Detecting sarcasm on Vietnamese social media platforms

1st Nguyễn Duy Tâm Anh
Khoa Khoa học Máy tính
Trường Đại học Công nghệ Thông tin
Hồ Chí Minh, Việt Nam
22520054@gm.uit.edu.vn

2nd Nguyễn Quốc Khánh
Khoa Khoa học Máy tính
Trường Đại học Công nghệ Thông tin
Hồ Chí Minh, Việt Nam
22520646@gm.uit.edu.vn

3rd Nguyễn Mạnh Tường
Khoa Khoa học Máy tính
Trường Đại học Công nghệ Thông tin
Hồ Chí Minh, Việt Nam
22521626@gm.uit.edu.vn

4th Nguyễn Thế Vĩnh
Khoa Khoa học Máy tính
Trường Đại học Công nghệ Thông tin
Hồ Chí Minh, Việt Nam
22521677@gm.uit.edu.vn

5th Nguyễn Xuân Linh
Khoa Khoa học Máy tính
Trường Đại học Công nghệ Thông tin
Hồ Chí Minh, Việt Nam
22520775@gm.uit.edu.vn

Abstract

Trong thử nghiệm này, chúng tôi ứng dụng các kỹ thuật xử lý ngôn ngữ tự nhiên để phát hiện mỉa mai trong văn bản tiếng Việt trên các nền tảng mạng xã hội. Chúng tôi áp dụng các phương pháp như mô hình ngôn ngữ dựa trên transformer và nhúng ngữ cảnh vào tập dữ liệu đã được chú thích. Ngoài ra, các kỹ thuật xử lý đặc trưng ngôn ngữ tiếng Việt được tích hợp nhằm cải thiện độ chính xác và khả năng phát hiện châm biếm trong các ngữ cảnh đa dạng.

1. Giới thiệu

Mỉa mai (sarcasm) là một hình thức giao tiếp ngôn ngữ trong đó người nói sử dụng lời nói để diễn đạt ý nghĩa trái ngược với điều được nói ra, thường nhằm mục đích châm biếm hoặc tạo hiệu ứng hài hước.

Trên các nền tảng mạng xã hội như Facebook, Twitter, và TikTok, sắc thái mỉa mai xuất hiện phổ biến, nhưng việc nhận diện chúng không dễ dàng, đặc biệt trong ngữ cảnh tiếng Việt với đặc trưng đa nghĩa và sự phong phú về cách biểu đạt.

Như câu: "Đi một ngày đàng, học một sàng khôn", sẽ có hàm ý ẩn đằng sau của nó:

- Nghĩa đen: Đi xa sẽ học được nhiều điều mới.
- Nghĩa bóng: Khuyến khích trải nghiệm để tích lũy kiến thức.

Các ngôn ngữ khác cũng có tính đa nghĩa nhưng Tiếng Việt nổi bật hơn nhờ sự phong phú trong thanh điệu (6 thanh) và cách tổ hợp từ linh hoạt, làm tăng tính biểu cảm và đa nghĩa, một điểm đặc trưng khó có trong các ngôn ngữ khác.

Ngày nay với sự phát triển mạnh mẽ của các công nghệ sản xuất phần cứng máy tính, các mô hình ngôn ngữ ứng dụng Deep Learning có độ chính xác cao hơn, đặc biệt với sự xuất hiện của Transfomer giúp cho việc nhận diện trở nên chính xác hơn.

2. Bộ dữ liệu

2.1 Tổng quan

Trong đề tài này sử dụng bộ dữ liệu do ban tổ chức cuộc thi UIT Data Science cung cấp.

Được lấy từ các trang mạng xã hội Việt Nam, với hai kiểu dữ liệu : text và image

2.2 Cấu trúc của bộ dữ liệu

Bộ dữ liệu được chia thành 2 tập là train và test.

Tuy nhiên vì là dữ liệu cuộc thi nên tập test sẽ không có label.

Cả 2 tập đều có image và text :

- Image: được định dạng dưới dạng ảnh .jpg
- Text : sẽ được lưu dưới file .json với định dạng là 1 dictionary.

Ví dụ:

"id": {

"image":"9ef522967985161dcd81e821b3820fb8ab88a11ad2f7fb0cc7ef858eee50e980.jpg",

"caption": "nhất thắm rồi nhé!!!",

"label": "not-sarcasm"

}

Với:

- id: là số thứ tự
- image: là hình ảnh ứng với text đó
- caption: đoạn văn bản chú thích cho bức ảnh
- label: có 4 nhãn:
    - multi-sarcam : text và image kết hợp mang sắc thái mỉa mai
    - not-sarcasm: : text và image đều không có chứa sắc thái mỉa mai trong nội dung
    - text-sarcasm: Chỉ có caption là mỉa mai còn image không chứa sắc thái mỉa mai
    - image-sarcasm: Chỉ có image là mỉa mai còn text không chứa sắc thái mỉa mai

2.3 Tiền xử lý dữ liệu

- Chuyển caption sang chữ thường
- Loại bỏ hashtag
- Chuyển emoji sang dạng string

  ví dụ : 😀 chuyển thành :)))
- Loại bỏ các ký tự đặc biệt: [; . ; \ ; \ ; : ; = ;]
- Thay thế các dấu xuống dòng ‘ \n ’ bằng ký tự rỗng
- Chuẩn hóa dấu cách giữa các từ

2.4 Các thông số quan trọng

Số lượng dữ liệu:

	Số lượng dữ liệu
Train	10728
Test	1504

Số lượng labels trong tập Train:

	not-sarcasm	text-sarcasm	image-sarcasm	multi-sarcasm
Train	6062	77	442	4224

3. Metric

Sử dụng độ đo F1-score để đánh giá model:

$$\text{F1 (macro)} = \frac{1}{N}\sum_{i=1}^{N} F1_i$$

Với:

- $F1_i$: là F1 của mỗi class
- N: số class

4. Cài đặt và thực nghiệm

4.1 Trích xuất đặc trưng

4.1.1 Sử dụng pre-trained model CLIP

CLIP là mô hình đa phương thức mạnh mẽ, được huấn luyện để liên kết đặc trưng của hình ảnh và văn bản trong cùng một không gian nhúng, giúp hỗ trợ các tác vụ như tìm kiếm hoặc phân loại liên quan giữa hai miền dữ liệu này.

CLIP có khả năng trích xuất đặc trưng đồng thời từ hình ảnh và văn bản, đảm bảo các đặc trưng này nằm trong cùng một không gian nhúng và thể hiện mối liên quan chặt chẽ giữa hai miền dữ liệu.

Sử dụng trực tiếp model pre-train : 'openai/clip-vit-large-patch14'

Với:

- max sequence = 77
- truncation = True
- padding = True

Để trích xuất đặc trưng từ hình ảnh và văn bản bằng CLIP, ta sử dụng token đặc biệt (CLS) từ các encoder: text_encoder và vision_encoder. Sau đó, đặc trưng này được đưa qua projection layer để chuyển đổi về kích thước vector là 768.

4.1.2 Sử dụng pre-trained model ViT5

ViT5 là mô hình Transformer hướng tới việc xử lý ngôn ngữ Tiếng Việt, được thiết kế để tận dụng lượng token lớn hơn, giúp nâng cao khả năng hiểu ngữ nghĩa sâu hơn trong các tác vụ liên quan đến ngôn ngữ.

So với CLIP (chỉ hỗ trợ tối đa 77 token cho đầu vào văn bản), ViT5 hỗ trợ tối đa 1024 token, giúp mô hình xử lý tốt hơn các văn bản dài, đồng thời cải thiện khả năng nắm bắt ngữ nghĩa của văn bản Tiếng Việt.

Sử dụng model pretrain Vit5: ‘ VietAI/vit5-base ’

Đầu vào văn bản được xử lý qua ViT5, sử dụng token [CLS] từ last_hidden_state của encoder_block. Vector đặc trưng này tiếp tục được chiếu qua projection layer để đưa về kích thước vector đặc trưng là dim=768.

4.2 Kết hợp các đặc trưng

Kết hợp các đặc trưng bằng phương pháp Early Fusion:

- Sau khi trích xuất feature từ 2 model CLIP, ViT5:
  - Image feature, text feature từ model pretrained CLIP.
  - Text feature từ last_hidden_state của model ViT5
- Sử dụng phương pháp early fusion để kết hợp các feature theo chiều ngang (axis = 1)

Algorithm 1: Multimodal Early Fusion

Parameter Initialization

- Input A = $\{a_1, a_2, \dots a_n\}$ is set of text vectors, B= $\{b_1, b_2, \dots b_n\}$ is set of corresponding images, C= $\{c_1, c_2, \dots c_n\}$ is set of labels for A and B.
- n is the size of training set
- Split A, B and C into three subsets for 70% training, 10% validation and 20% testing.

1: **For 1 to 30 epochs do**
2: Extract feature vector using BERT/ALBERT model M_1 for text input A.
3: Extract feature vector using Inception-ResNet-v2 model M_2 for image input B.
4: Convert feature vectors obtained from M_1 and M_2 into unidimensional.
5: Perform concatenation of feature vectors.
6: Add a series of fully connected layers (dense, batch normalization, relu, dropout=0.4)
7: Apply binary sigmoid classifier and calculate prediction probabilities
8: Apply binary cross-entropy loss and Adam optimizer
9: **end for**
10: Evaluate the performance on test set

Thuật toán Early Fusion

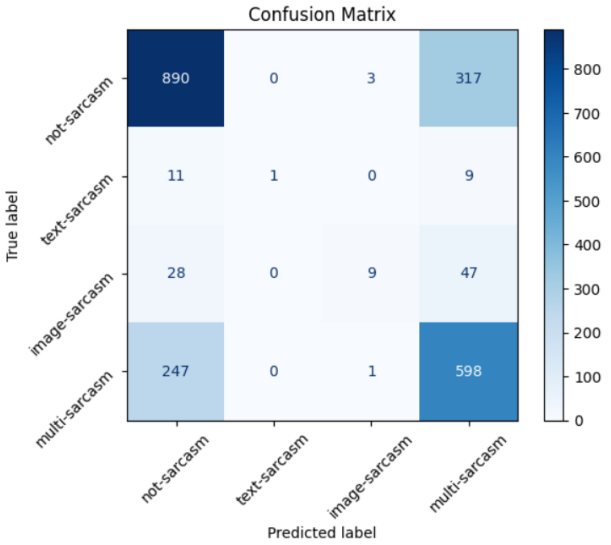
  - Sử dụng các feature được trích xuất từ các model khác nhau, sau đó kết hợp lại để đưa vào một model training khác sau đó sẽ trả về output.
  - Mục tiêu : Vì cần kết hợp cả text và image để tạo ra mối liên kết chặt chẽ trong việc dự đoán ra label, nên khi kết hợp lại sẽ học được cả feature của text và image sẽ không bị bias vào một dữ liệu khi train riêng lẻ.
- Sử dụng feature đã được kết hợp của tập train chia 8/2 để huấn luyện và đánh giá.

4.3 Mô hình huấn luyện

4.3.1 SVM

Vì dữ liệu có chiều lớn và xảy ra hiện tượng mất cân bằng giữa các lớp, nên việc lựa chọn model SVM sẽ giúp giải quyết được phần nào vấn đề này, tận dụng kernel trick = 'rbf ' để thu được tính tổng quát cho model giúp cho việc dự đoán sẽ chính xác hơn.

- Các feature sẽ được scale bằng StandardScaler giúp chuẩn hóa về dạng chuẩn, làm cân bằng dữ liệu hơn đặc biệt khi SVM sử dụng khoảng cách để đưa ra quyết định
- Finetune tham số n_components của phương pháp giảm chiều PCA với 0.85 (Giúp làm tăng tốc độ training) .
- Sử dụng model SVM để training, với việc đi finetune các tham số :
  - C =1
  - kernel = 'rbf'



Confusion Matrix khi data train được chia 8:2

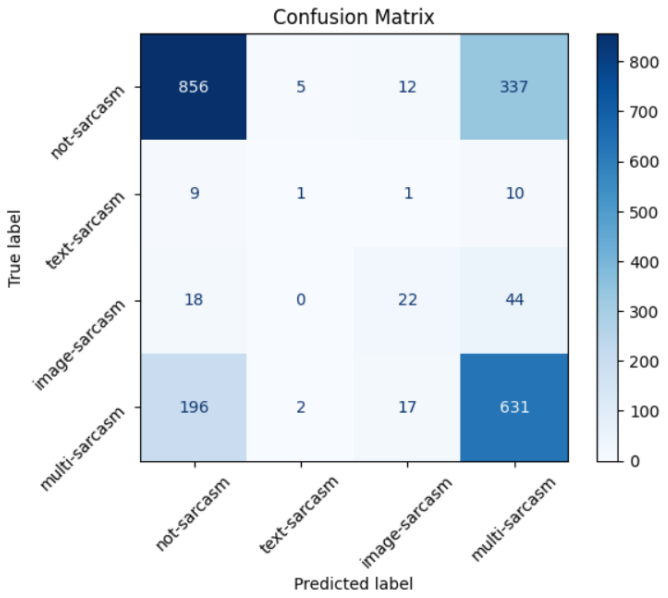
4.3.2 SVM + SMOTE

SMOTE (Synthetic Minority Over-sampling Technique) là một phương pháp phổ biến được sử dụng để giải quyết vấn đề mất cân bằng dữ liệu trong học máy. Phương pháp này tạo ra các mẫu dữ liệu mới thuộc lớp thiểu số (minority class) bằng cách tổng hợp dữ liệu nhân tạo từ các điểm dữ liệu hiện có. SMOTE giúp tăng số lượng mẫu thuộc lớp thiểu số để cân bằng với lớp đa số (majority class), từ đó cải thiện hiệu suất của mô hình khi phân loại.

Sau đó cũng sử dụng Scale (chuẩn hóa) và PCA (giảm chiều).

Sử dụng model SVM để training, với việc đi finetune các tham số :

- C =1
- kernel = 'rbf'

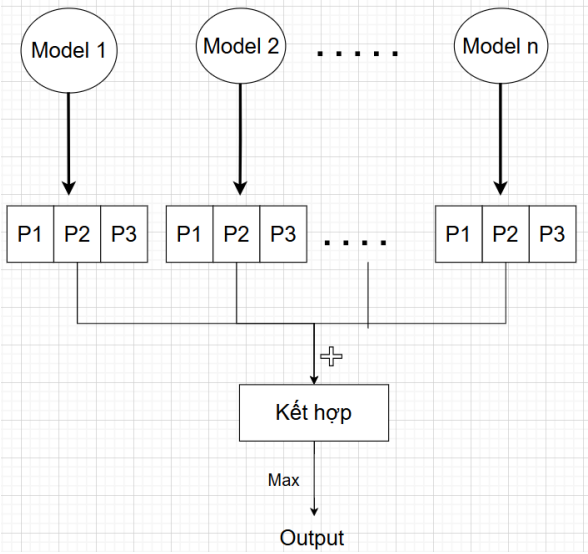


Confusion Matrix khi data train được chia 8:2

4.3.3 Ensemble + SMOTE

Ensemble là một kỹ thuật học máy trong đó nhiều mô hình (gọi là base models hoặc base learners) được huấn luyện để giải quyết cùng một bài toán, và sau đó kết hợp đầu ra của chúng bằng cách kết hợp các kết quả đó lại, nhằm cải thiện hiệu suất dự đoán.

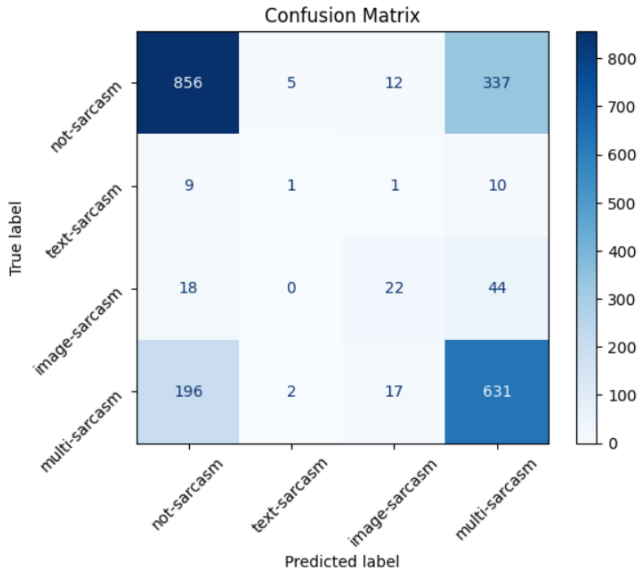
Sử dụng đầu ra là xác suất của từng label, sau đó sẽ cộng xác suất của các model lại r lấy giá trị max để làm giá trị label trả về.



Finetune các model độc lập trước khi kết hợp các đầu ra của các model được đưa vào

- Support Vector Machine
- Random Forest
- XGBoost.

Các outputs của model độc lập sẽ trả về xác suất của từng class, sau đó sẽ kết hợp lại với nhau bằng cách cộng tất cả lại sau đó sẽ lấy giá trị max của từng sample để trả về output.



Confusion Matrix khi data train được chia 8:2

5. Kết quả thực nghiệm

	Thử nghiệm với tập val (8/2)	Submit trên hệ thống chấm điểm
SVM	0.42	0.378
SVM + Smote	**0.45**	**0.4078**
Ensemble	0.44	0.3807

6. Kết luận

Dựa vào confusion matrix và F1-score có thể đưa ra được một số nhận xét như sau:

- Dự đoán chính xác class multi-sarcasm và text-sarcasm đóng vai trò quan trọng trong việc cải thiện hiệu suất tổng thể của mô hình, vì đây là một lớp phức tạp phản ánh sự kết hợp đa chiều giữa văn bản và hình ảnh.
- Việc sử dụng ensemble giúp tăng F1-score của class image-sarcasm, nhưng đồng thời lại làm giảm F1-score của class multi-sarcasm. Điều này dẫn đến hiệu suất tổng thể của mô hình bị giảm khi áp dụng trên thực tế.

Reference

https://huggingface.co/openai/clip-vit-large-patch14
https://huggingface.co/VietAI/vit5-base
https://www.sciencedirect.com/science/article/abs/pii/S0957417423010394
https://scikit-learn.org/stable/modules/svm.html
https://scikit-learn.org/stable/modules/ensemble.html