# Text Mining and Social Media Mining

THS. LÊ NHẬT TÙNG

# Text categorization

THS. LÊ NHẬT TÙNG

# Text categorization

Text categorization, hay còn được gọi là ***phân loại văn bản***, là quá trình tự động gán nhãn hoặc phân loại các văn bản vào một hoặc nhiều danh mục đã được xác định trước dựa trên nội dung của chúng.

# Text categorization

Đây là một lĩnh vực quan trọng trong xử lý ngôn ngữ tự nhiên (NLP) và học máy, với nhiều ứng dụng thực tế như:

- Phân loại email (thư rác, không phải thư rác)
- Phân loại các bài báo tin tức theo chủ đề (thể thao, chính trị, giải trí...)
- Phân tích cảm xúc (tích cực, tiêu cực, trung lập)
- Phân loại văn bản theo ngôn ngữ
- Phân loại tài liệu theo thể loại

# Biểu diễn văn bản

**Vector đặc trưng của văn bản**

- **Cấu trúc**: x = (x(1), x(2), ..., x(p))
- **Thành phần**: Mỗi giá trị x(i) mã hóa sự hiện diện của:
  - Từ đơn (unigrams)
  - Cụm từ liên tiếp (n-grams)
  - Cụm từ có gắn thẻ (tagged phrases)
  - Thực thể có tên (Named Entities)
  - Các tham số khác trong văn bản

# Ví dụ Vector đặc trưng

- **Văn bản mẫu**: "Việt Nam có nền ẩm thực phong phú và đa dạng"
- **Vector đơn giản (unigrams)**:

x = (1, 1, 1, 1, 1, 1, 1, 0, 0, ...), với các thành phần tương ứng từng từ

Từ điển: {"Việt", "Nam", "có", "nền", "ẩm", "thực", "phong",

"phú", "và", "đa", "dạng", ...}

# Biểu diễn văn bản

## Mô hình Bag of Words & TF-IDF

- **Công thức**: $x(i) = TF(i,d) \times IDF(i)$
- **Trong đó**:
  - **TF (Term Frequency)**: Số lần xuất hiện của từ i trong văn bản d
  - **IDF (Inverse Document Frequency)**: $\log(N/n(i))$
    - N: Tổng số văn bản trong bộ sưu tập
    - n(i): Số văn bản chứa từ i

# Ví dụ Bag of Words & TF-IDF

- **Bộ dữ liệu**: 3 văn bản
- D1: "Việt Nam có nền ẩm thực phong phú"
- D2: "Ẩm thực Việt Nam rất đa dạng"
- D3: "Du lịch Việt Nam phát triển nhanh"
- **TF của từ "Việt" trong D1**: 1 (xuất hiện 1 lần)
- **IDF của từ "Việt"**: log(3/3) = log(1) = 0 (vì xuất hiện trong cả 3 tài liệu)
- **IDF của từ "ẩm thực"**: log(3/2) = 0.176 (vì xuất hiện trong 2 tài liệu)
- **TF-IDF của "Việt" trong D1**: 1 × 0 = 0
- **TF-IDF của "ẩm thực" trong D1**: 1 × 0.176 = 0.176

# Biểu diễn văn bản

## Chuẩn hóa vector đặc trưng

- **Mục đích**: Đảm bảo tính nhất quán và hiệu quả trong quá trình xử lý
- **Phương pháp**:
  - Chuẩn hóa L1/L2 (tổng = 1 hoặc độ dài đơn vị)
  - Giảm chiều dữ liệu (PCA, LSA)
  - Chọn lọc đặc trưng quan trọng nhất

# Ví dụ chuẩn hóa vector

- **Vector gốc**: (2, 1, 3, 0, 1) (tần suất từng từ trong một văn bản)
- **Chuẩn hóa L1**: (2/7, 1/7, 3/7, 0/7, 1/7) = (0.286, 0.143, 0.429, 0, 0.143)
- **Chuẩn hóa L2**: (2/√15, 1/√15, 3/√15, 0/√15, 1/√15) ≈ (0.516, 0.258, 0.775, 0, 0.258)

# Chuẩn hóa L1 (còn gọi là chuẩn hóa Manhattan):

- Chia mỗi giá trị cho tổng tất cả các giá trị (lấy giá trị tuyệt đối)
- Vector gốc: (2, 1, 3, 0, 1)
- Tổng các giá trị: |2| + |1| + |3| + |0| + |1| = 7
- Do đó, chia mỗi phần tử cho 7: (2/7, 1/7, 3/7, 0/7, 1/7)

# Chuẩn hóa L2 (còn gọi là chuẩn hóa Euclidean):

- Chia mỗi giá trị cho căn bậc hai của tổng bình phương các giá trị
- Vector gốc: (2, 1, 3, 0, 1)
- Tổng bình phương: $2^2 + 1^2 + 3^2 + 0^2 + 1^2 = 4 + 1 + 9 + 0 + 1 = 15$
- Căn bậc hai của tổng bình phương: $\sqrt{15}$
- Do đó, chia mỗi phần tử cho $\sqrt{15}$: $(2/\sqrt{15}, 1/\sqrt{15}, 3/\sqrt{15}, 0/\sqrt{15}, 1/\sqrt{15})$

# Các loại bài toán phân loại văn bản

- Bài toán kiểm tra số lớp của văn bản (Check the number of classes of the document )

- Bài toán phân loại nhị phân (Binary Text Classification)

- Bài toán phân loại đa lớp (Multiclass Classification)

- Bài toán phân loại đa nhãn (Multi-label Classification)

# Bài toán phân loại nhị phân (Binary Text Classification)

- **Định nghĩa**: Phân loại văn bản vào đúng một trong hai lớp
- **Đặc điểm**:
- Chỉ có hai lớp (positive/negative, spam/ham, relevant/irrelevant...)
- Mỗi văn bản phải thuộc về một và chỉ một lớp
- **Ví dụ thực tế**:
- Phân loại email (spam hay không spam)
- Phân tích cảm xúc đơn giản (tích cực hay tiêu cực)
- Phân loại bài viết (liên quan hay không liên quan)
- **Độ đo đánh giá phổ biến**:
- Accuracy, Precision, Recall, F1-score, AUC-ROC

# Bài toán phân loại đa lớp (Multiclass Classification)

•**Định nghĩa**: Phân loại văn bản vào đúng một trong nhiều lớp khác nhau
•**Đặc điểm**:
•Có nhiều hơn hai lớp (thường từ 3 đến hàng trăm lớp)
•Mỗi văn bản vẫn phải thuộc về một và chỉ một lớp
•**Ví dụ thực tế**:
•Phân loại tin tức theo chủ đề (thể thao, chính trị, giải trí, khoa học...)
•Phân loại văn bản theo thể loại (tiểu thuyết, thơ, kịch, truyện ngắn...)
•Phân loại ý kiến khách hàng theo sản phẩm
•**Phương pháp tiếp cận**:
•One-vs-Rest (OvR): xây dựng n mô hình nhị phân (n = số lớp)
•One-vs-One (OvO): xây dựng n(n-1)/2 mô hình nhị phân
•Native Multiclass: sử dụng thuật toán hỗ trợ nhiều lớp trực tiếp

# Bài toán phân loại đa nhãn (Multi-label Classification)

- **Định nghĩa**: Gán một hoặc nhiều nhãn cho mỗi văn bản
- **Đặc điểm**:
- Có nhiều lớp (thường từ vài lớp đến hàng nghìn lớp)
- Mỗi văn bản có thể thuộc về nhiều lớp cùng một lúc
- Các lớp không loại trừ lẫn nhau

# Bài toán phân loại đa nhãn (Multi-label Classification)

•**Ví dụ thực tế**:

•Gán thẻ cho bài viết blog (có thể thuộc nhiều chủ đề: "du lịch", "ẩm thực", "văn hóa"...)

•Phân loại bài báo khoa học (có thể thuộc nhiều lĩnh vực: "machine learning", "NLP", "computer vision"...)

•Phân loại tài liệu pháp lý (liên quan đến nhiều luật khác nhau)

•**Phương pháp tiếp cận**:

•Binary Relevance: xây dựng n mô hình nhị phân độc lập

•Classifier Chains: xây dựng n mô hình nhị phân phụ thuộc

•Label Powerset: chuyển đổi thành bài toán multiclass với 2^n lớp

•**Độ đo đánh giá phổ biến**:

•Hamming Loss, Exact Match Ratio, F1 (micro, macro, weighted)

# So sánh giữa các loại bài toán

| **Tiêu chí** | **Nhị phân (Binary)** | **Đa lớp (Multiclass)** | **Đa nhãn (Multi-label)** |
|---|---|---|---|
| Số lớp | 2 | >2 | >2 |
| Số nhãn cho mỗi văn bản | 1 | 1 | ≥1 |
| Độ phức tạp | Thấp nhất | Trung bình | Cao nhất |
| Ứng dụng phổ biến | Phân loại spam, phân tích cảm xúc | Phân loại tin tức theo chủ đề | Gán thẻ, phân loại nội dung đa dạng |

# Dữ liệu và quá trình huấn luyện

- Tập dữ liệu huấn luyện (Training Dataset)
- Giả định và nguyên lý học máy
- Quy trình huấn luyện và đánh giá
- Thách thức trong huấn luyện

# Tập dữ liệu huấn luyện (Training Dataset)

- **Định dạng cơ bản**: $S = \{(x_1,y_1), (x_2,y_2), ..., (x_n,y_n)\}$
- **Trong đó**:
- $x_i$: Vector đặc trưng của văn bản thứ i
- $y_i$: Nhãn lớp của văn bản thứ i
- n: Tổng số mẫu huấn luyện
- **Nhãn lớp**:
- Nhị phân: $y_i \in \{-1, +1\}$ hoặc $y_i \in \{0, 1\}$
- Đa lớp: $y_i \in \{1, 2, ..., k\}$ với k là số lớp
- Đa nhãn: $y_i \subseteq \{1, 2, ..., k\}$ (tập hợp các nhãn)

# Giả định và nguyên lý học máy

- **Giả định i.i.d (independently and identically distributed)**:
- Các mẫu huấn luyện độc lập với nhau
- Các mẫu huấn luyện tuân theo cùng một phân phối xác suất
- Đây là giả định quan trọng đảm bảo tính khái quát hóa của mô hình
- **Mục tiêu của quá trình học**:
- Tìm quy tắc phân loại tối ưu f: X → Y
- Giảm thiểu hàm mất mát trên tập huấn luyện
- Tối đa hóa khả năng dự đoán chính xác trên dữ liệu mới

# Quy trình huấn luyện và đánh giá

•**Chia dữ liệu**:

•Tập huấn luyện (Training set): Dùng để huấn luyện mô hình (~70-80%)

•Tập kiểm thử (Validation set): Dùng để điều chỉnh tham số (~10-15%)

•Tập đánh giá (Test set): Dùng để đánh giá hiệu suất cuối cùng (~10-15%)

•**Phương pháp kiểm tra chéo (Cross-validation)**:

•K-fold cross-validation: chia dữ liệu thành k phần

•Huấn luyện mô hình k lần, mỗi lần sử dụng k-1 phần làm tập huấn luyện, 1 phần làm tập kiểm thử

•Kết quả cuối cùng là trung bình của k lần huấn luyện

# Thách thức trong huấn luyện

- **Mất cân bằng dữ liệu (Imbalanced data)**:
- Số lượng mẫu giữa các lớp khác nhau rất nhiều
- Giải pháp: Oversampling, undersampling, SMOTE, class weights
- **Overfitting (quá khớp)**:
- Mô hình quá phức tạp, hoạt động tốt trên dữ liệu huấn luyện nhưng kém trên dữ liệu mới
- Giải pháp: Regularization, dropout, early stopping, data augmentation
- **Underfitting (dưới khớp)**:
- Mô hình quá đơn giản, không nắm bắt được mẫu trong dữ liệu
- Giải pháp: Tăng độ phức tạp mô hình, thêm đặc trưng, giảm regularization

# Các thuật toán phân loại văn bản

**Thuật toán truyền thống:**

- Support Vector Machines (SVM)
- Naïve Bayes
- k-Nearest Neighbors (k-NN)
- Decision Trees

# Các thuật toán phân loại văn bản

**Thuật toán truyền thống:**

- Support Vector Machines (SVM)
- Naïve Bayes
- k-Nearest Neighbors (k-NN)
- Decision Trees

**Các phương pháp Ensemble:**

- Boosting
- Random Forest

**Mạng nơ-ron và Deep Learning:**

- Neural Networks
- Mạng nơ-ron cho xử lý văn bản

**Phương pháp chuyên biệt:**

- Concept Minin
- Soft/Rough Set-based Learning

**Phương pháp xử lý ngôn ngữ tự nhiên:**

- Tf-idf và Biểu diễn văn bản
- Latent Semantic Indexing/Analysis (LSI/LSA)
- Phương pháp NLP hiện đại

# Đánh giá

Tương tự các bài toán học có giám sát

# Các thách thức và giải pháp trong phân loại văn bản

**Thách thức với số lượng lớn danh mục**

- **Vấn đề**: Khi có rất nhiều danh mục, việc sử dụng phân loại hai chiều (two-way classifiers) trở nên không khả thi về mặt tính toán
- **Tác động**:
  - Chi phí tính toán tăng theo cấp số nhân
  - Hiệu suất giảm do mỗi lớp có ít dữ liệu huấn luyện hơn
  - Không gian tìm kiếm quá rộng cho các tham số mô hình
- **Giải pháp**:
  - Sử dụng phân loại phân tầng (hierarchical classification)
  - Áp dụng các thuật toán có khả năng mở rộng tốt
  - Phương pháp giảm chiều dữ liệu

# Các thách thức và giải pháp trong phân loại văn bản

**Phân loại phân tầng (Hierarchical Classification)**

- **Nguyên lý**:
  - Tổ chức các danh mục theo cấu trúc phân cấp (cây)
  - Phân loại từ trên xuống dưới (top-down approach)
  - Tập trung vào từng tập con của dữ liệu ở mỗi nút
- **Cách tiếp cận**:
  - Local Classifier Approach: huấn luyện mô hình riêng cho mỗi nút trong cây
  - Global Classifier Approach: xây dựng một mô hình duy nhất cho toàn bộ cây phân cấp
- **Ưu điểm**:
  - Giảm độ phức tạp tính toán
  - Tận dụng mối quan hệ ngữ nghĩa giữa các danh mục
  - Thích hợp cho SVM và kNN trên từng cụm nhỏ dữ liệu

# Các thách thức và giải pháp trong phân loại văn bản

**Học hiệu quả từ dữ liệu huấn luyện thưa thớt**

- **Vấn đề**:
  - Thiếu dữ liệu cho một số danh mục
  - Mất cân bằng nghiêm trọng giữa các danh mục
  - Chất lượng dữ liệu không đồng đều
- **Giải pháp**:
  - **Data Augmentation**: Tạo thêm dữ liệu bằng cách biến đổi, dịch, paraphrase...
  - **Transfer Learning**: Sử dụng kiến thức từ miền tương tự
  - **Semi-supervised Learning**: Kết hợp dữ liệu có nhãn và không nhãn
  - **Few-shot Learning**: Học từ ít ví dụ

# Các thách thức và giải pháp trong phân loại văn bản

**Mô hình hóa chung các danh mục liên quan**

- **Nguyên lý**: Khai thác mối tương quan giữa các danh mục để cải thiện hiệu suất
- **Phương pháp**:
  - **Joint Modeling**: Xây dựng mô hình chung cho các danh mục có liên quan
  - **Multi-task Learning**: Huấn luyện mô hình đồng thời cho nhiều nhiệm vụ liên quan
  - **Dependency Networks**: Mô hình hóa rõ ràng các phụ thuộc giữa các danh mục
- **Ưu điểm**:
  - Tận dụng thông tin chung giữa các danh mục
  - Giảm thiểu vấn đề dữ liệu thưa thớt
  - Cải thiện độ chính xác tổng thể

# Các thách thức và giải pháp trong phân loại văn bản

**Các thách thức khác**

- **Xử lý ngôn ngữ tự nhiên**:
  - Đa nghĩa và đồng nghĩa
  - Sai lỗi chính tả và ngữ pháp
  - Biến đổi ngôn ngữ theo thời gian
- **Tính toán**:
  - Tối ưu hóa hiệu suất với dữ liệu lớn
  - Cân bằng giữa độ chính xác và thời gian đáp ứng
  - Giảm chi phí tính toán trong quá trình huấn luyện và dự đoán
- **Đạo đức và công bằng**:
  - Tránh thiên kiến trong phân loại
  - Đảm bảo tính minh bạch và khả năng giải thích
  - Bảo vệ thông tin nhạy cảm trong dữ liệu huấn luyện

# Thank you!