục G).

J.10. Nhận xét

Kết quả chạy lại theo quy trình thống nhất cho các giá trị thấp hơn số liệu đã công bố. Khi mọi mô hình chỉ sử dụng thông tin của bộ năm (Cách 2), Balanced Accuracy của cả mô hình rừng, mô hình học máy, mô hình học sâu và mô hình Transformer đều nằm trong khoảng 0,49–0,56, cho thấy lượng thông tin về cực tính

cảm xúc chứa trong bốn thành phần (e, a, h, t) là hạn chế; khi được sử dụng văn bản gốc (Cách 1), BERT và RoBERTa đạt 0,719 và 0,783. Điều này phù hợp với nhận định tại §5.1 rằng hiệu năng của các mô hình rừng phụ thuộc vào chất lượng và nội dung của bước trích xuất bộ năm, và củng cố hướng phát triển tại §5.2: kết hợp biểu diễn ngữ nghĩa của văn bản với cấu trúc rừng để giữ khả năng diễn giải và chi phí tính toán thấp của các mô hình đề xuất. Mục J.11 (Bảng J.10) làm rõ nguyên nhân chênh lệch so với số liệu đã công bố; mục J.12 (Bảng J.11–J.17) củng cố nhận xét trên bằng nhiều hạt giống với cỡ mẫu lớn hơn, theo cả Cách 1 (Bảng J.11) và Cách 2 (Bảng J.14 - Bảng J.17).

J.11. Ảnh hưởng của cách chia dữ liệu đến chỉ số đánh giá

Để làm rõ nguyên nhân chênh lệch giữa số liệu đã công bố (Bảng 3.7–3.9, Bảng 4.4) và kết quả chạy lại tại mục J.7–J.8, nghiên cứu sinh thực hiện một kiểm tra có kiểm soát trên yelpHotelData. Trên tập dữ liệu này, mỗi đánh giá được tách thành ba dòng bộ năm có cùng thực thể e, người đánh giá h, thời gian t và cùng nhãn s, chỉ khác khía cạnh a (ba khía cạnh AcceptsCreditCards, PriceRange, WiFi lặp lại ở mọi đánh giá). Khi chia ngẫu nhiên theo dòng bộ năm như các bài báo, một phần dòng của cùng một đánh giá có thể đồng thời nằm ở tập huấn luyện và tập kiểm tra; khi đó mô hình có thể dự đoán đúng nhờ ghi nhớ bộ (e, h, t) đã thấy thay vì khái quát hóa. Ngược lại, khi chia theo đánh giá gốc, mọi dòng của một đánh giá cùng nằm trong một tập, nên tập kiểm tra chỉ gồm các đánh giá mô hình chưa từng thấy.

Cấu hình kiểm tra: Random Forest 300 cây, mã hóa số thứ tự như các bài báo, ngưỡng 0,5, chia 80/20, cỡ mẫu 112.275 bộ năm như luận án, hạt giống 2026; chỉ thay đổi cách lấy mẫu và cách chia dữ liệu. Kết quả được trình bày tại Bảng J.10.

Số liệu chi tiết tại Bảng J.10 (trình bày ở cuối Phụ lục J).

Với cùng dữ liệu và cùng mô hình, Accuracy thay đổi từ 0,611 đến 0,946 chỉ do cách chia dữ liệu Bảng J.10): chỉ số càng cao khi tỷ lệ dòng kiểm tra có dòng cùng đánh giá trong tập huấn luyện càng lớn (19,1% và 96,0%), trong khi chia theo đánh giá gốc cho kết quả ổn định quanh 0,61 bất kể cách lấy mẫu. Như vậy, các chỉ số đã công bố (Accuracy 0,75–0,82) cần được hiểu là kết quả theo quy trình chia theo dòng của các bài báo; quy trình chia theo đánh giá gốc dùng tại mục J.8 và J.12 cho chỉ số

thấp hơn nhưng phản ánh sát hơn khả năng dự đoán cho đánh giá mới, và là cách chia được khuyến nghị cho các nghiên cứu tiếp theo trên dữ liệu bộ năm (§5.1).

J.12. So sánh thống nhất mở rộng: 5 hạt giống × 10.000 đánh giá

Để có kết luận chắc chắn hơn so với thực nghiệm một lần chia trên 3.000 đánh giá tại mục J.8 (Bảng J.8, giữ nguyên để đối chiếu), phép so sánh thống nhất được lặp lại với cỡ mẫu 10.000 đánh giá yelpHotelData và 5 hạt giống (2026–2030). Mọi điều kiện khác giữ nguyên như mục J.8: chia theo đánh giá gốc 60/20/20, ngưỡng quyết định chọn trên tập xác thực theo Balanced Accuracy, cấu hình mô hình như Bảng J.9. Các mô hình dùng bộ năm được huấn luyện bằng R; LSTM, BiLSTM-Attention bằng gói torch cho R; BERT, RoBERTa được tinh chỉnh trên GPU NVIDIA L4 (Google Colab), mỗi mô hình khoảng 2 phút cho một hạt giống. Kết quả trung bình ± độ lệch chuẩn qua 5 hạt giống được trình bày tại Bảng J.11; chênh lệch ghép cặp theo hạt giống so với Random Forest tại Bảng J.12, Bảng J.11 là kết quả theo Cách 1 (các mô hình văn bản đọc văn bản gốc); kết quả theo Cách 2 (mọi mô hình chỉ dùng bộ năm) được trình bày tại Bảng J.14.

Số liệu chi tiết Bảng J.11, Bảng J.12 (trình bày ở cuối Phụ lục J).

Bảng J.11 cho thấy: (i) khi dùng văn bản gốc, RoBERTa (Balanced Accuracy 0,807) và BERT (0,755) cao hơn rõ rệt các mô hình chỉ dùng bộ năm; (ii) trong nhóm dùng bộ năm, các mô hình đạt 0,534–0,568; F_{DF} (0,561) và F_{CF} (0,557) cao hơn Random Forest (0,554) ở 4/5 hạt giống, nhưng chênh lệch chưa có ý nghĩa thống kê (Bảng J.12); F_{MF} thấp hơn Random Forest có ý nghĩa (−0,020), do bước phân cụm đa tạp và các rừng thành phần chỉ học trên mẫu con 5.000 dòng. Độ lệch chuẩn nhỏ (0,004–0,013) cho thấy kết luận ổn định giữa các lần lặp, khác với Bảng J.8 (3.000 đánh giá, một lần chia). Ở hạt giống 2030, BERT không hội tụ (tổn thất đứng ở khoảng 0,658 qua cả ba epoch, mô hình dự đoán toàn bộ là positive), là hiện tượng tinh chỉnh không ổn định đã được ghi nhận với BERT. LSTM chỉ đạt 0,518 do cài đặt dùng trạng thái ẩn cuối trong khi chuỗi được đệm ở cuối, nên kết quả này không đại diện cho năng lực của kiến trúc LSTM; BiLSTM-Attention không chịu ảnh hưởng này nhờ cơ chế chú ý trên toàn chuỗi.

Song song, nghiên cứu sinh thử nghiệm ba điều chỉnh cho các mô hình rừng đề xuất trên cùng 5 hạt giống × 10.000 đánh giá, chỉ dùng bộ năm (Bảng J.13): (1)

target encoding ngoài fold, đặc trưng tần suất của thực thể và người đánh giá, cùng kích thước nút lá tối thiểu lớn hơn để tránh học thuộc các giá trị mã hóa hiếm (F_{CF}+); (2) định tuyến mỗi mẫu tới các rừng của cụm lân cận theo mật độ k láng giềng, kết hợp với một rừng toàn cục (F_{DF}+); (3) mở rộng nhãn cụm đa tạp từ 5.000 điểm mốc cho toàn bộ tập huấn luyện bằng độ gần đa tạp, để các rừng thành phần học trên đủ dữ liệu (F_{MF}+). Khi mọi mô hình dùng cùng bộ đặc trưng mới (nhóm B), ba bản điều chỉnh cao hơn Random Forest khoảng 0,011–0,013 Balanced Accuracy (F_{CF}+ có ý nghĩa thống kê) và F_{MF}+ thu hẹp khoảng cách của F_{MF} (0,547 so với 0,523). Ở cỡ mẫu bằng luận án (37.425 đánh giá, tương đương 112.275 bộ năm, một hạt giống), các bản điều chỉnh đạt Balanced Accuracy 0,585–0,586, cao hơn Random Forest (0,578) và tương đương Gradient Boosting (0,586), Logistic Regression (0,590). Đây là kết quả thăm dò, được ghi nhận như hướng phát triển của các mô hình đề xuất.

Số liệu chi tiết tại Bảng J.13 (trình bày ở cuối Phụ lục J).

Bảng J.14 lặp lại phép so sánh theo Cách 2: mọi mô hình chỉ dùng bộ năm (e, a, h, t), các mô hình văn bản đọc bộ năm dưới dạng chuỗi, với cùng 5 hạt giống × 10.000 đánh giá và cùng cách chia như Bảng J.11 Khi không còn văn bản gốc, ưu thế của các mô hình văn bản không còn: Balanced Accuracy của BiLSTM-Attention giảm từ 0,701 xuống 0,556, BERT từ 0,755 xuống 0,538, RoBERTa từ 0,807 xuống 0,523; tổn thất trên tập xác thực của BERT, RoBERTa chỉ dao động quanh 0,66 qua ba epoch, và ở hạt giống 2026, 2029 RoBERTa dự đoán toàn bộ là positive. Trên cùng thông tin bộ năm, F_{DF} (0,561) và F_{CF} (0,557) đạt Balanced Accuracy tương đương BiLSTM-Attention (0,556) và cao hơn BERT (0,538), RoBERTa (0,523), LSTM (0,504), trong khi thời gian huấn luyện chỉ vài giây so với khoảng 1,5–6 phút của các mô hình học sâu và Transformer; RoBERTa và LSTM thấp hơn Random Forest có ý nghĩa thống kê. Như vậy, chênh lệch giữa mô hình rừng và Transformer tại Bảng J.11 đến từ thông tin trong văn bản gốc chứ không phải từ năng lực mô hình hóa trên cùng đầu vào: khi cùng dùng bộ năm, các mô hình rừng đề xuất (trừ F_{MF}) đạt mức tương đương hoặc cao hơn các mô hình học sâu và Transformer với chi phí tính toán thấp hơn nhiều.

Số liệu chi tiết tại Bảng J.14 (trình bày ở cuối Phụ lục J).

Để kiểm tra tính nhất quán trên miền dữ liệu thứ hai, phép so sánh theo Cách 1 được lặp lại trên yelpResData với cùng quy trình: 10.000 đánh giá, 5 hạt giống (2026–2030), chia theo đánh giá 60/20/20, ngưỡng quyết định chọn trên tập xác thực theo Balanced Accuracy, cấu hình mô hình như Bảng J.9; kết quả tại Bảng J.15. Xu hướng giống yelpHotelData: các mô hình đọc văn bản gốc đạt Balanced Accuracy cao nhất (RoBERTa 0,823; BERT 0,769; BiLSTM-Attention 0,735), các mô hình dùng bộ năm đạt 0,514–0,553. Trong nhóm dùng bộ năm, F_{DF} (0,543) tương đương Random Forest (0,543), F_{CF} (0,541) thấp hơn không đáng kể (chênh lệch −0,002, khoảng tin cậy chứa 0); SVM và Gradient Boosting cao hơn Random Forest khoảng 0,008–0,010 có ý nghĩa thống kê, còn F_{MF} thấp hơn có ý nghĩa (−0,029). Trên yelpResData, BERT hội tụ ở cả 5 hạt giống và LSTM đạt 0,542, ngang các mô hình dùng bộ năm. Độ lệch chuẩn nhỏ (0,007–0,017) cho thấy kết luận ổn định giữa các lần lặp và giữa hai tập dữ liệu: khoảng cách giữa mô hình đọc văn bản và mô hình dùng bộ năm đến từ lượng thông tin của đầu vào, phù hợp với phân tích theo Cách 2 tại Bảng J.14.

Số liệu chi tiết tại Bảng J.14 (trình bày ở cuối Phụ lục J).

Bảng J.16 lặp lại Cách 2 trên yelpResData với cùng mẫu, cùng cách chia và cùng 5 hạt giống như Bảng J.15. Do mỗi đánh giá nhà hàng sinh khoảng 19 dòng bộ năm (tập huấn luyện khoảng 114.000 dòng), LSTM, BiLSTM-Attention, BERT, RoBERTa được huấn luyện trên 30.000 dòng bộ năm lấy ngẫu nhiên từ tập huấn luyện. Kết quả nhất quán với yelpHotelData (Bảng J.14): khi chỉ có bộ năm, BiLSTM-Attention (0,542) chỉ ngang Random Forest (0,543) và F_{DF} (0,543); BERT (0,510), RoBERTa (0,502) và LSTM (0,502) gần như không học được – tổn thất trên tập xác thực của BERT, RoBERTa dừng quanh 0,65 qua ba epoch và ở 4/5 hạt giống hai mô hình này dự đoán toàn bộ là positive – và đều thấp hơn Random Forest có ý nghĩa thống kê. Như vậy, trên cả hai tập dữ liệu, ưu thế của các mô hình học sâu và Transformer tại Bảng J.11, Bảng J.15 đến từ văn bản gốc; trên cùng đầu vào bộ năm, các mô hình rừng đề xuất F_{CF}, F_{DF} đạt mức tương đương hoặc cao hơn; thời gian huấn luyện của F_{CF} (11,9 giây) thấp hơn nhiều, còn của F_{DF} trên tập này (221,9 giây) cùng bậc với BiLSTM-Attention (167,6 giây) và thấp hơn BERT, RoBERTa.

Số liệu chi tiết tại Bảng J.16 (trình bày ở cuối Phụ lục J).

Thử nghiệm các bản điều chỉnh của mô hình rừng (như Bảng J.13) được lặp lại trên yelpResData với 5 hạt giống × 10.000 đánh giá (Bảng J.17). Trên tập này, các bản điều chỉnh không mang lại cải thiện: F_{CF}+, F_{DF}+, F_{MF}+ đạt 0,537–0,539, tương đương Random Forest cùng bộ đặc trưng (0,537; chênh lệch +0,001 đến +0,002, khoảng tin cậy chứa 0); trọng số kết hợp a chọn trên tập xác thực chỉ từ 0 đến 0,2, tức F_{DF}+, F_{MF}+ gần như trùng với rừng toàn cục F_{CF}+. Với bộ đặc trưng điều chỉnh, Logistic Regression và Gradient Boosting cao hơn Random Forest khoảng 0,02–0,03 có ý nghĩa thống kê. Ở nhóm đặc trưng như Bảng J.8, F_{CF} tương đương Random Forest (chênh lệch −0,004, khoảng tin cậy chứa 0), F_{DF} thấp hơn 0,020 (khoảng tin cậy sát 0) và F_{MF} thấp hơn có ý nghĩa (−0,020). Vì vậy, các điều chỉnh tại Bảng J.13 chỉ được ghi nhận như kết quả thăm dò trên yelpHotelData, chưa khái quát cho yelpResData.

Số liệu chi tiết tại Bảng J.17 (trình bày ở cuối Phụ lục J).

Tổng hợp mục J.11–J.12: theo quy trình chia theo đánh giá gốc, trên cả yelpHotelData và yelpResData (Bảng J.11, Bảng J.15, Bảng J.17,), F_{CF} và F_{DF} đạt Balanced Accuracy tương đương Random Forest (chênh lệch không có ý nghĩa thống kê) và thấp hơn SVM tuyến tính khoảng 0,01; F_{MF} thấp hơn Random Forest khoảng 0,02–0,03 có ý nghĩa thống kê do bước phân cụm đa tạp chỉ học trên mẫu con. Các mô hình rừng đề xuất giữ ưu thế về khả năng diễn giải; giới hạn chung của mọi mô hình dùng bộ năm nằm ở lượng thông tin của bốn thành phần (e, a, h, t), phù hợp với hướng kết hợp biểu diễn ngữ nghĩa văn bản với cấu trúc rừng tại §5.2. Theo Cách 2 (Bảng J.14, Bảng J.16), khi mọi mô hình cùng chỉ dùng bộ năm, F_{DF} và F_{CF} đạt Balanced Accuracy tương đương BiLSTM-Attention và cao hơn về trung bình so với LSTM, BERT, RoBERTa trên cả hai tập dữ liệu; thời gian huấn luyện của F_{CF} thấp hơn nhiều so với các mô hình học sâu, còn của F_{DF} thấp hơn nhiều trên yelpHotelData và cùng bậc với BiLSTM-Attention trên yelpResData. Các bản điều chỉnh tại Bảng J.13 cải thiện trên yelpHotelData nhưng chưa cải thiện trên yelpResData (Bảng J.17).

BẢNG SỐ LIỆU CỦA PHỤ LỤC J

Bảng J.1: Kết quả Kịch bản 1 (có cân bằng bootstrap và tối ưu ngưỡng) trên Hotel và Restaurant

Chỉ số	Hotel	Restaurant
Accuracy	0,6558	0,6170
Precision	0,7475	0,7397
Recall (Sensitivity)	0,6987	0,6366
Specificity	0,5791	0,5802
F1	0,7223	0,6843
Balanced Accuracy	0,6389	0,6084
Ngưỡng tối ưu (trên tập Validation)	(*)	0,53
Cỡ mẫu N	112.275	112.275
Tỷ lệ Train / Validation / Test	60% / 20% / 20%	60% / 20% / 20%

Ghi chú: (*) Với cả hai tập dữ liệu, ngưỡng quyết định được chọn tự động bằng tìm kiếm lưới trong khoảng 0,05–0,95 trên tập Validation để tối đa hóa Balanced Accuracy (mục J.1); giá trị ngưỡng cụ thể của tập Hotel không được lưu lại trong nhật ký chạy (log) gốc, giá trị của tập Restaurant là 0,53.

Bảng J.2: Ma trận nhầm lẫn trên tập Test — Hotel

Thực tế \ Dự đoán	Negative	Positive
Negative	4.673	3.396

Thực tế \ Dự đoán	Negative	Positive
Positive	4.334	10.052

Bảng J.3: Ma trận nhầm lẫn trên tập Test — Restaurant

Thực tế \ Dự đoán	Negative	Positive
Negative	4.534	3.280
Positive	5.321	9.320

Bảng J.4: So sánh thiết lập thực nghiệm giữa các công bố và mục J.1–J.6

Nguồn	Bộ kết hợp K rừng	Cân bằng bootstrap	Tối ưu ngưỡng	Tách dữ liệu	Báo cáo Balanced Accuracy
Bài báo số 4 (2023)	Majority vote	Không	Không (0,5)	80 / 20	Không
Bài báo số 5 (2024)	Không nêu rõ	Không nêu rõ	Không nêu rõ	Không nêu rõ	Không
Bài báo số 6 (2024, đối chứng)	Stacking	Không	Không	60 / 20 / 20	Không
Mục J.1–J.6 (2026)	Trung bình mật độ xác suất (đúng §3.7.3.1)	Có	Có	60 / 20 / 20	Có

Bảng J.5: So sánh chỉ số trên tập Hotel

Nguồn	Accuracy	Precision	Recall	F1	Balanced Accuracy
Bài báo số 4 (test)	0,92	0,91	0,60	0,72	—
Bài báo số 5 (test)	0,75	0,71	0,62	0,67	—
Bài báo số 6 (test, đối chứng)	0,78	0,74	0,62	0,67	—
Mục J.1–J.6 (test)	0,6558	0,7475	0,6987	0,7223	0,6389

Bảng J.6: So sánh chỉ số trên tập Restaurant

Nguồn	Accuracy	Precision	Recall	F1	Balanced Accuracy
Bài báo số 4 (test)	0,91	0,89	0,59	0,71	—
Bài báo số 5 (test)	0,80	0,72	0,67	0,69	—
Bài báo số 6 (test, đối chứng)	0,82	0,75	0,67	0,71	—
Mục J.1–J.6 (test)	0,6170	0,7397	0,6366	0,6843	0,6084

Bảng J.7: Ba mô hình rừng đề xuất khi chạy lại (trung bình ± độ lệch chuẩn qua 5 lần lặp, Phụ lục H)

Tập dữ liệu	Mô hình	Accuracy	Balanced Accuracy	F1-macro	Số liệu công bố (Accuracy)
yelpHotelData	Rừng phân loại (F_{CF})	0,6465 ± 0,0085	0,6355 ± 0,0043	0,6294 ± 0,0058	–
yelpHotelData	Rừng mật độ (F_{DF})	0,6499 ± 0,0082	0,6360 ± 0,0056	0,6312 ± 0,0062	0,75 (Bài báo 5)
yelpHotelData	Rừng đa tạp (F_{MF})	0,6147 ± 0,0062	0,5971 ± 0,0061	0,5936 ± 0,0058	0,78 (Bài báo 6)
yelpResData	Rừng phân loại (F_{CF})	0,6202 ± 0,0036	0,6040 ± 0,0037	0,5974 ± 0,0024	–
yelpResData	Rừng mật độ (F_{DF})	0,6165 ± 0,0120	0,6055 ± 0,0040	0,5966 ± 0,0070	0,80 (Bài báo 5)
yelpResData	Rừng đa tạp (F_{MF})	0,5937 ± 0,0059	0,5679 ± 0,0040	0,5647 ± 0,0044	0,82 (Bài báo 6)

Bảng J.8: Balanced Accuracy trên tập kiểm tra theo hai cách đưa đầu vào

Mô hình	Nhóm	BA – Cách 1	BA – Cách 2 (KTC 95%)	ROC-AUC – Cách 2	Thời gian huấn luyện (s)
Rừng phân loại (F_{CF})	Đề xuất	0,544	0,544 (0,515–0,572)	0,545	0,4
Rừng mật độ (F_{DF})	Đề xuất	0,511	0,511 (0,472–0,548)	0,543	0,8
Rừng đa tạp (F_{MF})	Đề xuất	0,526	0,526 (0,485–0,569)	0,536	811,3
Random Forest	Học máy	0,547	0,547 (0,519–0,575)	0,544	0,5
Gradient Boosting	Học máy	0,561	0,561 (0,526–0,595)	0,554	0,6

Mô hình	Nhóm	BA – Cách 1	BA – Cách 2 (KTC 95%)	ROC-AUC – Cách 2	Thời gian huấn luyện (s)
SVM (tuyến tính)	Học máy	0,545	0,545 (0,518–0,573)	0,565	0,2
LSTM	Học sâu	0,506	0,503 (0,495–0,513)	0,525	22,2
BiLSTM-Attention	Học sâu	0,634	0,488 (0,447–0,529)	0,488	38,3
BERT (tinh chỉnh)	Transformer	0,719	0,512 (0,470–0,553)	0,525	8.312,2
RoBERTa (tinh chỉnh)	Transformer	0,783	0,500 (0,500–0,500)	0,523	2.903,5

Ghi chú: ba mô hình rừng và ba mô hình học máy dùng cùng đầu vào bộ năm ở cả hai cách nên BA không đổi; thời gian huấn luyện lấy từ lần chạy Cách 2 trên cùng một máy CPU. Đây là một lần chia trên mẫu 3.000 đánh giá, dùng để so sánh tương đối giữa các mô hình trong cùng điều kiện.

Bảng J.9: Cấu hình huấn luyện của 10 mô hình trong thực nghiệm so sánh thống nhất (Bảng J.8)

Mô hình	Đầu vào	Cấu hình chính	Huấn luyện / dừng sớm
Rừng phân loại (F_{CF})	Bộ năm (e, a, h, t): target encoding (m = 10), chuẩn hóa Z-score	300 cây, Bootstrap cân bằng lớp (BalancedBoot)	Không lặp theo epoch
Rừng mật độ (F_{DF})	Bộ năm (e, a, h, t): target encoding (m = 10), chuẩn hóa Z-score	DBSCAN Euclidean (minPts = 5, eps theo phân vị khoảng cách k-láng giềng); mỗi cụm một rừng 300 cây BalancedBoot; trung bình xác suất	Không lặp theo epoch

Mô hình	Đầu vào	Cấu hình chính	Huấn luyện / dừng sớm
Rừng đa tạp (F_{MF})	Bộ năm (e, a, h, t): target encoding (m = 10), chuẩn hóa Z-score	Ma trận W từ rừng không giám sát 300 cây trên mẫu con 5.000 dòng; DBSCAN trên $\sqrt{(1 - W)}$ (minPts = 5); mỗi cụm một rừng BalancedBoot; Stacking (meta-learner rừng 300 cây trên tập xác thực)	Không lặp theo epoch
Random Forest	Bộ năm (e, a, h, t): target encoding (m = 10), chuẩn hóa Z-score	randomForest, 300 cây, tham số mặc định	Không lặp theo epoch
Gradient Boosting	Bộ năm (e, a, h, t): target encoding (m = 10), chuẩn hóa Z-score	gbm, 300 cây, độ sâu tương tác 3, phân phối Bernoulli	Không lặp theo epoch
SVM (tuyến tính)	Bộ năm (e, a, h, t): target encoding (m = 10), chuẩn hóa Z-score	e1071::svm, nhân tuyến tính, ước lượng xác suất	Không lặp theo epoch
LSTM	Văn bản đánh giá gốc	Từ điển 20.000 từ, nhúng 128 chiều, 64 đơn vị ẩn, độ dài tối đa 128	Adam, lr = 10^{-3}, batch 32, tối đa 10 epoch
BiLSTM-Attention	Văn bản đánh giá gốc	Như LSTM, LSTM hai chiều kèm lớp chú ý (attention)	Adam, lr = 10^{-3}, batch 32, tối đa 10 epoch

Mô hình	Đầu vào	Cấu hình chính	Huấn luyện / dừng sớm
BERT (tinh chỉnh)	Văn bản đánh giá gốc	bert-base-multilingual-cased, độ dài tối đa 128	AdamW, lr = 2×10^{-5}, batch 16, tối đa 3 epoch
RoBERTa (tinh chỉnh)	Văn bản đánh giá gốc	roberta-base, độ dài tối đa 128	Như BERT

Ghi chú: mọi mô hình dùng cùng dữ liệu (3.000 đánh giá yelpHotelData, hạt giống 2026), cùng cách chia theo đánh giá 60/20/20 và chọn ngưỡng quyết định trên tập xác thực theo Balanced Accuracy; toàn bộ thực nghiệm chạy trên CPU, không dùng GPU (LSTM, BiLSTM-Attention bằng gói torch cho R; BERT, RoBERTa bằng PyTorch/Transformers). Thời gian huấn luyện tại Bảng J.8.

Bảng J.10: Ảnh hưởng của cách lấy mẫu và cách chia dữ liệu đến chỉ số đánh giá (Random Forest, yelpHotelData, 112.275 bộ năm)

Cách lấy mẫu 112.275 bộ năm	Cách chia dữ liệu	Tỷ lệ dòng Test có dòng cùng đánh giá trong Train	Accuracy	Recall	F1	Balanced Accuracy
Ngẫu nhiên trong toàn tệp	Theo dòng bộ năm (như bài báo)	19,1%	0,664	0,866	0,766	0,589
Ngẫu nhiên trong toàn tệp	Theo đánh giá gốc	0%	0,611	0,833	0,732	0,527
112.275 dòng đầu tệp	Theo dòng bộ năm (như bài báo)	96,0%	0,946	0,984	0,960	0,930
112.275 dòng đầu tệp	Theo đánh giá gốc	0%	0,611	0,793	0,725	0,537

Ghi chú: "dòng cùng đánh giá" là dòng bộ năm có cùng (e, h, t) và cùng nhãn với một dòng thuộc tập huấn luyện. Tỷ lệ positive của mẫu là 63,9% (lấy ngẫu nhiên) và 65,1% (dòng đầu tệp).

Bảng J.11: So sánh thống nhất 10 mô hình trên yelpHotelData (10.000 đánh giá, 5 hạt giống; trung bình ± độ lệch chuẩn trên tập kiểm tra)

Mô hình	Nhóm	Đầu vào	Balanced Accuracy	ROC-AUC	Accuracy	F1-macro	Thời gian huấn luyện (s)
Rừng phân loại (F_{CF})	Đề xuất	Bộ năm	0,557 ± 0,009	0,572	0,584	0,551	2,1
Rừng mật độ (F_{DF})	Đề xuất	Bộ năm	0,561 ± 0,006	0,586	0,592	0,558	3,9
Rừng đa tạp (F_{MF})	Đề xuất	Bộ năm	0,534 ± 0,010	0,544	0,556	0,531	943,6
Random Forest	Học máy	Bộ năm	0,554 ± 0,011	0,572	0,569	0,543	2,5
Gradient Boosting	Học máy	Bộ năm	0,558 ± 0,006	0,580	0,575	0,548	1,5
SVM (tuyến tính)	Học máy	Bộ năm	0,568 ± 0,009	0,590	0,589	0,562	7,7
LSTM	Học sâu	Văn bản gốc	0,518 ± 0,008	0,526	0,597	0,493	45,8
BiLSTM-Attention	Học sâu	Văn bản gốc	0,701 ± 0,004	0,774	0,698	0,687	104,1
BERT (tinh chỉnh) (*)	Transformer	Văn bản gốc	0,755 ± 0,013	0,836	0,752	0,742	130,8
RoBERTa (tinh chỉnh)	Transformer	Văn bản gốc	0,807 ± 0,007	0,895	0,818	0,805	108,2

Ghi chú: (*) BERT tính trên 4 lần chạy hội tụ; ở hạt giống 2030, BERT không hội tụ (Balanced Accuracy 0,500), tính cả lần này BERT đạt 0,704 ± 0,114. Thời gian huấn luyện của các mô hình dùng bộ năm và LSTM, BiLSTM-Attention đo trên CPU 12 luồng; BERT, RoBERTa đo trên GPU NVIDIA L4.

Bảng J.12: Chênh lệch Balanced Accuracy so với Random Forest (ghép cặp theo hạt giống, số liệu Bảng J.11)

Mô hình	Chênh lệch trung bình	Khoảng tin cậy 95%	Số hạt giống cao hơn Random Forest
Rừng phân loại (F_{CF})	+0,0036	[−0,0033; 0,0105]	4/5
Rừng mật độ (F_{DF})	+0,0074	[−0,0042; 0,0190]	4/5

Mô hình	Chênh lệch trung bình	Khoảng tin cậy 95%	Số hạt giống cao hơn Random Forest
Rừng đa tạp (F_{MF})	−0,0198	[−0,0239; −0,0157]	0/5
Gradient Boosting	+0,0044	[−0,0062; 0,0150]	3/5
SVM (tuyến tính)	+0,0144	[0,0076; 0,0212]	5/5

Ghi chú: khoảng tin cậy 95% theo phân phối t (4 bậc tự do) của chênh lệch ghép cặp qua 5 hạt giống; khoảng chứa 0 nghĩa là chênh lệch chưa có ý nghĩa thống kê.

Bảng J.13: Các bản điều chỉnh của mô hình rừng đề xuất (chỉ dùng bộ năm; yelpHotelData, 10.000 đánh giá, 5 hạt giống)

Nhóm đặc trưng	Mô hình	Balanced Accuracy (TB ± SD)	ROC-AUC	Chênh lệch so với Random Forest cùng nhóm	Khoảng tin cậy 95%
A (như Bảng J.8)	Rừng phân loại (F_{CF})	0,547 ± 0,003	0,565	−0,0029	[−0,0063; 0,0005]
A (như Bảng J.8)	Rừng mật độ (F_{DF})	0,545 ± 0,019	0,564	−0,0052	[−0,0279; 0,0174]
A (như Bảng J.8)	Rừng đa tạp (F_{MF})	0,523 ± 0,012	0,533	−0,0272	[−0,0401; −0,0142]
A (như Bảng J.8)	Random Forest	0,550 ± 0,004	0,563	–	–
A (như Bảng J.8)	Gradient Boosting	0,544 ± 0,011	0,568	−0,0059	[−0,0191; 0,0073]
A (như Bảng J.8)	SVM (tuyến tính)	0,554 ± 0,015	0,577	+0,0038	[−0,0147; 0,0223]
A (như Bảng J.8)	Logistic Regression	0,553 ± 0,014	0,576	+0,0035	[−0,0134; 0,0204]

Nhóm đặc trưng	Mô hình	Balanced Accuracy (TB ± SD)	ROC-AUC	Chênh lệch so với Random Forest cùng nhóm	Khoảng tin cậy 95%
B (điều chỉnh)	F_{CF}+	0,548 ± 0,008	0,560	+0,0121	[0,0034; 0,0208]
B (điều chỉnh)	F_{DF}+	0,549 ± 0,011	0,565	+0,0134	[−0,0004; 0,0271]
B (điều chỉnh)	F_{MF}+	0,547 ± 0,011	0,557	+0,0106	[−0,0026; 0,0237]
B (điều chỉnh)	Random Forest	0,536 ± 0,006	0,558	–	–
B (điều chỉnh)	Gradient Boosting	0,539 ± 0,005	0,560	+0,0028	[−0,0074; 0,0130]
B (điều chỉnh)	SVM (tuyến tính)	0,515 ± 0,021	0,526	−0,0208	[−0,0528; 0,0112]
B (điều chỉnh)	Logistic Regression	0,551 ± 0,014	0,579	+0,0153	[−0,0003; 0,0309]

Ghi chú: nhóm A dùng đặc trưng như Bảng J.8 (target encoding ước lượng trên chính tập huấn luyện, thời gian theo ngày); nhóm B dùng target encoding ngoài fold (5 fold theo đánh giá), tần suất của thực thể, người đánh giá và năm; các rừng nhóm B dùng nút lá tối thiểu 150 dòng. Thực nghiệm này dùng bộ đặc trưng và quy trình riêng nên giá trị nhóm A khác nhẹ so với các dòng tương ứng tại Bảng J.11.

Bảng J.14: So sánh thống nhất 10 mô hình theo Cách 2 – mọi mô hình chỉ dùng bộ năm (yelpHotelData, 10.000 đánh giá, 5 hạt giống; trung bình ± độ lệch chuẩn trên tập kiểm tra)

Mô hình	**Nhóm**	**Balanced Accuracy**	**ROC-AUC**	**Accuracy**	**F1-macro**	**Thời gian huấn luyện (s)**	**Chênh lệch BA so với Random Forest [KTC 95%]**	**BA theo Cách 1** (Bảng J.11)
Rừng phân loại (F_{CF})	Đề xuất	0,557 ± 0,009	0,572	0,584	0,551	2,2	+0,0036 [−0,0033; 0,0105]	0,557
Rừng mật độ (F_{DF})	Đề xuất	0,561 ± 0,006	0,586	0,592	0,558	3,8	+0,0074 [−0,0042; 0,0190]	0,561
Rừng đa tạp (F_{MF})	Đề xuất	0,534 ± 0,010	0,544	0,556	0,531	1.044,3	−0,0198 [−0,0239; −0,0157]	0,534
Random Forest	Học máy	0,554 ± 0,011	0,572	0,569	0,543	2,6	–	0,554
Gradient Boosting	Học máy	0,558 ± 0,006	0,580	0,575	0,548	1,6	+0,0044 [−0,0062; 0,0150]	0,558
SVM (tuyến tính)	Học máy	0,568 ± 0,009	0,590	0,589	0,562	7,5	+0,0144 [0,0076; 0,0212]	0,568
LSTM	Học sâu	0,504 ± 0,008	0,505	0,454	0,369	62,0	−0,0494 [−0,0623; −0,0364]	0,518

Mô hình	Nhóm	Balanced Accuracy	ROC-AUC	Accuracy	F1-macro	Thời gian huấn luyện (s)	Chênh lệch BA so với Random Forest [KTC 95%]	BA theo Cách 1 (Bảng J.11)
BiLSTM-Attention	Học sâu	0,556 ± 0,012	0,577	0,539	0,530	95,7	+0,0017 [−0,0085; 0,0119]	0,701
BERT (tinh chỉnh)	Transformer	0,538 ± 0,009	0,556	0,534	0,517	375,7	−0,0157 [−0,0379; 0,0066]	0,755 (*)
RoBERTa (tinh chỉnh)	Transformer	0,523 ± 0,024	0,543	0,581	0,473	307,1	−0,0304 [−0,0554; −0,0054]	0,807

Ghi chú: cùng mẫu 10.000 đánh giá, cùng cách chia theo đánh giá 60/20/20, cùng 5 hạt giống (2026–2030) và cùng quy tắc chọn ngưỡng như Bảng J.11; sáu mô hình dùng bộ năm vì vậy có Balanced Accuracy trùng với Bảng J.11 (thời gian huấn luyện đo ở phiên chạy khác). LSTM, BiLSTM-Attention, BERT, RoBERTa đọc mỗi dòng bộ năm dưới dạng chuỗi “hotel <e> aspect <a> reviewer <h> time <t>” (tối đa 32 token), dự đoán theo từng dòng rồi lấy trung bình xác suất về đánh giá như các mô hình dùng bộ năm. Ở hạt giống 2026 và 2029, RoBERTa dự đoán toàn bộ là positive (Balanced Accuracy 0,500); ở hạt giống 2030, LSTM cũng dự đoán toàn bộ là positive. (*) BERT theo Cách 1 tính trên 4 lần chạy hội tụ (Bảng J.11). Khoảng tin cậy 95% theo phân phối t (4 bậc tự do) của chênh lệch ghép cặp qua 5 hạt giống.

Bảng J.15: So sánh thống nhất 10 mô hình trên yelpResData theo Cách 1 (10.000 đánh giá, 5 hạt giống; trung bình ± độ lệch chuẩn trên tập kiểm tra)

Mô hình	**Nhóm**	**Đầu vào**	**Balanced Accuracy**	**ROC-AUC**	**Accuracy**	**F1-macro**	**Thời gian huấn luyện (s)**	**Chênh lệch BA so với Random Forest [KTC 95%]**	**BA trên yelpHotelData** (Bảng J.11)
Rừng phân loại (F_{CF})	Đề xuất	Bộ năm	0,541 ± 0,010	0,557	0,518	0,514	11,9	−0,0019 [−0,0049; 0,0011]	0,557
Rừng mật độ (F_{DF})	Đề xuất	Bộ năm	0,543 ± 0,011	0,562	0,534	0,516	221,9	+0,0007 [−0,0100; 0,0114]	0,561
Rừng đa tạp (F_{MF})	Đề xuất	Bộ năm	0,514 ± 0,017	0,520	0,522	0,507	1.048,4	−0,0288 [−0,0472; −0,0105]	0,534
Random Forest	Học máy	Bộ năm	0,543 ± 0,009	0,560	0,528	0,518	14,7	–	0,554
Gradient Boosting	Học máy	Bộ năm	0,551 ± 0,009	0,568	0,573	0,547	8,4	+0,0079 [0,0029; 0,0128]	0,558
SVM (tuyến tính)	Học máy	Bộ năm	0,553 ± 0,010	0,580	0,576	0,543	13,5	+0,0098 [0,0007; 0,0190]	0,568

Mô hình	Nhóm	Đầu vào	Balanced Accuracy	ROC-AUC	Accuracy	F1-macro	Thời gian huấn luyện (s)	Chênh lệch BA so với Random Forest [KTC 95%]	BA trên yelpHotelData (Bảng J.11)
LSTM	Học sâu	Văn bản gốc	0,542 ± 0,007	0,555	0,595	0,537	45,1	−0,0011 [−0,0195; 0,0174]	0,518
BiLSTM-Attention	Học sâu	Văn bản gốc	0,735 ± 0,014	0,812	0,733	0,721	107,3	+0,1923 [0,1814; 0,2032]	0,701
BERT (tinh chỉnh)	Transformer	Văn bản gốc	0,769 ± 0,013	0,850	0,771	0,759	130,5	+0,2259 [0,2118; 0,2401]	0,755 (*)
RoBERTa (tinh chỉnh)	Transformer	Văn bản gốc	0,823 ± 0,015	0,911	0,828	0,817	108,2	+0,2798 [0,2664; 0,2932]	0,807

Ghi chú: mẫu 10.000 đánh giá lấy ngẫu nhiên từ tệp bộ năm yelpResData đã ghép văn bản đánh giá gốc; mỗi đánh giá nhà hàng sinh tối đa 19 dòng bộ năm (yelpHotelData: 3 dòng). Cùng quy trình như Bảng J.11: chia theo đánh giá 60/20/20, ngưỡng chọn trên tập xác thực theo Balanced Accuracy, xác suất các dòng bộ năm gộp về đánh giá bằng trung bình. Do số dòng bộ năm lớn hơn, mô hình SVM tuyến tính chỉ huấn luyện trên tối đa 30.000 dòng Train lấy ngẫu nhiên. Thời gian huấn luyện của các mô hình dùng bộ năm và LSTM, BiLSTM-Attention đo trên CPU 12 luồng; BERT, RoBERTa trên GPU NVIDIA L4. (*) BERT trên yelpHotelData tính trên 4 lần chạy hội tụ. Khoảng tin cậy 95% theo phân phối t (4 bậc tự do) của chênh lệch ghép cặp qua 5 hạt giống.

Bảng J.16: So sánh thống nhất 10 mô hình trên yelpResData theo Cách 2 – mọi mô hình chỉ dùng bộ năm (10.000 đánh giá, 5 hạt giống; trung bình ± độ lệch chuẩn trên tập kiểm tra)

Mô hình	**Nhóm**	**Balanced Accuracy**	**ROC-AUC**	**Accuracy**	**F1-macro**	**Thời gian huấn luyện (s)**	**Chênh lệch BA so với Random Forest [KTC 95%]**	**BA theo Cách 1** (Bảng J.15)
Rừng phân loại (F_{CF}) (**)	Đề xuất	0,541 ± 0,010	0,557	0,518	0,514	11,9	−0,0019 [−0,0049; 0,0011]	0,541
Rừng mật độ (F_{DF}) (**)	Đề xuất	0,543 ± 0,011	0,562	0,534	0,516	221,9	+0,0007 [−0,0100; 0,0114]	0,543
Rừng đa tạp (F_{MF}) (**)	Đề xuất	0,514 ± 0,017	0,520	0,522	0,507	1.048,4	−0,0288 [−0,0472; −0,0105]	0,514
Random Forest (**)	Học máy	0,543 ± 0,009	0,560	0,528	0,518	14,7	–	0,543
Gradient Boosting (**)	Học máy	0,551 ± 0,009	0,568	0,573	0,547	8,4	+0,0079 [0,0029; 0,0128]	0,551
SVM (tuyến tính) (**)	Học máy	0,553 ± 0,010	0,580	0,576	0,543	13,5	+0,0098 [0,0007; 0,0190]	0,553
LSTM	Học sâu	0,502 ± 0,003	0,501	0,580	0,388	93,9	−0,0410 [−0,0539; −0,0281]	0,542

Mô hình	Nhóm	Balanced Accuracy	ROC-AUC	Accuracy	F1-macro	Thời gian huấn luyện (s)	Chênh lệch BA so với Random Forest [KTC 95%]	BA theo Cách 1 (Bảng J.15)
BiLSTM-Attention	Học sâu	0,542 ± 0,010	0,563	0,559	0,531	167,6	−0,0011 [−0,0102; 0,0080]	0,735
BERT (tinh chỉnh)	Transformer	0,510 ± 0,021	0,522	0,639	0,420	692,8	−0,0332 [−0,0547; −0,0117]	0,769
RoBERTa (tinh chỉnh)	Transformer	0,502 ± 0,004	0,502	0,591	0,378	580,2	−0,0409 [−0,0552; −0,0266]	0,823

Ghi chú: cùng mẫu 10.000 đánh giá, cùng cách chia 60/20/20 theo đánh giá, cùng 5 hạt giống và cùng quy tắc chọn ngưỡng như Bảng J.15 (**) Sáu mô hình dùng bộ năm có cùng đầu vào ở hai cách nên không chạy lại; số liệu lấy từ Bảng J.15 (trên yelpHotelData, hai cách cho kết quả trùng nhau, Bảng J.14. Mô hình LSTM, BiLSTM-Attention, BERT, RoBERTa đọc mỗi dòng bộ năm dưới dạng chuỗi “restaurant <e> aspect <a> reviewer <h> time <t>” (tối đa 32 token), được huấn luyện trên 30.000 dòng bộ năm lấy ngẫu nhiên từ tập huấn luyện (giữ tỉ lệ lớp), dự đoán trên toàn bộ dòng của tập xác thực và tập kiểm tra rồi lấy trung bình xác suất về đánh giá. BERT dự đoán toàn bộ là positive ở hạt giống 2026, 2027, 2029, 2030; RoBERTa ở hạt giống 2026–2029; LSTM ở hạt giống 2029. Thời gian huấn luyện của LSTM, BiLSTM-Attention đo trên CPU 12 luồng; BERT, RoBERTa trên GPU NVIDIA L4. Khoảng tin cậy 95% theo phân phối t (4 bậc tự do) của chênh lệch ghép cặp qua 5 hạt giống.

Bảng J.17: Các bản điều chỉnh của mô hình rừng đề xuất trên yelpResData (chỉ dùng bộ năm; 10.000 đánh giá, 5 hạt giống)

Nhóm đặc trưng	**Mô hình**	**Balanced Accuracy (TB ± SD)**	**ROC-AUC**	**Chênh lệch so với Random Forest cùng nhóm**	**Khoảng tin cậy 95%**	**BA trên yelpHotelData** (Bảng J.13)
A (như Bảng J.8)	Rừng phân loại (F_{CF})	0,551 ± 0,013	0,567	−0,0044	[−0,0167; 0,0078]	0,547
A (như Bảng J.8)	Rừng mật độ (F_{DF})	0,535 ± 0,020	0,547	−0,0202	[−0,0405; 0,0001]	0,545
A (như Bảng J.8)	Rừng đa tạp (F_{MF})	0,535 ± 0,005	0,540	−0,0203	[−0,0300; −0,0105]	0,523
A (như Bảng J.8)	Random Forest	0,555 ± 0,007	0,572	–	–	0,550
A (như Bảng J.8)	Gradient Boosting	0,558 ± 0,015	0,575	+0,0030	[−0,0119; 0,0179]	0,544
A (như Bảng J.8)	SVM (tuyến tính)	0,569 ± 0,011	0,592	+0,0138	[0,0048; 0,0229]	0,554
A (như Bảng J.8)	Logistic Regression	0,566 ± 0,010	0,592	+0,0108	[0,0029; 0,0187]	0,553
B (điều chỉnh)	F_{CF}+	0,539 ± 0,011	0,560	+0,0022	[−0,0168; 0,0212]	0,548
B (điều chỉnh)	F_{DF}+	0,537 ± 0,009	0,560	+0,0005	[−0,0182; 0,0191]	0,549
B (điều chỉnh)	F_{MF}+	0,539 ± 0,011	0,560	+0,0019	[−0,0165; 0,0203]	0,547
B (điều chỉnh)	Random Forest	0,537 ± 0,013	0,553	–	–	0,536
B (điều chỉnh)	Gradient Boosting	0,557 ± 0,014	0,579	+0,0204	[0,0084; 0,0324]	0,539

Nhóm đặc trưng	Mô hình	Balanced Accuracy (TB ± SD)	ROC-AUC	Chênh lệch so với Random Forest cùng nhóm	Khoảng tin cậy 95%	BA trên yelpHotelData (Bảng J.13)
B (điều chỉnh)	SVM (tuyến tính)	0,540 ± 0,019	0,553	+0,0030	[−0,0219; 0,0278]	0,515
B (điều chỉnh)	Logistic Regression	0,566 ± 0,006	0,595	+0,0287	[0,0142; 0,0431]	0,551

Ghi chú: quy trình và đặc trưng như Bảng J.13; mẫu 10.000 đánh giá lấy từ toàn bộ tệp bộ năm yelpResData (không cần văn bản gốc), nên khác mẫu của Bảng J.15–J.16 và giá trị nhóm A khác nhẹ so với các dòng tương ứng tại Bảng J.15. Khoảng tin cậy 95% theo phân phối t (4 bậc tự do) của chênh lệch ghép cặp qua 5 hạt giống (2026–2030). SVM tuyến tính huấn luyện trên tối đa 30.000 dòng Train. F_{DF}+, F_{MF}+ chọn trọng số kết hợp a từ 0 đến 0,2 trên tập xác thực nên kết quả gần trùng với F_{CF}+. Ở hạt giống 2026 và 2030, DBSCAN chọn eps = 0 nên có cụm gồm toàn các điểm trùng nhau; rừng của cụm như vậy không thể tách nút nên được thay bằng dự đoán hằng số 0,5 (đầu ra của rừng cân bằng lớp khi không tách được nút).