BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC CẦN THƠ

TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

NGUYÊN LÝ MÁY HỌC

Đề tài

ĐÁNH GIÁ CÁC MÔ HÌNH MÁY HỌC TRÊN DỮ LIỆU VĂN BẢN VÀ HÌNH ẢNH

Sinh viên: Dương Quốc Kiệt

MSSV: B2205883

Ngành: Công nghệ thông tin

Khóa: K48

Cần Thơ, 07/2025

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG

NGUYÊN LÝ MÁY HỌC

Đề tài

ĐÁNH GIÁ CÁC MÔ HÌNH MÁY HỌC TRÊN DỮ LIỆU VĂN BẢN VÀ HÌNH ẢNH

Người hướng dẫn:

TS. Phạm Thế Phi

Sinh viên thực hiện:

Dương Quốc Kiệt

MSSV: B2205883

Ngành: Công nghệ thông tin

Khóa: K48

Cần Thơ, 07/2025

NHẬN XÉT CỦA GIẢNG VIÊN

Thầy Phạm Thế Phi

LỜI CẢM ƠN

Trong quá trình làm đề tài niên luận cơ sở em xin chân thành cảm ơn thầy TS. Phạm Thế Phi – Trường Công nghệ Thông tin và Truyền Thông, đã tận tình chỉ bảo, định hướng, giúp em thực hiện đề tài này. Nhờ có sự chỉ bảo, định hướng quý báu của thầy đã giúp hiểu sâu sắc hơn về đề tài và giúp em hoàn thành đề tài này một cách tốt nhất.

Em cũng xin chân thành gửi lời cảm ơn đến các Thầy, Cô giảng viên Đại học Cần Thơ, đặc biệt là các Thầy, Cô đang công tác và giảng dạy tại Trường Công nghệ Thông tin và Truyền Thông đã truyền đạt cho em những kiến thức nền tảng và tạo điều kiện thuận lợi để em có thể áp dụng vào quá trình thực hiện đề tài này. Những kiến thức được học trong quá trình học tập tại trường đã giúp cho em có kiến thức, nền tảng vững chắc để hoàn thành đề tài này.

Bên cạnh đó, em xin gửi lời cảm ơn đến gia đình và bạn bè đã luôn đồng hành, động viên, chia sẻ tài liệu và hỗ trợ em cả về tinh thần lẫn học thuật trong suốt quá trình nghiên cứu và hoàn thành đề tài.

Mặc dù đã cố gắng hoàn thiện, nhưng không thể tránh khỏi những thiếu sót. Em mong nhận được những đóng góp của quý Thầy, Cô và bạn bè để giúp em có thể rút kinh nghiệm, hoàn thiện hơn trong những nghiên cứu sau này.

Cần Thơ, ngày 27, tháng 07 , năm2025

Người viết

Dương Quốc Kiệt

MỤC LỤC

DANH MỤC HÌNH

DANH MỤC BẢNG

DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt	Diễn giải
1	KNN	K-Nearest Neighbors
2	SVM	Support Vector Machine
3	RBF	Radial Basis Function
4	DT	Decision Tree
5	RF	Random Forest
6	ANN	Artificial Neural Network
7	NB	Naive Bayes
8	GIST	Generalized Search Tree
9	TF-IDF	Term Frequency - Inverse Document Frequency
10	VGG-19	Một kiến trúc mạng nơ-ron tích chập sâu có 19 lớp

ABSTRACT

In the context of strong digital technology development, data is becoming more and more diverse in both form and content, including text, images, audio, video and combined forms. The development of artificial intelligence, especially machine learning techniques, has shown great potential in recognizing, classifying and predicting on complex data sets. However, each machine learning model has its own characteristics in processing and exploiting information from each type of data, leading to differences in efficiency in specific application scenarios.

In practice, many problems, especially in areas such as e-commerce, healthcare, or smart transportation, require the ability to process diverse and combined data, such as product classification from images and descriptions or news recognition with illustrations. Therefore, determining the most suitable machine learning model for each type of data - text, image or a combination of both - is an important and necessary issue.

Based on that need, the topic "Evaluation of machine learning models on text and image data" was carried out to compare the effectiveness of algorithms such as: Linear SVM, K-Nearest Neighbors, Naive Bayes, Decision Tree, Random Forest, AdaBoost, RBF SVM and Artificial Neural Network when applied on different types of data. Through experiments and analysis, the topic aims to determine the optimal model for each type of data, thereby making appropriate recommendations for practical applications.

TÓM TẮT

Trong bối cảnh công nghệ số phát triển mạnh mẽ, dữ liệu ngày càng trở nên đa dạng về cả hình thức và nội dung, bao gồm văn bản, hình ảnh, âm thanh, video và các dạng kết hợp. Sự phát triển của trí tuệ nhân tạo, đặc biệt là các kỹ thuật học máy (machine learning), đã cho thấy tiềm năng lớn trong việc nhận dạng, phân loại và dự đoán trên các tập dữ liệu phức tạp. Tuy nhiên, mỗi mô hình học máy có đặc điểm riêng trong việc xử lý và khai thác thông tin từ từng loại dữ liệu, dẫn đến sự khác biệt về hiệu quả trong các tình huống ứng dụng cụ thể.

Trong thực tiễn, nhiều bài toán, đặc biệt trong các lĩnh vực như thương mại điện tử, y tế, hay giao thông thông minh, đòi hỏi khả năng xử lý dữ liệu đa dạng và kết hợp, ví dụ như phân loại sản phẩm từ hình ảnh và mô tả hoặc nhận diện tin tức có hình minh họa. Do đó, việc xác định mô hình học máy phù hợp nhất với từng loại dữ liệu - văn bản, hình ảnh hoặc sự kết hợp của cả hai - là một vấn đề quan trọng và cần thiết.

Xuất phát từ nhu cầu đó, đề tài "Đánh giá các mô hình máy học trên dữ liệu văn bản và hình ảnh" được thực hiện nhằm so sánh hiệu quả của các thuật toán như: Linear SVM, K-Nearest Neighbors, Naive Bayes, Cây quyết định, Rừng ngẫu nhiên, AdaBoost, RBF SVM và Mạng nơ-ron nhân tạo khi áp dụng trên các loại dữ liệu khác nhau. Thông qua thực nghiệm và phân tích, đề tài hướng đến việc xác định mô hình tối ưu cho từng dạng dữ liệu, từ đó đưa ra các đề xuất phù hợp cho ứng dụng trong thực tiễn.

CHƯƠNG 1: GIỚI THIỆU VÀ MÔ TẢ BÀI TOÁN

1. Đặt vấn đề

Trong thời đại bùng nổ công nghệ số hiện nay, dữ liệu ngày càng đa dạng về hình thức lẫn nội dung. Chúng không chỉ đơn thuần là dữ liệu về văn bản, mà dữ liệu hiện nay bao gồm nhiều loại như: hình ảnh, âm thanh, video, và các dạng dữ liệu kết hợp. Cùng với sự phát triển mạnh mẽ của trí tuệ nhân tạo, đặc biệt là kỹ thuật máy học, với khả năng nhận dạng, phân loại và dự đoán trên các tập dữ liệu phức tạp đã đạt được nhiều thành tựu.

Trong các mô hình máy học, mỗi mô hình có cách tiếp cận và khai thác các đặt trưng khác nhau. Việc lựa chọn các mô hình phải dựa vào cấu trúc của dữ liệu đầu vào và độ phức tạp của mô hình máy học. Một số loại mô hình thì có thể hoạt động tốt, trích xuất đặc trưng tốt với các dữ liệu dạng văn bản, nhưng một số lại không hiệu quả và ngược lại. Trong khi đó các yêu cầu thực tế trong cuộc sống, đặc biệt là trong lĩnh vực thương mại điện tử, y tế, giao thông thông minh,… Lại cần xử lý dữ liệu đa dạng, kết hợp phân loại dựa trên hình ảnh và mô tả hay phân loại tin tức kèm hình ảnh minh hoạ.

Tuy nhiên trên thực tế, với dữ liệu văn bản hay hình ảnh hoặc có sự kết hợp của cả hai loại dữ liệu này, thì mô hình máy học nào sẽ thật sự hiệu quả, mô hình nào sẽ cho ra kết quả phân loại tốt nhất. Việc giải đáp trả lời câu hỏi này có ý nghĩa quan trọng, đặc biệt trong các hệ thống tìm kiếm sản phẩm, phân loại dữ trên đánh giá, nhận diện các bài viết có kèm hình ảnh minh hoạ hay không.

Trên những vấn đề đó, đề tài " Đánh giá các mô hình máy học trên dữ liệu văn bản và hình ảnh" được thực hiện. Đề tài tập trung vào việc so sánh hiệu quả của các mô hình máy học như KNN, Naive Bayes, Cây quyết định, Rừng ngẫu nhiên, AdaBoost, SVM, và mạng nơ-ron nhân tạo khi áp dụng trên dữ liệu của văn bản, hình ảnh và kết hợp cả hai. Nhằm khảo sát, đánh giá và lựa chọn mô hình tối ưu trên từng loại dữ liệu. Thông qua hình thức thực nghiệm, kiểm chứng, đề tài sẽ làm rõ hơn về mối quan hệ giữa các dạng dữ liệu và mô hình máy học, từ đó có những đề xuất mô hình phù hợp trong các ứng dụng thực tế và sau này.

2. Mục tiêu đề tài

- Xây dựng hệ thống thực nghiệm có khả năng xử lý dữ liệu dạng văn bản và hình ảnh và dữ liệu kết hợp.
- Tiền xử lý và trích xuất đặc trưng cho phù hợp cho từng loại dữ liệu.
- Triển khai, huấn luyện và đánh giá mô hình trên các mô hình tiêu biểu như: KNN, Native Bayes, cây quyết định, rừng ngẫu nhiên, AdaBoost, SVM và mạng nơ-ron nhân tạo.
- Đánh giá hiệu quả của các mô hình trên tập dữ liệu văn bản.

- Đánh giá hiệu quả của các mô hình trên tập dữ liệu hình ảnh.
- Đánh giá và so sánh hiệu quả của các mô hình trên tập dữ liệu kết hợp văn bản và hình ảnh.
- Xác định các mô hình và kỹ thuật tốt nhất cho từng loại dữ liệu và ứng dụng.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu:

- Các mô hình máy học dùng cho bài toán như: K-Nearest Neighbors (KNN), Native Bayes, Cây quyết định (Descition Tree), Rừng ngẫu nhiên (Random Forest), AdaBoost, Support Vector Machine (SVM) và mạng nơ-ron nhân tạo (Artificial Neural Network –ANN).
- Các phương pháp trích xuất đặc trưng: văn bản sử dụng TF-IDF; hình ảnh sử dụng HOG (Histogram of Oriented Gradients), GIST, đặc trưng từ mạng nơ-ron học sâu (Deep Features) và các phương pháp kết hợp dữ liệu văn bản và hình ảnh.

Phạm vi nghiên cứu:

- Đánh giá và so sánh hiệu quả của các mô hình máy học khi áp dụng riêng trên từng tập dữ liệu văn bản, hình ảnh và trên dữ liệu kết hợp. Nhằm xác định mô hình phù hợp với từng loại dữ liệu và phương pháp tối ưu nhất cho độ chính xác phân loại cao.
- Tập trung vào việc khảo sát khả năng tổng quát hóa và hiệu suất mô hình dựa trên các đặc trưng đầu vào đã được trích xuất.

4. Phương pháp nghiên cứu

Mỗi mô hình sẽ được huấn luyện trên từng loại dữ liệu và đánh giá hiệu quả phân loại dựa vào các tiêu chí đánh giá. Kết quả từ quá trình chạy thực nghiệm sẽ được tổng hợp và so sánh đánh giá. Từ đó xác định loại mô hình nào phù hợp nhất trên từng tập dữ liệu, các bước thực hiện bao gồm:

- Bước 1: Tiền xử lý dữ liệu
    - Dữ liệu văn bản: làm sạch, chuẩn hóa, biểu diễn dưới dạng vector bằng phương pháp TF-IDF.
    - Dữ liệu hình ảnh: chuẩn hóa kích thước, chuyển đổi màu và trích xuất đặc trưng bằng các phương pháp HOG, GIST hoặc deep features.
- Bước 2: Huấn luyện mô hình

- Huấn luyện từng mô hình học máy riêng biệt trên từng loại dữ liệu: văn bản, hình ảnh.
- Kết hợp đặc trưng từ hai loại dữ liệu (early fusion), rồi huấn luyện lại các mô hình học máy trên tập dữ liệu kết hợp.

- Bước 3: Đánh giá mô hình
  - Đánh giá hiệu quả của mô hình dựa vào các chỉ số đánh giá.
  - Mô hình được thử nghiệm nhiều lần để đảm bảo độ chính xác.
- Bước 4: Phân tích và tổng hợp kết quả
  - Phân tích tổng hợp, ghi nhận kết quả và so sánh hiệu suất giữa các mô hình khác nhau trên cùng một loại dữ liệu văn bản, hình ảnh và giữa các loại dữ liệu kết hợp với nhau.
  - Đưa ra nhận xét xác định mô hình tối ưu nhất cho từng loại dữ liệu.

5. Kết quả đạt được

Đánh giá được hiệu quả phân loại của từng mô hình máy học dựa trên dữ liệu văn bản, hình ảnh và kết hợp dữ liệu,

Xác định được mô hình có độ chính xác cao nhất khi xử lý trên từng loại dữ liệu riêng biệt và kết hợp.

Phân tích các nguyên nhân và giải thích tại sao một số mô hình lại cho kết quả tốt hơn trong từng trường hợp.

Xác định được phương thức kết hợp của mô hình nào giữa văn bản và hình ảnh hiệu quả nhất để tăng độ chính xác khi phân loại.

6. Môi trường thực nghiệm

Toàn bộ quá trình huấn luyện và đánh giá các mô hình học máy trong đề tài được thực hiện trên cùng một môi trường như sau

- Môi trườn chạy thực nghiệm: Google Colab
- CPU: Intel(R) Xeon(R) @ 2.20GHz (2 nhân vật lý).
- RAM: 13 GB

7. Bố cục bài báo cáo

Chương 1: Giới thiệu và mô tả bài toán
Chương 2: Tập dữ liệu và các mô hình
Chương 3: Huấn luyện và đánh giá
Chương 4: Kết luận và hướng phát triển

CHƯƠNG 2: TẬP DỮ LIỆU VÀ CÁC MÔ HÌNH MÁY HỌC

1. Tổng qua về tập dữ liệu

Tập dữ liệu được sử dụng trong nghiên cứu là tập dữu liệu đa phương tiện (multimedia dataset), bao gồm hai loại thông tin chính: dữ liệu về hình ảnh và văn bản đi kèm. Mỗi mẫu dữ liệu đại diện cho một loại sản phẩm cụ thể được gán nhãn thuộc một trong hai lớp là: giày (shoes) hoặc túi xách (bags). Tập dữ liệu này được thiết kế để phục vụ cho bài toán phân loại, đồng thời cho phép đánh giá hiệu quả của các mô hình học máy trên dữ liệu văn bản, hình ảnh và sự kết hợp giữa hai loại dữ liệu.

Hình 2.1 Một mẫu dữ liệu hình ảnh mô tả túi xách trong tập dữ liệu.

Hình 2.1 minh họa một mẫu dữ liệu hình ảnh thuộc lớp "túi xách". Hình ảnh này là đại diện trực quan cho sản phẩm và được sử dụng làm đầu vào cho các mô hình học máy khi xử lý dữ liệu dạng hình ảnh.

```
Satisfy your sweet tooth with the delicious Brown Sugar hobo from Lucky Brand&reg;
handbags. ; Made of genuine leather. ; Holds your wallet, sunglasses, personal technology
and a small cosmetic case. ; Single handle. Handle drop: 6 length. ; Peace sign hardware
detail on strap. ; Top zip closure. ; Interior lining with a bac...
```

Hình 2.2 Một mẫu dữ liệu văn bản mô tả túi xách trong tập dữ liệu

Dữ liệu văn bản tương ứng với Hình 2.1 được thể hiện trong Hình 2.2. Nội dung mô tả có thể bao gồm kiểu dáng, thương hiệu, màu sắc, chất liệu, hoặc các đặc điểm liên quan đến

sản phẩm túi xách. Đây là nguồn thông tin đầu vào cho mô hình khi xử lý dữ liệu dạng văn bản.

Hình 2.3 Một mẫu dữ liệu hình ảnh mô tả giày trong tập dữ liệu

Hình 2.3 minh họa một mẫu hình ảnh thuộc lớp "giày". Giống như ảnh túi xách, hình ảnh này đại diện cho một sản phẩm thực tế và được dùng trong quá trình huấn luyện, đánh giá mô hình phân loại với đầu vào là hình ảnh.

```
Known for their edgy, modern designs suitable for
every fashionable woman, Balenciaga proves that
true style is never hard to achieve.
```

Hình 2.4 Một mẫu dữ liệu văn bản mô tả giày trong tập dữ liệu

Dữ liệu văn bản mô tả cho sản phẩm giày (Hình 2.3) được thể hiện trong Hình 2.4. Phần mô tả có thể bao gồm các thông tin như loại giày, màu sắc, thương hiệu, chất liệu, công dụng, v.v. Đây là dữ liệu giúp mô hình hiểu rõ hơn về đặc trưng văn bản của sản phẩm.

Tập dữ liệu về văn bản và hình ảnh được phân chia theo tỷ lệ dưới bảng sau:

Bảng 2.1 Tỷ lệ giữa tập train và test trong dữ liệu

Số lượng	Túi xách		Giày	
	Train	Test	Train	Test
Văn bản	18	20	18	20
Hình ảnh	18	20	18	20

2. Tổng qua về các thuật toán máy học

Trong lĩnh vực máy học, các thuật toán sẽ có những đặc trưng riêng phù hợp với từng loại dữ liệu khác nhau. Trong đề tài nhiều thuật toán được sử dụng để phân loại trên các loại dữ liệu khác nhau. Mỗi thuật toán có nhưng ưu điểm và nhược điểm riêng, tuỳ thuộc vào loại bài toán và kiểu dữ liệu đầu vào.Dưới đây là chi tiết tổng quan một số thuật toán được sử dụng.

2.1. K láng giềng gần nhất

K láng giềng gần nhất (K Nearest Neighbors – KNN): là một thuật toán đơn giản sử dụng cho bài toán mục đích phân loại và hồi quy. KNN là một thuật toán phân loại phi tham số, nổi bật bởi sự đơn giản nhưng mang lại hiệu quả cao trong nhiều trường hợp thực tế [1]. Thuật toán này không yêu cầu giai đoạn huấn luyện mô hình, mà thực hiện phân loại trực tiếp dựa trên toàn bộ tập dữ liệu huấn luyện đã có.

Để thực hiện phân loại một dữ liệu mới gọi là t, thuật toán sẽ tìm k điểm dữ liệu lân cận trong tập dữ liệu huấn luyện có khoảng cách gần nhất với t – nên mới gọi là k láng giềng gần nhất. Tập hợp này tạo thành một lân cận xung quanh t. Quyết định phân loại sau đó được đưa ra dựa trên nguyên tắc "biểu quyết đa số" từ các nhãn của k láng giềng .Ngoài ra, một số phiên bản cải tiến của KNN có thể áp dụng trọng số theo khoảng cách, trong đó các điểm gần hơn sẽ có ảnh hưởng lớn hơn đến kết quả phân loại [2].

Tuy nhiên, một yếu tố quan trọng ảnh hưởng trực tiếp đến kết quả của quá trình phân loại là dựa vào việc lựa chọn k. Nếu giá trị k quá nhỏ (ví dụ 1 hoặc 2), thuật toán trở nên dễ nhạy cảm với nhiễu và dẫn đến kết quả không ổn định làm sai lệch, nếu k quá lớn mô hình có thể làm mờ ranh giới giữa các lớp, gây cản trở khó khăn trong việc phân lớp dữ liệu. Do vậy, việc lựa chọn k tối ưu là một thách thức. Một cách tiếp cận là chúng ta sẽ thử nghiệm k trên nghiệm trường hợp với các giá trị k khác nhau và chọn ra giá trị k mang lại độ chính xác cao nhất trên tập kiểm tra.

Một quy tắc thực nghiệm phổ biến là chọn $k = \sqrt{n}$, trong đó n là số lượng mẫu, giá trị k thường là số lẻ để tránh biểu quyết hoà trong phân loại nhị phân.

Dưới đây là 3 cách cơ bản để tính khoảng cách trong 2 điểm dữ liệu x, y có k thuộc tính như sau:

Khoảng cách Euclid (Euclidean Distance)

$$d(x,y) = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

Khoảng cách Manhattan (Manhattan Distance)

$$d(x,y) = \sum_{i=1}^{k} | x_i - y_i|$$

Khoảng cách Minkowski

$$d(x,y) = (\sum_{i=1}^{k} |x_i - y_i|^p)^{\frac{1}{p}}$$

Thuật toán KNN có một nhược điểm lớn là chi phí tính toán cao khi phân loại trên các mẫu dữ liệu mới. Điều này xuất phát từ bản chất lười học của KNN, toàn bộ quá trình tính toán diễn ra ở giai đoạn dự đoán, thay vì trong giai đoạn huấn luyện. Điều đó khiến cho KNN trở nên kém phù hợp hơn với các hệ thống yêu cầu phân loại động hoặc quy mô lớn. Một số ưu điểm của thuật toán KNN là thuật toán đơn giản, dễ dàng triển khai cho các bài toán, có khả năng chịu đựng và chống lại nhiễu phổ biến trong tập dữ liệu được sử dụng để huấn luyện, nó nhanh chóng dễ diễn giải ngay cả khi tập dữ liệu lớn.

Mặc dù vậy, kNN vẫn là một trong những phương pháp phân loại hiệu quả và đã được ứng dụng thành công từ rất sớm trong lĩnh vực phân loại văn bản, đặc biệt trên tập dữ liệu chuẩn Reuters dùng trong đánh giá thuật toán phân loại văn bản [3].

2.2. Máy học véc-tơ hỗ trợ

Máy học véc-tơ hỗ trợ (Support Vector Machine – SVM): Là thuật toán máy học được sử dụng cho cả bài toán phân lớp và hồi quy. Nhiều nghiên cứu gần đây cho thấy SVM thường đạt hiệu suất cao hơn các thuật toán phân loại dữ liệu khác về độ chính xác phân loại [4]. SVM đã được áp dụng thành công trong nhiều lĩnh vực như:

phân loại văn bản, nhận dạng chữ viết tay, nhận dạng âm thanh, phân loại và phát hiện đối tượng trong ảnh, phân tích dữ liệu gene biểu hiện vi mô, v.v.

Mục tiêu chính của SVM là tìm ra một siêu mặt phẳng (hyperplane) tối ưu nhất để phân tách các lớp dữ liệu khác nhau trong không gian đặc trưng. SVM là tối đa hóa khoảng cách (margin) giữa siêu mặt phẳng và các điểm dữ liệu gần nhất thuộc hai lớp được gọi là các vector hỗ trợ (support vectors). Chính nhờ đặc điểm đó, SVM có khả năng tạo ra một ranh giới phân loại chắc chắn và giảm nguy cơ overfitting.

Có hai loại chính của SVM là Linear SVM và Non-linear SVM. Với Linear SVM, dữ liệu có thể phân tách bằng một đường thẳng (trong không gian hai chiều) hoặc một mặt phẳng (trong không gian ba chiều). Trong khi đó, Non-linear SVM được sử dụng khi dữ liệu không thể phân tách bằng một đường thẳng đơn giản [5].

Khi đó, SVM sử dụng kỹ thuật ánh xạ hàm hàm nhân (kenel function). Nhờ có kenel dữ liệu được ánh xạ lên một không gian đặc trưng có số chiều cao hơn, nơi nó có thể trở nên đẽ phân tách bằng siêu phẳng hơn. Một số kernel thường được sử dụng là: Linear Kernel, Polynomial Kernel, Gaussian Radial Basis Function (RBF), và Sigmoid Kernel. SVM hoạt động bằng việc xác định các véc-tơ hỗ trợ và sau đó tìm siêu phẳng dựa trên các điểm tối ưu này. Trong đó, Gaussian RBF là kernel phổ biến nhờ khả năng xử lý tốt dữ liệu phi tuyến. SVM có hai biến thể tiêu biểu: Linear SVM, thích hợp cho dữ liệu tuyến tính và RBF SVM, thường được dùng cho dữ liệu phi tuyến.

SVM hoạt động tốt trongcasc không gian mà số chiều cao (high-dimensional space), SVM tiết kiệm bộ nhớ vì chỉ sử dụng các vector hỗ trợ để xây dựng mô hình. Tuy vậy, SVM đòi hỏi lượng tài nguyên tính toán lớn đặc biệt là các tập dữ liệu và các hàm nhân phức tạp, nó cũng dễ bị ảnh hưởng bởi nhiễu hoặc sự chồng lắp của các lớp, việc chọn kernel phù hợp cũng như điều chỉnh siêu tham số là tương đối phức tạp.

2.3. Cây quyết định

Thuật toán Cây quyết định (Decision Tree – DT) là một phương pháp học máy có giám sát (supervised learning), được sử dụng rộng rãi trong các bài toán phân loại và hồi quy, tuy nhiên phổ biến nhất vẫn là trong phân loại. Mô hình cây quyết định có cấu trúc phân nhánh: bắt đầu từ nút gốc (root node), dữ liệu được chia ra thông qua các nút trong (decision nodes), cho đến khi đến các nút lá (leaf nodes) là nơi đưa ra kết quả đầu ra cuối cùng. Tại mỗi nút trong, mô hình sẽ kiểm tra điều kiện dựa trên các thuộc tính của dữ liệu để chia thành các nhánh con. Các nhánh này tiếp tục được chia nhỏ cho đến khi đạt được độ chính xác nhất, nghĩa là tất cả dữ liệu trong nhánh thuộc về cùng một lớp hoặc không thể chia nhỏ tiếp [5].

Trong giải thuật cây quyết định, việc lựa chọn thuộc tính để phân tách là yếu tố quan trọng đóng vai trò then chốt. Có hai chỉ số đo phổ biến được sử dụng là Information Gain (độ lợi thông tin) và Gini Index (chỉ số Gini).

Information Gain tính toán thông tin được cung cấp bởi một thuộc tính về lớp (class) khi biết giá trị của một thuộc tính cụ thể. Nó dùng để đo mức độ hỗn loạn (entropy) của tập dữ liệu sau khi phân chia theo thuộc tính đó. Một thuộc tính mà có độ Information Gain cao sẽ giúp phân chia dữ liệu tốt hơn, làm cho các nút con trở nên đồng nhất hơn về mặt phân loại. Độ lợi thông tin được biểu diễn bằng toán học, được tính bằng đỗ hỗn loạn thông tin trước khi phân hoạch trừ đỗ hỗn loạn thông tin sau khi phân hoạch. Công thức tính Information Gain được biểu diễn như sau:

$$Information\ Gain =$$

$$Entropy(S) - \left(Weighted\ Average\ \times\ Entropy(Every\ Feature)\right)$$

Trong đó, Entropy(S) là đo lường độ hỗn loạn ban đầu của tập dữ liệu S, còn phần tử sau nó là trung bình có trọng số của entropy sau khi chia tập dữ liệu theo từng thuộc tính của nó. Entropy đo mức độ hỗn loạn thông tin trong dữ liệu và được tính bằng công thức:

$$Entropy(S) = -\sum_{i=1}^{n} P_i \times log_2(P_i)$$

Trong đó, Pi là xác suất của lớp i.

Chỉ số Gini đo lường độ thuần nhất trong quá trình tạo ra cây quyết định. Thuật toán cây quyết định ưu tiên các thuộc tính có chỉ số Gini nhỏ hơn so với các thuộc tính có chỉ số Gini lớn hơn khi đưa ra quyết định. Việc tính toán chỉ số gini được thực hiện theo công thức sau:

$$Gini(S)\ =\ 1\ -\ \sum pj^2$$

Trong đó, pj là tỉ lệ phần tử của lớp j trong tập dữ liệu S.

Thuật toán cây quyết định mang lại nhiều ưu điểm nổi bật như: thuật toán đơn giản, dễ hiểu và không cần kiến thức quá chuyên sâu. Bên cạnh đó cây quyết định cũng hữu ích trong giai đoạn khám phá dữ liệu (data exploration), một ưu điểm nữa là nó có yêu cầu bước làm sạch dữ liệu ít bước hơn và không bị ảnh hưởng bởi các giá trị thiếu. Nó có thể xử lý linh hoạt các biến số cũng như các biến có bản chất phân loại. Tuy nhiên cây quyết định cũng có một số nhược điểm như: vấn đề quá khớp overfitting, mô hình học quá chi tiết từ dữ liệu huấn luyện, tuy nhiên, bằng cách cắt tỉa và thiết lập các

ràng buộc tham số mô hình, các vấn đề quá khớp có thể được giảm bớt. Ngoài ra, cây quyết định cũng không phù hợp với các biến liên tục, do quá trình phân chia có thể làm mất đi một phần thông tin giá trị ban đầu, dẫn đến suy giảm hiệu quả của mô hình.

2.4. Rừng ngẫu nhiên

Thuật toán rừng ngẫu nhiên (Random Forest - RF) là: một kỹ thuật học máy thuộc nhóm học tổ hợp (ensemble learning), được ứng dụng không chỉ trong các bài toán phân loại mà còn trong các bài toán hồi quy. Khái niệm "ensemble" đề cập đến việc kết hợp nhiều mô hình khác nhau nhằm đưa ra kết quả đầu ra tổng hợp sau khi đã huấn luyện xong. Về bản chất, Random Forest được xem là một dạng mở rộng của kỹ thuật Bootstrap Aggregating (Bagging). Thuật toán sử dụng nhiều cây quyết định (Decision Trees), trong đó mỗi cây được xây dựng độc lập với nhau [6].

Trong quá trình huấn luyện, một tập dữ liệu con được chọn ngẫu nhiên từ tập dữ liệu ban đầu để huấn luyện riêng cho từng cây quyết định, theo phương pháp lấy mẫu bootstrap (lấy mẫu có hoàn lại). Mỗi cây sau đó được đánh giá độ chính xác độc lập, và kết quả cuối cùng được xác định bằng cách lấy trung bình (đối với hồi quy) hoặc theo nguyên tắc số đông (majority vote) trong phân loại. Việc huấn luyện các cây quyết định riêng lẻ giúp mô hình đạt độ chệch thấp (low bias), vì mỗi cây học rất tốt trên tập huấn luyện nhỏ. Tuy nhiên, các cây đơn lẻ có thể dễ bị dao động (high variance) khi gặp dữ liệu mới.

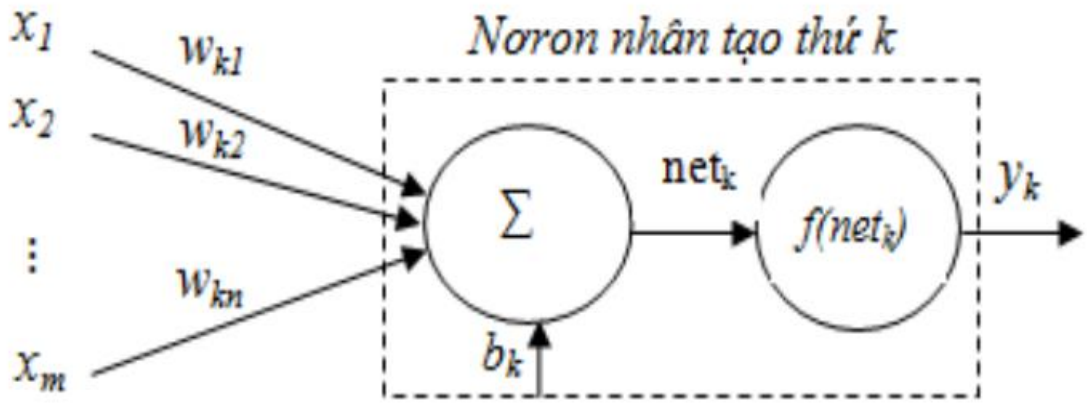
Thuật toán rừng ngẫu nhiên sử dụng nhiều cây để giảm nguy cơ quá khớp (overfitting). Ngoài ra, thuật toán này cần ít thời gian hơn trong giai đoạn huấn luyện. Đối với các tập dữ liệu lớn, thuật toán này hoạt động hiệu quả. Nó tạo ra các dự đoán có độ chính xác cao. Tuy nhiên, ngay cả khi một lượng lớn dữ liệu bị thiếu, thuật toán này vẫn có thể duy trì độ chính xác. Tuy nhiên, kết quả sinh ra rừng ngẫu nhiên lại khó diễn giải hơn so với mô hình cây quyết định, sử dụng nhiều tài nguyên tính toán và bộ nhớ do phải xây dựng và lưu trữ nhiều cây.

2.5. Mạng nơ-ron nhân tạo

Mạng nơ-ron nhân tạo (Artificial Neural Network - ANN - Hình 1), là một mô hình toán học được xây dựng dựa trên mô phỏng lại mạng nơ-ron sinh học, gồm một nhóm các nơ-ron nhân tạo nối với nhau và xử lý thông tin bằng cách truyền theo các kết nối và tính giá trị mới tại các nút kế tiếp. ANN học được kiến thức thông qua quá trình huấn luyện, dựa vào kiến thức được đào tạo để cải thiện khả năng học của mình và sử dụng kiến thức đó cho việt dự đoán kết quả của dữ liệu chưa biết [7].

Kiến trúc chung của mạng nơ-ron nhân tạo (ANN) gồm 3 thành phần đó là : tầng đầu vào (input layer), tầng ẩn (hidden layer) và tầng đầu ra (output layer), có thể có một hay nhiều tầng ẩn tuỳ vào bài toán.

Cấu trúc chung của một nơ-ron nhân tạo có các thành phần cơ bản mô tả ở hình 2.5 dưới đây.Trong ANN, thông tin được truyền qua các kết nối của nơ-ron hay là một nút (node), nó thực hiện công việc nhận tín hiệu từ các đơn vị phía trước hoặc bên ngoài sửa dụng chúng để tín hiệu ra lan truyền đến các đơn vị khác. Đầu vào cung cấp các tín hiệu vào (input signals) của nơ-ron, các tín hiệu này thường được đưa vào dưới dạng một vector N chiều.



Hình 2.5: Hình nơ-ron nhân tạo thứ k [8]

Các liên kết: mỗi liên kết thể hiện bằng một trọng số (gọi là trọng số liên kết – Synaptic weight). Trọng số liên kết giữa tín hiệu vào thứ i với nơ ron k thường được ký hiệu w_{ki}. Thông thường trọng số được khởi tạo ngẫu nhiên và được cập nhật liên tục trong quá trình huấn luyện.

Hàm kết hợp (combination function): mỗi một đơn vị trong một mạng kết hợp các giá trị đưa vào nó thông qua các liên kết với các đơn vị khác, sinh ra một giá trị net input. Hàm thực hiện nhiệm vụ này gọi là hàm kết hợp (combination function), được định nghĩa bởi một luật lan truyền cụ thể. Thường dùng để tính tổn của tích các đầu vào với trọng số liên kết của nó.

$$net_k = \sum_{i=1}^{n} w_{ki} x_i + b_k$$

Trong đó, b_k là ngưỡng (còn gọi là độ lệch bias): ngưỡng này thường được đưa vào như một thành phần của hàm truyền.

Tiếp theo, nó sẽ được xử lý bởi hàm truyền (tranfer function) để chuyển thành tín hiệu đầu ra. Hàm này dùng để giới hạn phạm vi đầu ra của mỗi nơ-ron . Nó nhận đầu vào là kết quả của hàm kết hợp và ngưỡng đã cho. Thường thì phạm vi của mỗi nơ-ron được giới hạn trong đoạn [0,1] hoặc [-1,1], tuỳ thuộc vào hàm truyền (hay còn tên gọi khác hàm kích hoạt) . Các hàm truyền có thể là các hàm tuyến tính hoặc phi tuyến tính . Một số hàm truyền thường được sử dụng như: hàm đồng nhất, hàm bước nhị phân, hàm sigmoid, hàm sigmoid lưỡng cực.

Đầu ra của nơ-ron là output ký hiệu $y_k = f(net_k)$, là kết quả đầu ra của một nơ-ron giải pháp của vấn đề, với mỗi nơ-ron sẽ có tối đa là một đầu ra, là kết quả của hàm truyền.

2.6. AdaBoost

AdaBoost (Adaptive Boosting) Là thuật toán học máy dùng trong bài toán phân lớp và là một những thuật toán tập hợp mô hình phổ biến nhất. AdaBoost hoạt động bằng cách xây dựng các mô hình phân loại yếu tuần tự. Thông qua việc duy trì và điều chỉnh một tập trọng số trên dữ liệu huấn luyện. Cụ thể, sau mỗi vòng huấn luyện, thuật toán sẽ tăng trọng số của các mẫu bị phân loại sai và giảm trọng số của các mẫu được phân loại đúng [9]. Cách tiếp cận thích nghi này buộc mô hình học yếu kế tiếp phải tập trung nhiều hơn vào các mẫu khó phân loại trong tập huấn luyện. Kết quả cuối cùng của AdaBoost là sự kết hợp có trọng số của tất cả các mô hình yếu, trong đó mỗi mô hình sẽ được gán trọng số dựa trên độ chính xác của nó trong quá trình huấn luyện.

Tuy nhiên, AdaBoost cũng tồn tại một số nhược điểm. Do quá trình huấn luyện diễn ra tuần tự và phụ thuộc vào kết quả của các vòng trước, thuật toán có thể yêu cầu thời gian huấn luyện dài hơn. Ngoài ra, AdaBoost khá nhạy cảm với nhiễu (noise) trong dữ liệu: nếu dữ liệu có chứa nhiều điểm ngoại lệ, mô hình có thể bị sai lệch do tập trung quá mức vào các điểm khó phân loại.

2.7. Bayes thơ ngây

Bayes thơ ngây (Native Bayes – NB): là một thuật toán được áp dụng rộng rãi cho bài toán phân lớp dựa trên nguyên lý của định lý Bayes với giả định "Ngây thơ" rằng các thuộc tính đều có độ quan trọng như nhau, các thuộc tính đều độc lập thống kê. Naive Bayes dựa trên công thức Bayes để tính xác suất của một lớp dựa trên các đặc trưng của dữ liệu. Đầu tiên, là giai đoạn học, huấn luyện mô hình (learning phase): xây dựng mô hình sẵn dùng (tính sẵn xác suất xuất hiện của tất cả các trường hợp). Tiếp theo là, dự đoán Khi có đối tượng,sự kiện mới xuất hiện cần phân loại: xác định nhãn của đối tượng mới đến thông qua giá trị xác suất lớn nhất tính được.

Định lý Bayes cung cấp công cụ để tính xác suất hậu nghiệm P(h|D) dựa trên xác suất tiên nghiệm P(h), cùng với xác suất quan sát P(D) và xác suất có điều kiện P(D|h) (Mitchell, 1997) được tính theo công thức sau:

$$P(h|D) = \frac{P\ (D|h)\ \times P(h)}{P(D)}$$

Trong đó,

- P(h|D) gọi là posterior probability: xác suất của mục tiêu h với điều kiện có đặc trưng D.
- P(D|h) gọi là likelihood: xác suất của đặc trưng D khi đã biết mục tiêu h.
- P(h) gọi là prior probability của mục tiêu h.
- P(D) gọi là prior probability của đặc trưng D.

Ngoài ra, Naive Bayes có nhiều biến thể để phù hợp với các loại dữ liệu khác nhau. Naive Bayes Gaussian được sử dụng khi các đặc trưng có phân phối chuẩn (Gaussian); Naive Bayes Multinomial thường áp dụng cho dữ liệu đếm, rất thích hợp trong phân loại văn bản như lọc thư rác; Naive Bayes Bernoulli phù hợp cho dữ liệu nhị phân; và Naive Bayes Categorical lại tỏ ra hiệu quả trong các tác vụ phân loại văn bản với đặc trưng dạng rời rạc. Tuy nhiên, hạn chế lớn nhất của Naive Bayes chính là giả định về tính độc lập giữa các đặc trưng, điều này không phải lúc nào cũng đúng trong thực tế, và có thể làm giảm độ chính xác của mô hình khi các đặc trưng thực sự có liên quan đến nhau.

CHƯƠNG 3: HUẤN LUYỆN VÀ ĐÁNH GIÁ

1. Mô hình phân lớp dựa trên tập dữ liệu văn bản

Dữ liệu văn bản được xử lý và trích xuất đặc trưng thông qua một chuỗi các bước bao gồm: token hóa (tách từ), loại bỏ dấu câu, chuyển đổi toàn bộ văn bản thành chữ thường, làm sạch văn bản và cuối cùng là chuyển đổi thành vector đặc trưng bằng phương pháp TF-IDF.

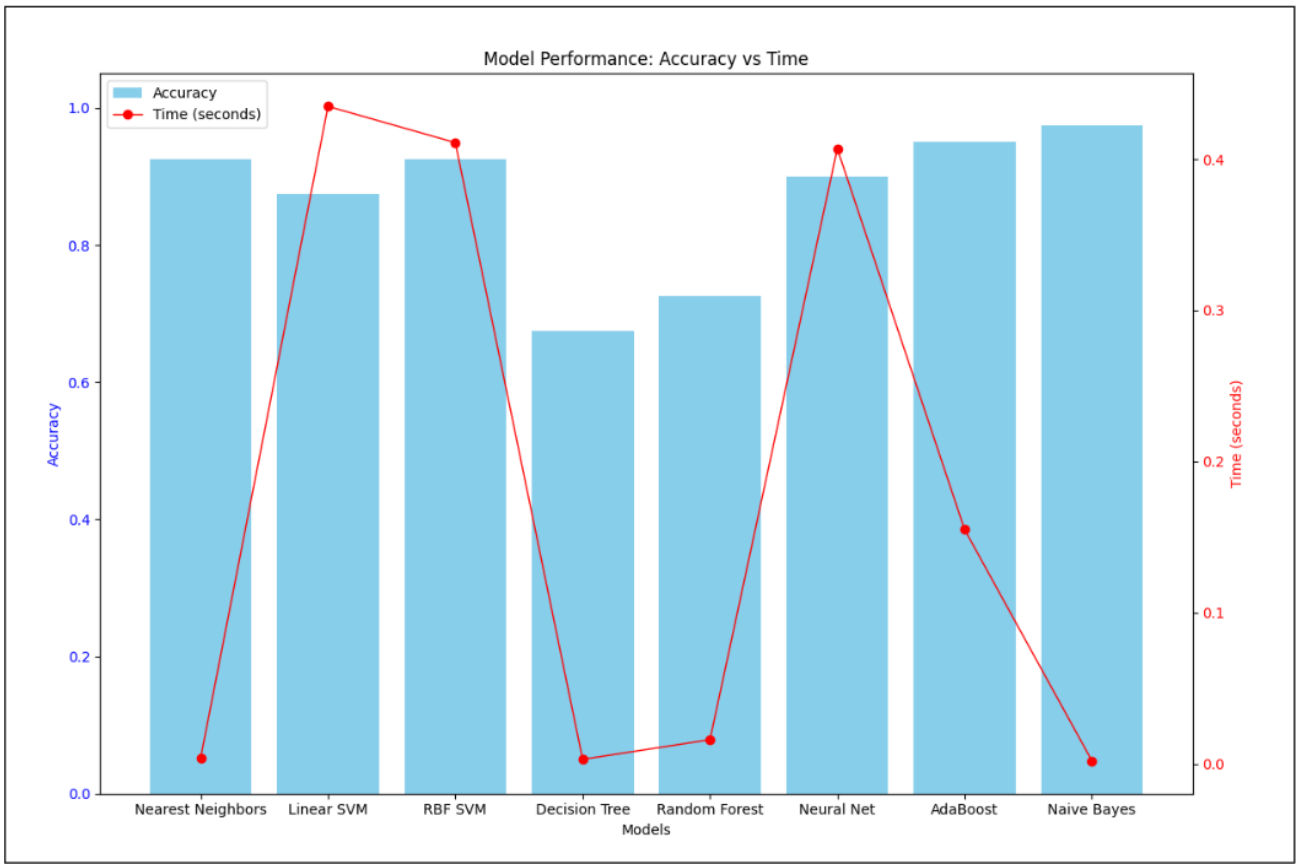
TF-IDF (Term Frequency - Inverse Document Frequency) là một phương pháp phổ biến trong lĩnh vực xử lý ngôn ngữ tự nhiên, dùng để xác định mức độ quan trọng của một từ trong một văn bản, xét trong mối tương quan với toàn bộ tập hợp tài liệu. Phương pháp này giúp giảm thiểu ảnh hưởng của những từ xuất hiện phổ biến nhưng không mang nhiều giá trị phân biệt có thể làm nhiễu dữ liệu.

Sau khi huấn luyện và đánh giá mô hình trên tập dữ liệu kiểm tra (Test set), ta thu được kết quả như trong bảng dưới đây:

Bảng 3.1 Độ chính xác và thời gian của các mô hình trong tập dữ liệu văn bản

STT	Model	Accuracy	Time (seconds)
1	K-Nearest Neighbors	0.925	0.004
2	Linear SVM	0.875	0.435
3	RBF SVM	0.925	0.411
4	Decision Tree	0.675	0.003
5	Random Forest	0.725	0.016
6	Neural Net	0.9	0.407
7	AdaBoost	0.95	0.155
8	**Naive Bayes**	**0.975**	**0.002**

Độ chính xác và thời gian huấn luyện với kiểm thử của các mô hình trên tập dữ liệu văn bản được trình bày chi tiết trong Bảng 3.1. Để tiện cho việc so sánh hiệu quả giữa các mô hình, hình minh họa 3.1 dưới đây sẽ cung cấp cái nhìn trực quan hơn về độ chính xác cũng như thời gian xử lý tương ứng của từng mô hình.



Hình 3.1 Độ chính xác và thời gian của các mô hình trong tập dữ liệu văn bản

Về độ chính xác:

- Kết quả thu được cho thấy, mô hình Naive Bayes đạt hiệu suất cao nhất, với độ chính xác lên tới 97.5%, khẳng định sự phù hợp vượt trội của nó trong bài toán phân loại văn bản. Theo sau là AdaBoost (95%) và hai mô hình K-Nearest Neighbors và RBF SVM (92.5%), đều cho thấy khả năng phân loại tốt.
- Ngược lại, các mô hình như Decision Tree và Random Forest lại cho độ chính xác khá khiêm tốn (lần lượt là 67.5% và 72.5%), cho thấy chúng không tối ưu trong việc xử lý loại dữ liệu này.

Về thời gian xử lý:

- Naive Bayes tiếp tục thể hiện ưu thế vượt trội khi chỉ mất 0.002 giây để thực thi toàn bộ quá trình. Decision Tree cũng rất nhanh, chỉ mất 0.003 giây.
- Trong khi đó, các mô hình phức tạp như Linear SVM, RBF SVM và Neural Net có thời gian xử lý dài hơn đáng kể (lần lượt là 0.435, 0.411 và 0.407 giây),

do đặc điểm thuật toán và quá trình tối ưu tốn kém. Mặc dù vẫn đạt độ chính xác cao.

Mô hình đạt hiệu quả nhất: Ta có thể thấy Naive Bayes nổi bật khi vừa đạt độ chính xác cao nhất, vừa có tốc độ thời gian chạy nhanh nhất.Với độ chính xác 97.5%, đây là một lợi thế lớn trong phân loại văn bản nhanh và chính xác, đặc biệt với những hệ thống yêu cầu đọ chính xác cao, phản hồi thời gian thực hoặc tài nguyên hạn chế.

Nguyên nhân khiến Naive Bayes phù hợp với phân loại văn bản:

- Đơn giản trong cấu trúc và thuật toán: Naive Bayes dựa trên định lý Bayes và các xác suất có điều kiện. Mô hình này chủ yếu dựa vào việc tính xác suất có điều kiện giữa các đặc trưng và nhãn, giúp giảm thiểu độ phức tạp và tăng tốc độ xử lý.
- Phù hợp với đặc trưng văn bản dạng rời rạc: Mô hình giả định rằng các đặc trưng (như các từ trong văn bản) là độc lập với nhau. Tuy giả định này không hoàn toàn đúng trong thực tế, nhưng với dữ liệu TF-IDF hoặc các vector biểu diễn văn bản, nó vẫn hoạt động rất hiệu quả.
- Khả năng thích ứng tốt với dữ liệu có số chiều cao: Mô hình không gặp vấn đề với các biểu diễn văn bản có số lượng đặc trưng lớn (như hàng ngàn từ), điều mà nhiều mô hình khác phải đối mặt.
- Khả năng tổng quát hoá và ổn định trong môi trường có nhiễu: Naive Bayes thường ít bị ảnh hưởng bởi dữ liệu ngoại lệ hoặc nhiễu, nhờ phương pháp tính toán xác suất tổng quát thay vì phụ thuộc vào từng mẫu cụ thể.

2. Mô hình phân lớp dựa trên tập dữ liệu hình ảnh

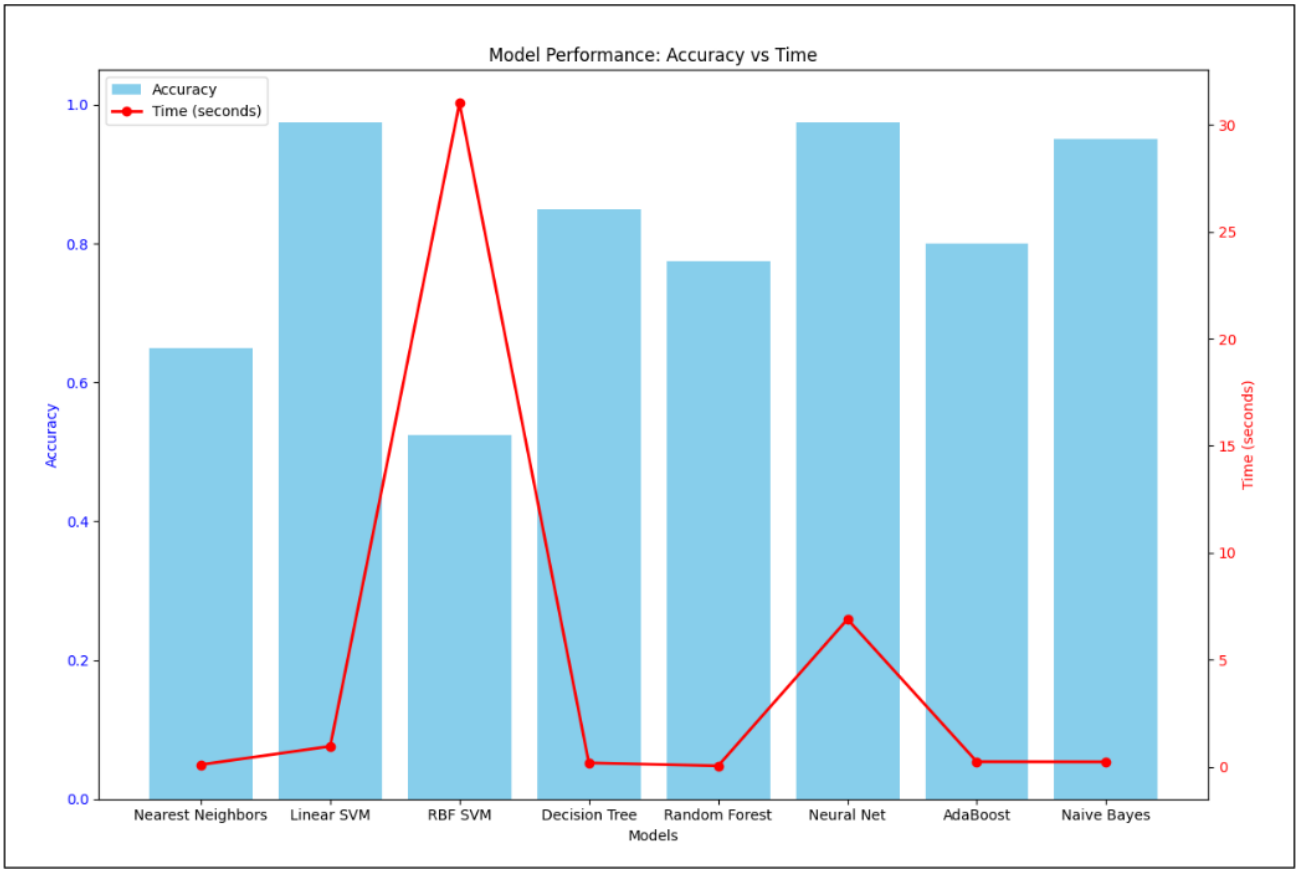
2.1. Tập dữ liệu hình ảnh được trích xuất bằng đặc trưng GIST

Tập dữ liệu hình ảnh lúc này sẽ được trích đặc trưng bằng phương pháp GIST – một kỹ thuật mô tả ảnh tổng quát nhằm tóm tắt bố cục không gian của toàn bộ hình ảnh. GIST không tập trung vào chi tiết từng đối tượng riêng lẻ mà hướng đến việc nắm bắt thông tin cảnh tổng thể, như cấu trúc, hướng cạnh, và độ tương phản ở các vùng khác nhau của ảnh. GIST được xây dựng dựa trên việc áp dụng một tập các bộ lọc Gabor lên các vùng lưới phân chia trong ảnh, từ đó thu được một vector đặc trưng có kích thước cố định đại diện cho ảnh đầu vào. Kết quả được thể hiện dưới đây:

Bảng 3.2 Độ chính xác và thời gian của các mô hình trong tập dữ liệu hình ảnh (GIST)

STT	Model	Accuracy	Time (seconds)
1	K-Nearest Neighbors	0.650	0.096
2	**Linear SVM**	**0.975**	**0.954**
3	RBF SVM	0.525	31.019
4	Decision Tree	0.850	0.179
5	Random Forest	0.775	0.041
6	Neural Net	0.975	6.894
7	AdaBoost	0.800	0.233
8	Naive Bayes	0.950	0.226

Độ chính xác và thời gian huấn luyện với kiểm thử của các mô hình trên tập dữ liệu hình ảnh được trích xuất bằng đặt trưng GIST, được trình bày chi tiết trong Bảng 3.2. Để tiện cho việc so sánh hiệu quả giữa các mô hình, ảnh minh họa 3.2 dưới đây sẽ cung cấp cái nhìn trực quan hơn về độ chính xác cũng như thời gian xử lý tương ứng của từng mô hình.



Hình 3.2 Độ chính xác và thời gian của các mô hình trong tập dữ liệu hình ảnh (GIST)

Về độ chính xác:

- Mô hình Linear SVM và Neural Net đều đạt độ chính xác cao nhất 97.5%, cho thấy đây là hai mô hình rất phù hợp khi làm việc với đặc trưng hình ảnh của GIST, cho thấy mô hình rất hiệu quả trong việc phân loại dữ liệu ảnh.
- Naive Bayes cũng cho kết quả tốt 95%, mặc dù mô hình này thường phù hợp với đặc trưng rời rạc nhưng vẫn hoạt động hiệu quả trên GIST, cho thấy khả năng phân loại tốt của mô hình.
- Các mô hình như Decision Tree 85%, AdaBoost 80%, Random Forest 77,5%, có độ chính xác cũng khá cao, chấp nhận được nhưng không quá nổi bật.
- Mô hình KNN chỉ đạt độ chính xác 65%, cho thấy chưa tận dụng tốt được đặc trưng GIST trong việc phân biệt lớp.

- RBF SVM có độ chính xác thấp nhất 52.5%, mặc dù lý thuyết có thể xử lý tốt đặc trưng phi tuyến, nhưng thực tế cho thấy chưa tối ưu. Cho thấy mô hình này không phù hợp với bài toán này khi áp dụng trích xuất bằng GIST.

Về thời xử lý:

- Random Forest và KNN là hai mô hình có thời gian xử lý nhanh nhất (0.041s và 0.096s). Đây là lợi thế nếu yêu cầu thời gian phản hồi nhanh, đặc biệt với dữ liệu lớn, cho thấy khả năng xử lý nhanh.
- Các mô hình đơn giản khác như Decision Tree (0.179s), Naive Bayes (0.226s) và AdaBoost (0.233s) cũng có thời xử lý rất tốt, phù hợp trong môi trường tài nguyên hạn chế.
- Linear SVM và Neural Net yêu cầu nhiều thời gian hơn các thuật toán trên (0.954s và 6.894s). Điều này phản ánh bản chất thuật toán có quá trình tối ưu và lan truyền ngược nhiều bước, dẫn đến chi phí tính toán cao hơn.
- RBF SVM có thời gian thực thi cao nhất (31.019s), vượt xa các mô hình còn lại. Với độ chính xác không tốt, mô hình này không phải lựa chọn hợp lý trên tập dữ liệu GIST.

Mô hình đạt hiệu quả cao nhất: Linear SVM là mô hình nổi bật nhất khi áp dụng trên tập dữ liệu hình ảnh sử dụng đặc trưng GIST. Mặc dù thời gian xử lý của nó không phải là thấp nhất (0.954 giây), nhưng độ chính xác đạt mức tuyệt đối 97.5% đã giúp mô hình này thể hiện rõ tính hiệu quả và ổn định khi làm việc với dữ liệu ảnh khi sử dụng GIST.

Nguyên nhân khiến Linear SVM phù hợp với trường hợp này là:

- Xử lý tốt dữ liệu có ranh giới rõ ràng: Khi đặc trưng GIST được trích xuất tốt, đã giúp biểu diễn ảnh một cách tổng quát và giúp các lớp ảnh tách biệt tương đối rõ, Linear SVM có thể tìm được siêu phẳng tốt nhất để phân tách chính xác giữa các lớp, từ đó đạt độ chính xác cao.
- Khả năng tổng quát hóa tốt: Với đặc trưng đầu vào phù hợp, Linear SVM tập trung tối ưu hóa biên độ an toàn giữa các lớp, nên dù dữ liệu ảnh đa dạng nên có thể tránh được tình trạng quá khớp, giúp cải thiện hiệu quả phân loại trong các tập dữ liệu ảnh đa dạng.
- Tốc độ xử lý tốt: Dù không nhanh bằng các mô hình đơn giản như Naive Bayes (0.226s) hay Decision Tree (0.179s), nhưng Linear SVM vẫn giữ được thời gian xử lý dưới 1 giây, vẫn phù hợp với các hệ thống không yêu cầu thời gian thực quá nghiêm ngặt.

- Ít phụ thuộc vào tham số: Không giống như mạng nơ-ron cần điều chỉnh nhiều siêu tham số, Linear SVM có cấu trúc đơn giản hơn ít tham số hơn và dễ kiểm soát, giúp tiết kiệm thời gian quá trình huấn luyện.

2.2. Tập dữ liệu hình ảnh được trích xuất bằng đặc trưng VGG19

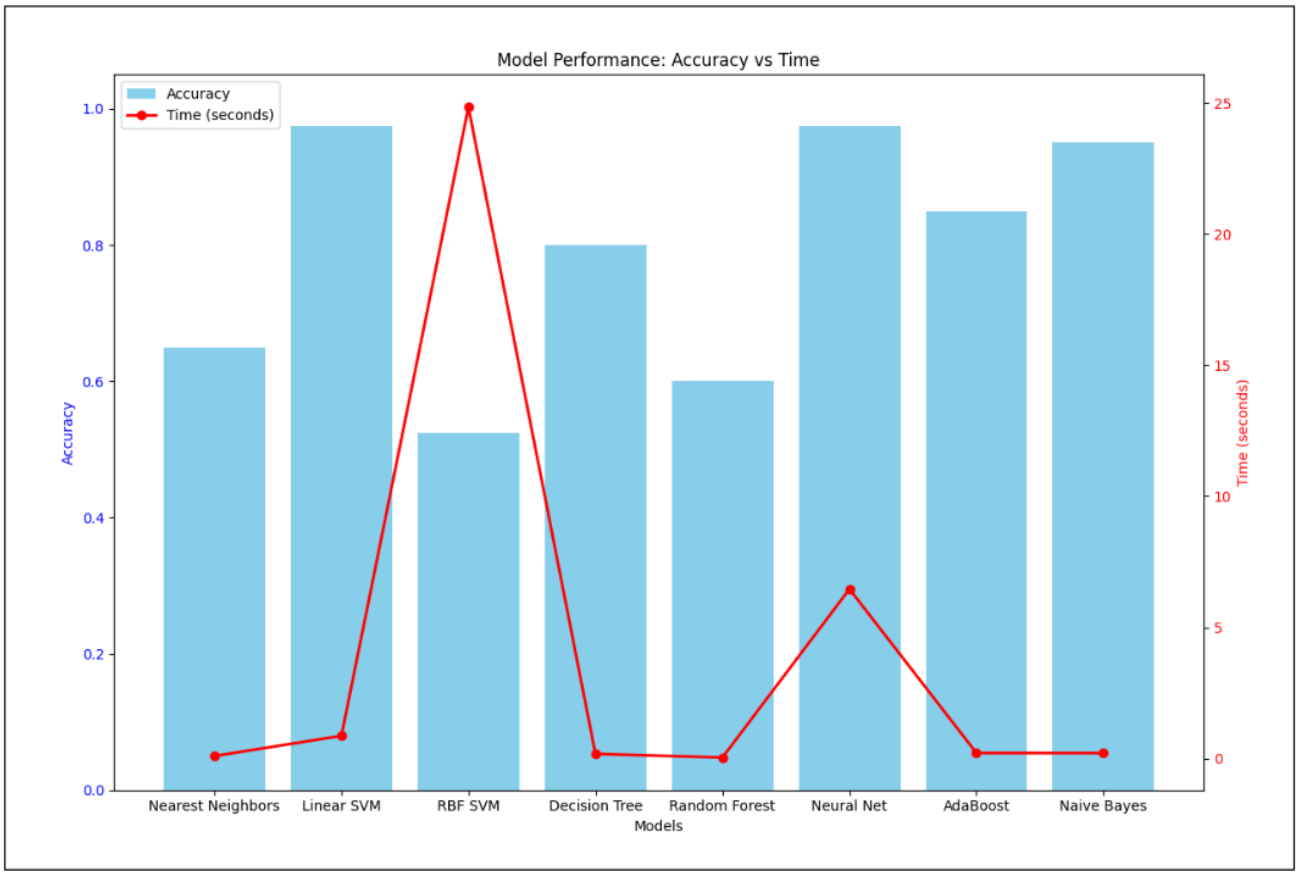
Trong giai đoạn này, tập dữ liệu hình ảnh được chuyển đổi thành dạng đặc trưng thông qua mạng nơ-ron tích chập VGG19. Đây là một mô hình mạng nơ-ron sâu được phát triển bởi nhóm Visual Geometry Group (VGG) tại Đại học Oxford, nổi bật với kiến trúc đơn giản nhưng hiệu quả. VGG19 gồm 19 lớp với các khối tích chập nhỏ (3x3) xếp chồng lên nhau, cho phép mô hình học được các đặc trưng thị giác phức tạp ở nhiều cấp độ.

Mục tiêu của quá trình trích đặc trưng là rút gọn thông tin hình ảnh đầu vào thành các vector đặc trưng giàu ngữ nghĩa, từ đó có thể sử dụng làm đầu vào cho các mô hình học máy truyền thống để thực hiện phân loại. Với khả năng học sâu, VGG19 đã chứng minh hiệu quả trong việc tạo ra các biểu diễn đặc trưng chất lượng cao từ hình ảnh.

Bảng 3.3 Độ chính xác và thời gian của các mô hình trong tập dữ liệu hình ảnh (VGG19)

STT	Model	Accuracy	Time (seconds)
1	K-Nearest Neighbors	0.650	0.095
2	**Linear SVM**	**0.975**	**0.865**
3	RBF SVM	0.525	24.836
4	Decision Tree	0.800	0.177
5	Random Forest	0.600	0.037
6	Neural Net	0.975	6.462
7	AdaBoost	0.850	0.210
8	Naive Bayes	0.950	0.206

Độ chính xác và thời gian huấn luyện với kiểm thử của các mô hình trên tập dữ liệu hình ảnh được trích xuất bằng đặt trưng VGG19, được trình bày chi tiết trong Bảng 3.3. Để tiện cho việc so sánh hiệu quả giữa các mô hình, ảnh minh họa 3.3 dưới đây sẽ cung cấp cái nhìn trực quan hơn về độ chính xác cũng như thời gian xử lý tương ứng của từng mô hình.



Hình 3. 3 Độ chính xác và thời gian của các mô hình trong tập dữ liệu hình ảnh (VGG19)

Về độ chính xác:

- Linear SVM và Neural Net đạt độ chính xác cao nhất 97.5%, cho thấy cả hai mô hình này khai thác đặc trưng trích xuất từ VGG19 rất hiệu quả. Đây là hai mô hình thường thể hiện tốt trên các dữ liệu hình ảnh phức tạp nhờ khả năng xử lý phi tuyến mạnh.
- Naive Bayes và AdaBoost cũng cho kết quả tốt, lần lượt đạt 95% và 85%. Điều này cho thấy VGG19 đã tạo ra đặc trưng đủ mạnh để các mô hình đạt hiệu quả tốt.
- Decision Tree đạt 80% kết quả tương đối ổn, tuy nhiên không vượt trội có thể do khó xử lý đặc trưng phức tạp từ VGG19.
- Ngược lại, Random Forest và KNN cho độ chính xác thấp 60% và 65%, trong khi RBF SVM đạt kết quả kém nhất với chỉ 52,5%. Điều này cho thấy các mô hình này có thể chưa thích nghi tốt với vector đặc trưng phức tạp và chiều cao từ VGG19. Cho thấy mô hình không phù hợp với bài toán này.

Về thời gian xử lý:

- Random Forest có thời gian xử lý nhanh nhất là 0,037s. KNN và Decision Tree có thời xử lý nhanh lần lượt là 0.095s và 0.177s, nhờ cấu trúc đơn giản và khả năng huấn luyện nhanh với vector đặc trưng cố định.
- Naive Bayes và AdaBoost có thời gian thực thi lần lượt là 0.206s và 0.21s giây, vẫn được xem là nhanh, hợp lý và phù hợp trong môi trường tài nguyên hạn chế.
- Linear SVM và đặc biệt là Neural Net có thời gian xử lý dài hơn, lần lượt là 0.865s và 6.462s giây cần nhiều thời gian hơn do quá trình tối ưu hóa phức tạp.
- RBF SVM có thời gian xử lý lâu nhất (24.836 giây), thời gian quá lớn so với độ chính xác đạt được, khiến nó trở thành lựa chọn kém phù hợp trong tình huống này.Do đó không phù hợp với dạng đặc trưng này.

Linear SVM và Neural Net là hai mô hình nổi bật khi xét về độ chính xác. Tuy nhiên, xét trên tổng thể giữa độ chính xác và tốc độ, Linear SVM là lựa chọn hợp lý nhất: vừa đạt 97.5% chính xác, vừa giữ được thời gian xử lý tốt 0.865s. Trong khi đó, mặc dù Neural Net cũng đạt độ chính xác tương đương, nhưng thời gian chạy dài hơn tới 6.462s.

Nguyên nhân khiến Linear SVM phù hợp với trường hợp này là:

- Khả năng phân tách tuyến tính hiệu quả: Linear SVM được thiết kế để tối ưu hóa siêu phẳng phân tách giữa các lớp. Đặc trưng do VGG19 có cấu trúc rõ ràng trong không gian vector, giúp mô hình dễ dàng tìm được siêu phẳng phân lớp chính xác, từ đó đạt được độ chính xác cao.
- Tối ưu hóa hiệu quả: Khác với các mô hình phi tuyến như RBF SVM hoặc các mạng nơ-ron, Linear SVM có thuật toán huấn luyện ít phức tạp hơn và thường hội tụ nhanh. Điều này giúp giảm thời gian xử lý đáng kể trong khi vẫn duy trì được hiệu quả phân loại.
- Hạn chế overfitting: Bằng cách tối đa hóa biên phân cách giữa các lớp, Linear SVM có khả năng tổng quát hóa tốt, đặc biệt trong trường hợp số lượng mẫu huấn luyện không quá lớn.

2.3. Tập dữ liệu hình ảnh được trích xuất bằng đặc trưng Color+Hog+Gist

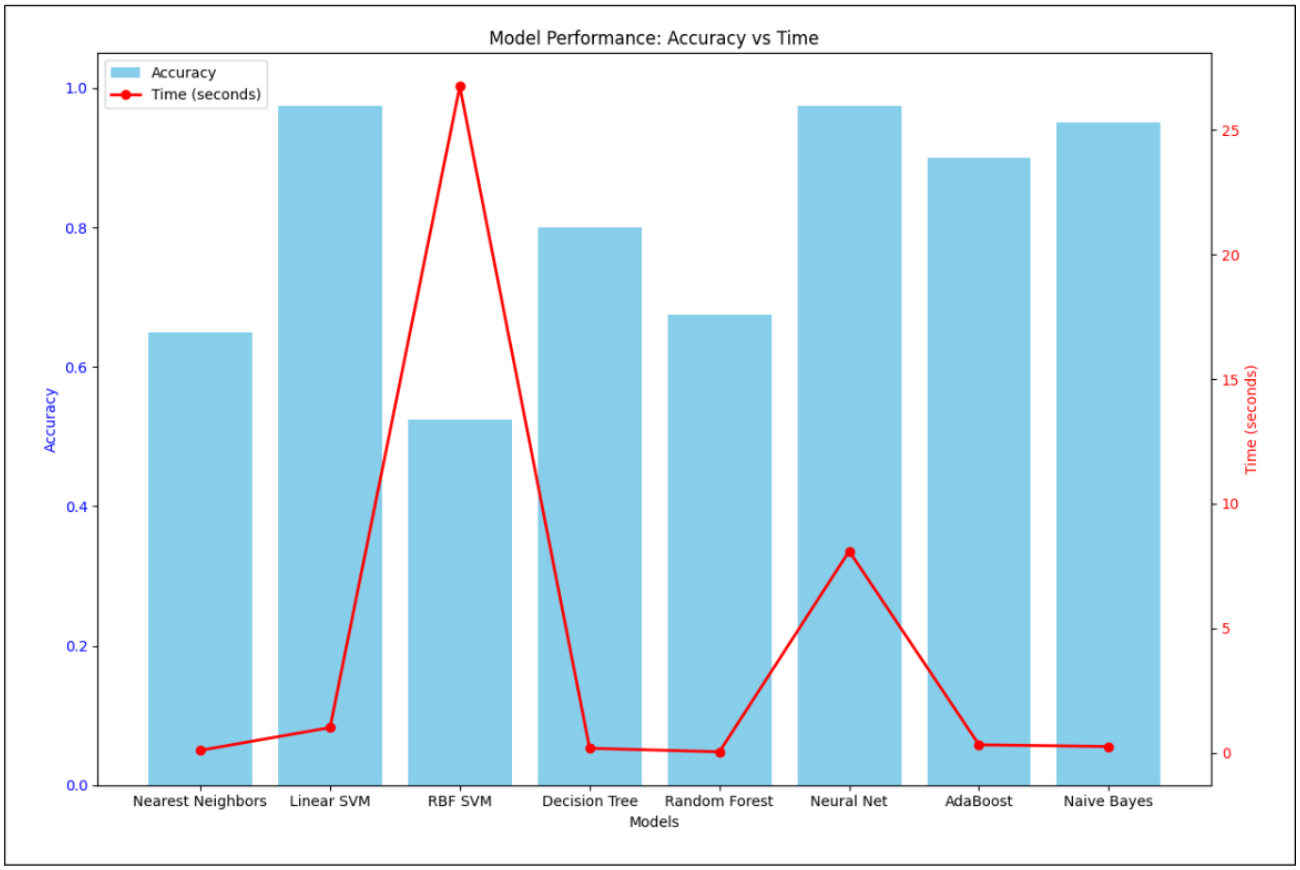
Trong giai đoạn này, tập dữ liệu hình ảnh được biểu diễn thông qua việc kết hợp ba loại đặc trưng thủ công phổ biến: Color histogram, Histogram of Oriented Gradients (HOG), và Gist descriptor. Đây là các kỹ thuật trích xuất đặc trưng cổ điển được áp dụng rộng rãi trong

thị giác máy tính trước thời kỳ của học sâu. Sự kết hợp giữa ba loại đặc trưng này nhằm tận dụng các thông tin khác nhau từ ảnh: Color thể hiện thông tin màu sắc, HOG nắm bắt cấu trúc cạnh và hình dạng, còn Gist mô tả tổng thể bố cục không gian của ảnh.

Bảng 3. 4 Độ chính xác và thời gian của các mô hình trong tập dữ liệu hình ảnh (Color+Hog+Gist)

STT	Model	Accuracy	Time (seconds)
1	K-Nearest Neighbors	0.650	0.098
2	**Linear SVM**	**0.975**	**1.013**
3	RBF SVM	0.525	26.751
4	Decision Tree	0.800	0.182
5	Random Forest	0.675	0.038
6	Neural Net	0.975	8.096
7	AdaBoost	0.900	0.322
8	Naive Bayes	0.950	0.253

Độ chính xác và thời gian huấn luyện với kiểm thử của các mô hình trên tập dữ liệu hình ảnh được trích xuất bằng đặt trưng Color+Hog+GIST, được trình bày chi tiết trong Bảng 3.4. Để tiện cho việc so sánh hiệu quả giữa các mô hình, ảnh minh họa 3.4 dưới đây sẽ cung cấp cái nhìn trực quan hơn về độ chính xác cũng như thời gian xử lý tương ứng của từng mô hình.



Hình 3. 4 Độ chính xác và thời gian của các mô hình trong tập dữ liệu hình ảnh (Color+Hog+Gist)

Về độ chính xác:

- Linear SVM và Neural Net tiếp tục dẫn đầu với độ chính xác 97.5%, chứng minh hiệu quả và độ ổn định khi xử lý cả đặc trưng học sâu và thủ công.
- Naive Bayes và AdaBoost đạt độ chính xác cao (95% và 90%), cho thấy các đặc trưng này vẫn đủ mạnh để hỗ trợ các mô hình đơn giản.
- Decision Tree, Random Forest, và KNN đạt mức trung bình (từ 65%–80%), với kết quả cải thiện nhẹ so với khi dùng VGG19, phù hợp hơn với đặc trưng đơn giản.
- RBF SVM tiếp tục cho kết quả thấp nhất (52.5%), cho thấy sự không phù hợp với không gian đặc trưng này.

Về thời gian xử lý:

- Random Forest, KNN, và Decision Tree có thời gian nhanh nhất (dưới 0.2s) nhờ cấu trúc đơn giản.

- Naive Bayes và AdaBoost vẫn duy trì thời gian hợp lý (0.2s–0.3s).
- Linear SVM và Neural Net có thời gian cao hơn (1.013s và 8.096s), nhưng vẫn chấp nhận được trong các ứng dụng thực tế.
- RBF SVM xử lý chậm nhất (26.751s), không tương xứng với độ chính xác thấp.

Tương tự như kết quả với đặc trưng VGG19, Linear SVM và Neural Net vẫn là hai mô hình cho độ chính xác cao nhất (97.5%), chứng minh sự ổn định của chúng khi làm việc với nhiều dạng biểu diễn đặc trưng. Tuy nhiên, xét trên yếu tố cân bằng giữa độ chính xác và thời gian xử lý, Linear SVM tiếp tục là lựa chọn tối ưu, với độ chính xác cao và thời gian xử lý chỉ hơn 1 giây.

Linear SVM phù hợp trong trường hợp này nhờ:

- Khả năng khai thác hiệu quả đặc trưng tuyến tính: Với các đặc trưng được trích xuất rõ ràng như Color, HOG và Gist, Linear SVM dễ dàng tìm được siêu phẳng phân lớp hiệu quả.
- Thuật toán huấn luyện nhanh: Giúp tiết kiệm thời gian xử lý, phù hợp với cả các hệ thống có tài nguyên tính toán giới hạn.
- Khả năng tổng quát hóa cao: Đặc biệt hữu ích khi số lượng mẫu không quá lớn hoặc khi vector đặc trưng không mang tính phi tuyến cao.

2.4. Tập dữ liệu hình ảnh được trích xuất bằng đặc trưng Color+Hog+VGG19

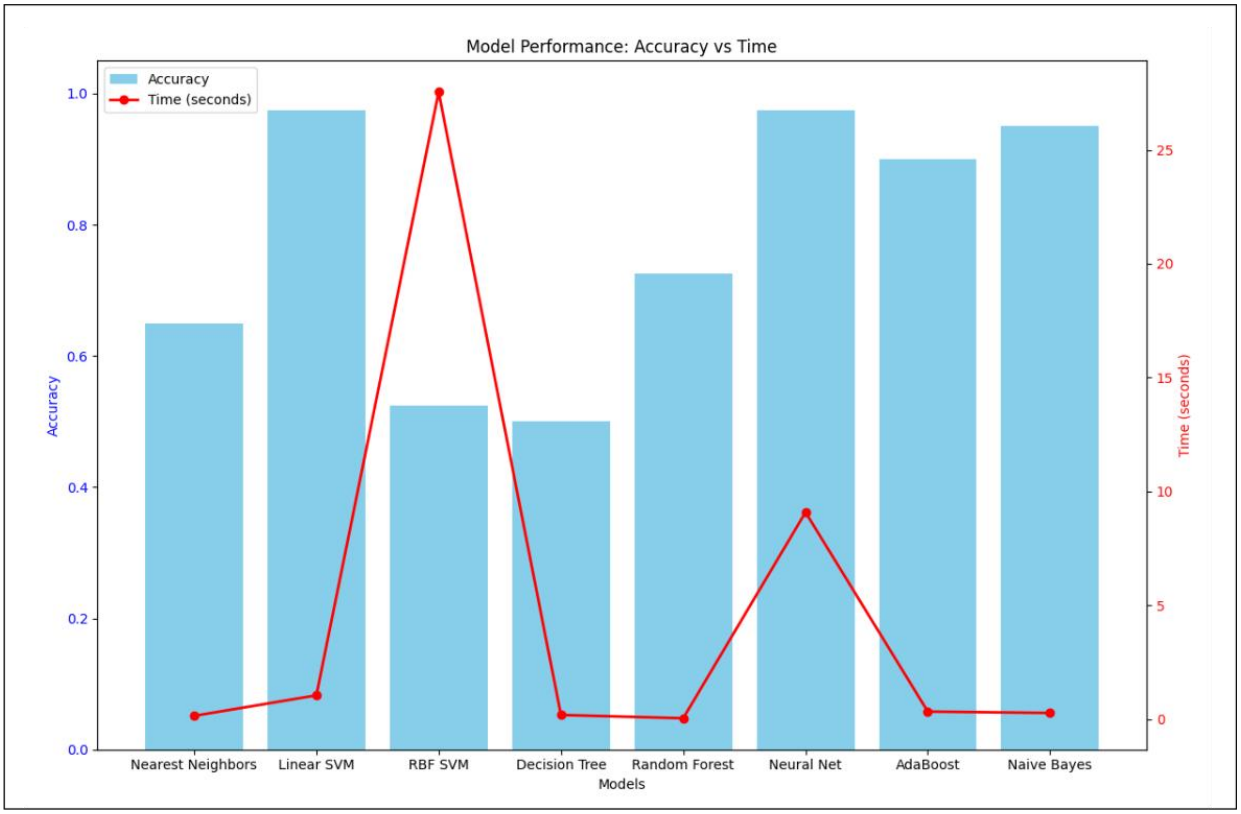
Trong giai đoạn này, tập dữ liệu hình ảnh được biểu diễn thông qua sự kết hợp giữa đặc trưng học sâu VGG19 và các đặc trưng thủ công như Color histogram và HOG. Mục tiêu của việc kết hợp này là tận dụng ưu điểm của cả hai hướng tiếp cận: khả năng học biểu diễn học sâu của VGG19 và tính trực quan, đơn giản nhưng giàu thông tin của Color và HOG.

Bảng 3. 5 Độ chính xác và thời gian của các mô hình trong tập dữ liệu hình ảnh Color+Hog+VGG19)

STT	Model	Accuracy	Time (seconds)
1	K-Nearest Neighbors	0.650	0.140
2	**Linear SVM**	**0.975**	**1.043**
3	RBF SVM	0.525	27.542
4	Decision Tree	0.500	0.180

5	Random Forest	0.725	0.037
6	Neural Net	0.975	9.083
7	AdaBoost	0.900	0.330
8	Naive Bayes	0.950	0.268

Độ chính xác và thời gian huấn luyện cũng như kiểm thử của các mô hình trên tập dữ liệu hình ảnh được trích xuất bằng đặt trưng Color+Hog+GIST, được trình bày chi tiết trong Bảng 3.5. Để tiện cho việc so sánh hiệu quả giữa các mô hình, ảnh minh họa 3.5 dưới đây sẽ cung cấp cái nhìn trực quan hơn về độ chính xác cũng như thời gian xử lý tương ứng của từng mô hình.



Hình 3. 5 Độ chính xác và thời gian của các mô hình trong tập dữ liệu hình ảnh (Color+Hog+VGG19)

Về độ chính xác:

- Linear SVM và Neural Net tiếp tục dẫn đầu với độ chính xác 97.5%, cho thấy cả hai mô hình này đều tận dụng tốt đặc trưng kết hợp.
- Naive Bayes và AdaBoost đạt lần lượt 95% và 90%, tiếp tục cho thấy hiệu quả của đặc trưng trích xuất.
- Random Forest đạt 72.5%, cao hơn so với khi dùng VGG19 đơn lẻ.
- KNN giữ ở mức 65%, trong khi Decision Tree là 50%, cho thấy không phù hợp với không gian đặc trưng phức tạp.
- RBF SVM vẫn thấp nhất với 52.5%, tiếp tục chứng minh là mô hình không phù hợp cho dạng đặc trưng này.

Về thời gian xử lý:

- Random Forest nhanh nhất (0.037s), theo sau là KNN (0.140s) và Decision Tree (0.180s).
- Naive Bayes và AdaBoost có thời gian hợp lý (0.268s và 0.330s).
- Linear SVM và Neural Net mất nhiều thời gian hơn (1.043s và 9.083s) do quá trình huấn luyện phức tạp hơn với dữ liệu chiều cao.
- RBF SVM tiếp tục chậm nhất (27.542s) nhưng độ chính xác lại thấp, không phù hợp với dạng đặc trưng này.

Sự kết hợp giữa đặc trưng học sâu (VGG19) và đặc trưng thủ công (Color + HOG) cho thấy hiệu quả tốt, đặc biệt khi được khai thác bởi các mô hình mạnh như Linear SVM và Neural Net, cả hai đều đạt độ chính xác 97.5%. Tuy nhiên, xét về mặt cân bằng giữa độ chính xác và thời gian xử lý, Linear SVM tiếp tục là lựa chọn hợp lý nhất, với độ chính xác cao và thời gian xử lý chỉ hơn 1 giây.

Linear SVM phù hợp trong trường hợp này nhờ:

- Khả năng phân tách tuyến tính tốt: đặc trưng kết hợp tạo không gian có cấu trúc rõ, thuận lợi cho việc xác định siêu phẳng phân lớp.
- Thuật toán huấn luyện hiệu quả, thời gian xử lý nhanh hơn so với các mô hình phức tạp như mạng nơ-ron.
- Hạn chế overfitting nhờ tối đa hóa biên phân cách, đặc biệt hiệu quả khi số lượng mẫu không quá lớn.

3. Mô hình phân lớp dựa trên tập dữ liệu kết hợp giữa văn bản và hình ảnh

Dữ liệu văn bản được xử lý bằng kỹ thuật TF-IDF sau khi trải qua các bước tiền xử lý như chuẩn hóa, loại bỏ dấu câu và làm sạch. Đối với dữ liệu hình ảnh, đặc trưng được trích xuất thông qua mô hình VGG19, do chạy mô hình với độ chính xác cao và với thời gian chấp nhận được. Các mô hình được huấn luyện riêng biệt trên từng loại dữ liệu để đánh giá độ chính xác. Cuối cùng, kết quả phân loại từ hai mô hình được kết hợp nhằm tạo ra đầu ra phân loại tổng hợp, nâng cao độ chính xác tổng thể.

3.1. Phân lớp dữ liệu kết hợp giữa văn bản và hình ảnh với alpha = 0.4

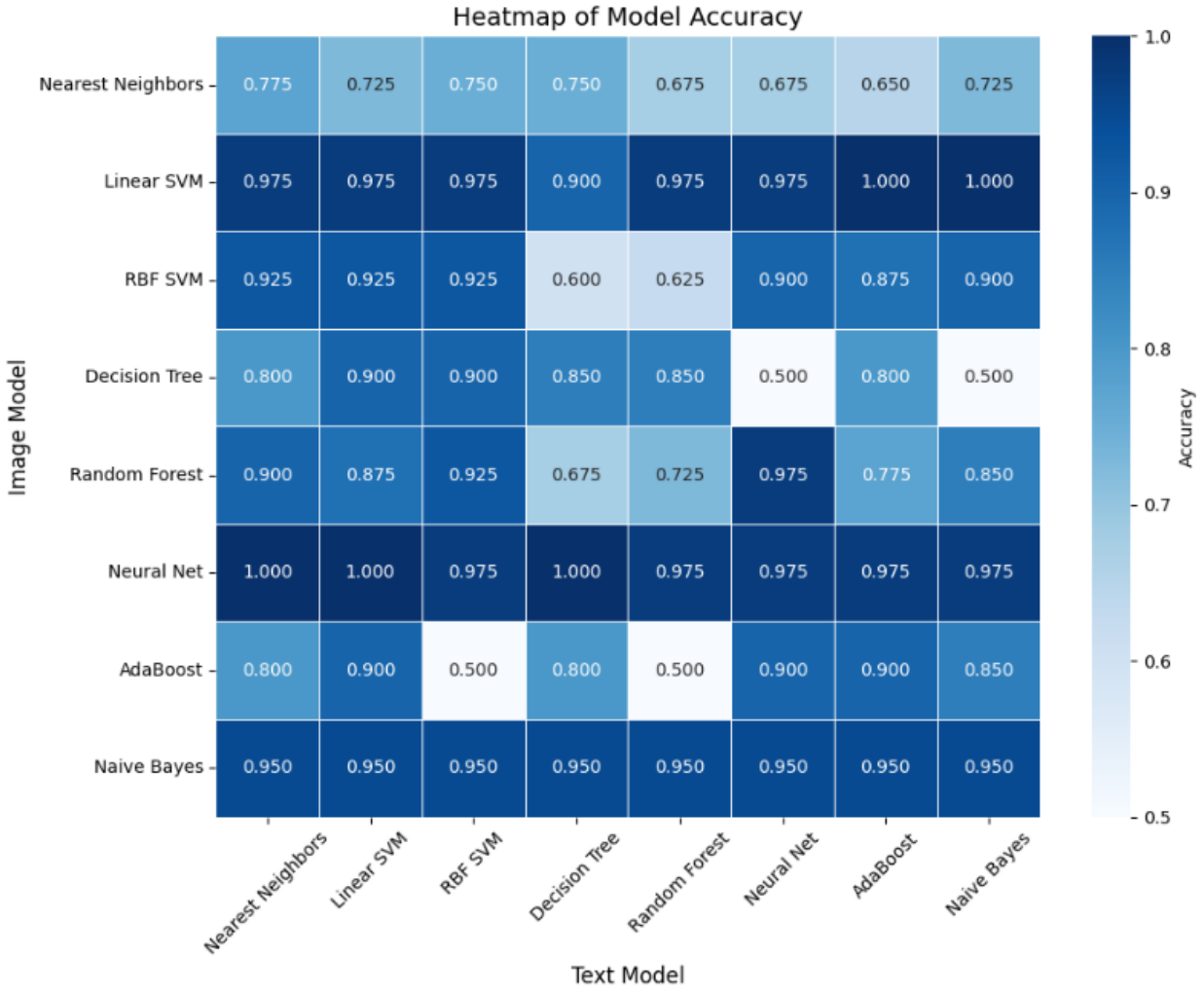
Với trường hợp alpha = 0.4, ta có:

Bảng 3.6 Độ chính xác khi áp dụng mô hình phân loại kết hợp alpha=0.4

Text / Image	KNN	Linear SVM	RBF SVM	Decision Tree	Random Forest	Neural Net	AdaBoost	Naive Bayes
K-Nearest Neighbors	0.775	0.725	0.75	0.750	0.675	0.675	0.650	0.725
Linear SVM	0.975	0.975	0.975	0.900	0.975	0.975	**1.000**	**1.000**
RBF SVM	0.925	0.925	0.925	0.600	0.625	0.900	0.875	0.900
Decision Tree	0.800	0.900	0.900	0.850	0.850	0.500	0.800	0.500
Random Forest	0.900	0.875	0.925	0.675	0.725	0.975	0.775	0.85
Neural Net	**1.000**	**1.000**	0.975	**1.000**	0.975	0.975	0.975	0.975
AdaBoost	0.800	0.900	0.500	0.800	0.500	0.900	0.900	0.850
Naive Bayes	0.950	0.950	0.950	0.950	0.950	0.950	0.950	0.950

Bảng 3.6 trình bày độ chính xác và thời gian huấn luyện – kiểm thử với các mô hình kết hợp, sử dụng giá trị alpha: 0.4. Hình 3.6 minh họa trực quan độ chính xác qua heatmap**.**

.



Hình 3.6 Biểu đồ nhiệt độ chính xác sử dụng giải thuật phân loại kết hợp alpha=0.4

Bảng kết quả thể hiện độ chính xác (Accuracy) khi kết hợp các mô hình xử lý trên dữ liệu văn bản và hình ảnh. Giá trị độ chính xác dao động từ 50% đến 100%, cho thấy sự phụ thuộc rõ rệt vào cách lựa chọn mô hình, phương pháp trích xuất đặc trưng và giá trị alpha.

Có thể thấy một cặp mô hình cho kết quả độ chính xác rất thấp như:

- RBF SVM (Text) + AdaBoost (Image): 50%
- Random Forest (Text) + AdaBoost (Image): 50%
- Neural Net (Text) + Decision Tree (Image): 50%
- Native Bayes (Text) + Decision Tree (Image): 50%

Điều này cho thấy rằng một số mô hình khi kết hợp không mang lại hiệu quả do sự không tương thích trong cách xử lý về đặc trưng giữa dữ liệu văn bản và hình ảnh.

Các mô hình văn bản như Linear SVM, Neural Net và Naive Bayes cho thấy độ ổn định cao khi kết hợp với đa số các mô hình hình ảnh, với độ chính xác dao động từ 90% đến 100%, như là:

- Linear SVM (Text) kết hợp với Neural Net, RBF SVM ,Naive Bayes hoặc chính Linear SVM (Image) đều cho độ chính xác từ 9,25% trở lên.
- Neural Net (Text) kết hợp với Linear SVM, Random Forest hoặc chính Neural Net (Image) đều đạt 97,5%.
- Naive Bayes (Text) khi kết hợp với Linear SVM, Neural Net, hoặc Naive Bayes (Image) đều đạt từ 95% trở lên.

Những cặp mô hình đạt độ chính xác cao nhất (100%), trong bảng alpha 0.4 bao gồm:

- Nearest Neighbors (Text) – Neural Net (Image)
- Linear SVM (Text) – Neural Net (Image)
- Decision Tree (Text) – Neural Net (Image)
- AdaBoost (Text) – Linear SVM (Image)
- Naive Bayes (Text) – Linear SVM (Image)

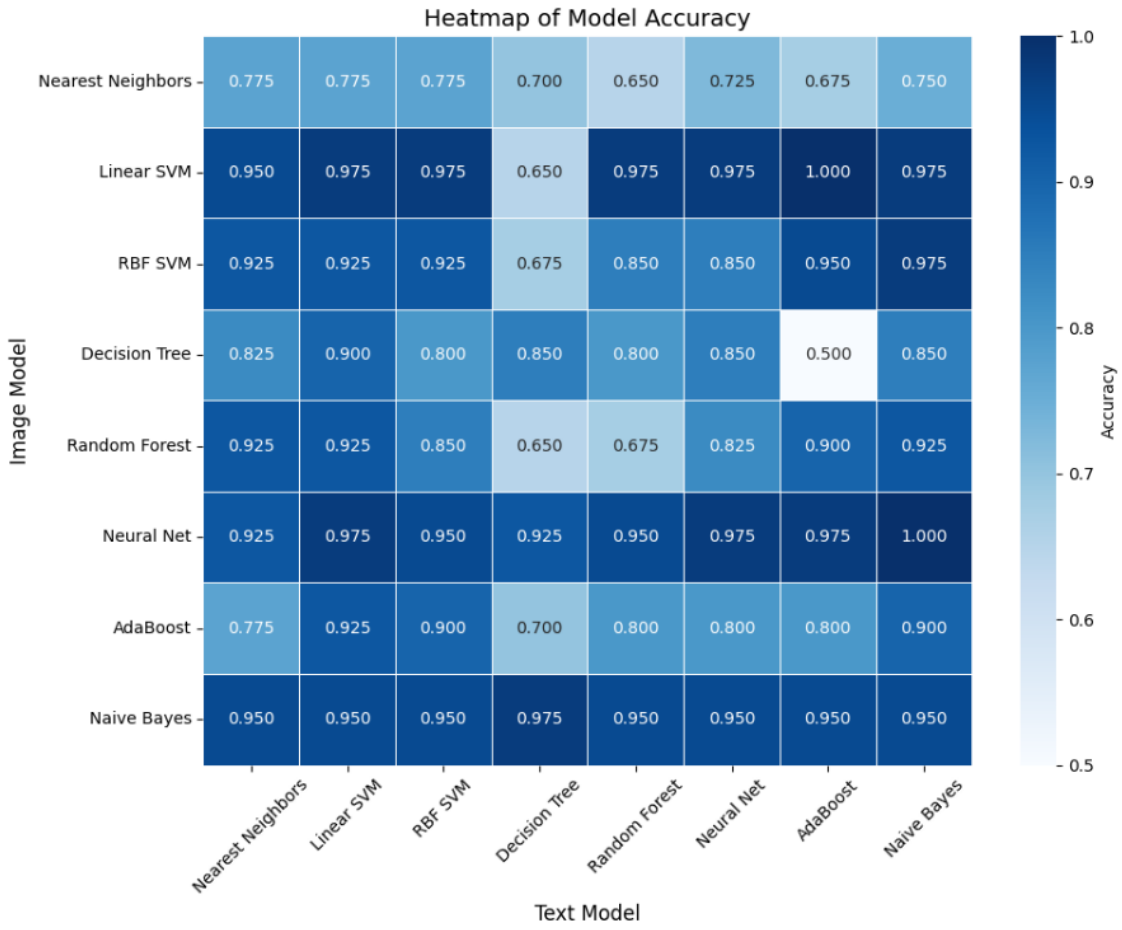
3.2. Phân lớp dữ liệu kết hợp giữa văn bản và hình ảnh với alpha = 0.5

Với trường hợp alpha = 0.5, ta có:

Bảng 3. 7 Độ chính xác khi áp dụng mô hình phân loại kết hợp alpha=0.5

Text / Image	KNN	Linear SVM	RBF SVM	Decision Tree	Random Forest	Neural Net	AdaBoost	Naive Bayes
K-Nearest Neighbors	0.775	0.775	0.775	0.700	0.650	0.725	0.675	0.750
Linear SVM	0.95	0.975	0.975	0.650	0.975	0.975	**1.000**	0.975
RBF SVM	0.925	0.925	0.925	0.675	0.850	0.850	0.950	0.975
Decision Tree	0.825	0.900	0.800	0.850	0.80	0.850	0.500	0.850
Random Forest	0.925	0.925	0.850	0.650	0.675	0.825	0.900	0.925
Neural Net	0.925	0.975	0.95	0.925	0.950	0.975	0.975	**1.000**
AdaBoost	0.775	0.925	0.900	0.700	0.800	0.800	0.800	0.900
Naive Bayes	0.950	0.950	0.950	0.975	0.950	0.950	0.950	0.950

Bảng 3.7 trình bày độ chính xác và thời gian huấn luyện – kiểm thử với các mô hình kết hợp, sử dụng ba giá trị alpha: 0.5. Hình 3.7 minh họa trực quan độ chính xác qua heatmap**.**



Hình 3. 7 Độ chính xác khi áp dụng mô hình phân loại kết hợp alpha=0.5

Bảng kết quả thể hiện độ chính xác (Accuracy) khi kết hợp các mô hình xử lý trên dữ liệu văn bản và hình ảnh với alpha = 0.5. Giá trị độ chính xác dao động từ 50% đến 100%, phản ánh rõ sự ảnh hưởng của việc lựa chọn mô hình và đặc trưng trích xuất đối với hiệu suất phân loại.

Cặp mô hình có hiệu suất thấp đáng kể, như là:

- AdaBoost (Text) + Decision Tree (Image) : 50%

Điều này cho thấy Decision Tree trong xử lý văn bản có xu hướng quá khớp. Ngược lại, một số mô hình văn bản như Linear SVM, Neural Net và Naive Bayes thể hiện sự ổn định và hiệu quả cao khi kết hợp với đa dạng mô hình hình ảnh, như là:

- Linear SVM (Text) kết hợp với Neural Net, Naive Bayes, hoặc chính Linear SVM (Image) đều đạt độ chính xác từ 95% đến 100%.

- Neural Net (Text) khi kết hợp với Linear SVM hoặc chính Neural Net (Image) đều cho kết quả 95% trở lên.
- Naive Bayes (Text) kết hợp với Neural Net, Linear SVM hoặc chính Naive Bayes (Image) đều đạt 95% trở lên.

Các cặp mô hình đạt hiệu suất tối đa (100%), trong bảng alpha bằng 0.5 gồm:

- AdaBoost (Text) – Linear SVM(Image)
- Native Bayes (Text) – Nerual Net (Image)

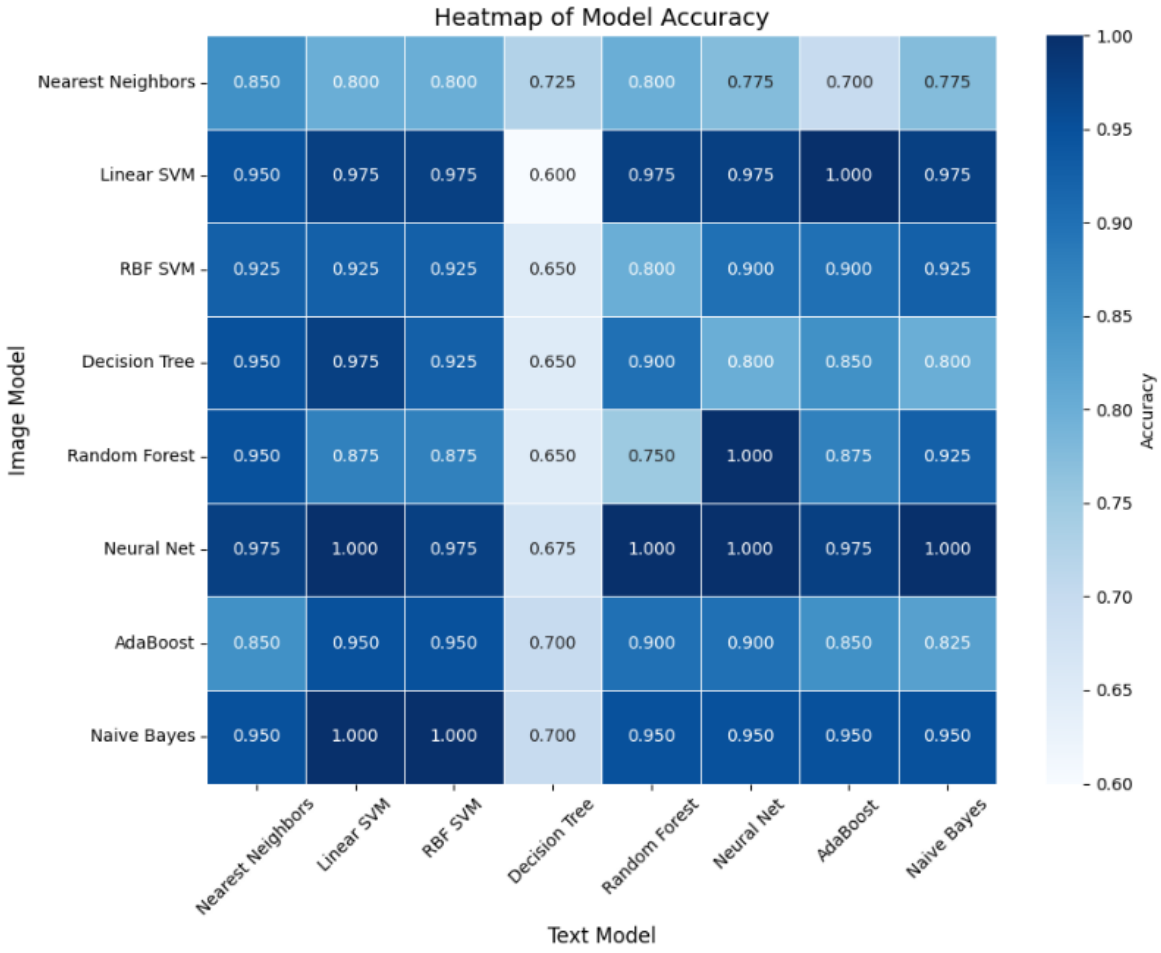
3.3. Phân lớp dữ liệu kết hợp giữa văn bản và hình ảnh với alpha = 0.6

Với trường hợp alpha = 0.6, ta có:

Bảng 3. 8 Độ chính xác khi áp dụng mô hình phân loại kết hợp alpha=0.6

Text / Image	KNN	Linear SVM	RBF SVM	Decision Tree	Random Forest	Neural Net	AdaBoost	Naive Bayes
K-Nearest Neighbors	0.850	0.800	0.800	0.725	0.800	0.775	0.700	0.775
Linear SVM	0.950	0.975	0.975	0.600	0.975	0.975	**1.000**	0.975
RBF SVM	0.925	0.925	0.925	0.650	0.800	0.900	0.900	0.925
Decision Tree	0.950	0.975	0.925	0.650	0.900	0.800	0.850	0.800
Random Forest	0.950	0.875	0.875	0.650	0.750	**1.000**	0.875	0.925
Neural Net	0.975	**1.000**	0.975	0.675	**1.000**	**1.000**	0.975	**1.000**
AdaBoost	0.850	0.950	0.950	0.700	0.900	0.900	0.850	0.825
Naive Bayes	0.950	**1.000**	**1.000**	0.700	0.950	0.950	0.950	0.950

Bảng 3.8 trình bày độ chính xác và thời gian huấn luyện – kiểm thử với các mô hình kết hợp, sử dụng ba giá trị alpha: 0.6. Hình 3.8 minh họa trực quan độ chính xác qua heatmap



Hình 3. 8 Độ chính xác khi áp dụng mô hình phân loại kết hợp alpha=0.6

Bảng kết quả thể hiện độ chính xác (Accuracy) khi kết hợp các mô hình xử lý trên dữ liệu văn bản và hình ảnh. Giá trị độ chính xác dao động từ 60% đến 100%, tiếp tục cho thấy sự phụ thuộc đáng kể vào cách lựa chọn mô hình và phương pháp trích xuất đặc trưng.

Một số cặp mô hình vẫn cho kết quả thấp, như là:

- Decision Tree (Text) + Linear SVM (Image): 60%
- Decision Tree (Text) + RBF SVM (Image): 65%
- Decision Tree (Text) + Decision Tree (Image) : 65%
- Decision Tree (Text) + Random Forest (Image): 65%

Điều này phản ánh rằng mô hình Decision Tree trong vai trò xử lý hình ảnh không mang lại hiệu quả khi kết hợp với nhiều mô hình, tương tự các nhận xét trước đó ở alpha 0.5.

Trong khi đó, các mô hình văn bản như Linear SVM, Neural Net và Naive Bayes tiếp tục thể hiện sự ổn định cao, khi kết hợp với phần lớn các mô hình xử lý hình ảnh, với độ chính xác đạt từ 90% đến 100%, cụ thể:

- Linear SVM (Text) kết hợp với RBF SVM, Decision Tree, Neural Net, AdaBoost, hoặc chính Linear SVM (Image) đều đạt ≥ 92.5%, trong đó với Neural Net và AdaBoost đạt 100%.
- Neural Net (Image) kết hợp với các mô hình Linear SVM, Random Forest, Neural Net hoặc Naive Bayes (Text) đều đạt 100%, cho thấy khả năng khái quát tốt và phù hợp khi trộn đặc trưng văn bản – hình ảnh.
- Naive Bayes (Text) khi kết hợp với các mô hình mạnh như Linear SVM, RBF SVM, Random Forest, Neural Net hoặc chính Naive Bayes (Image) đều đạt từ 92,5% đến 100%.

Các cặp mô hình đạt độ chính xác cao nhất (100%) trong bảng kết quả alpha = 0.6 gồm:

- Linear SVM (Text) – Neural Net/ Native Bayes (Image)
- RBF SVM (Text) – Native Bayes (Image)
- Random Forest (Text) – Neural Net (Image)
- Neural Net (Text) –Random Forest / Neural Net (Image)
- AdaBoost (Text) – Linear SVM (Image)
- Naive Bayes (Text) –Neural Net (Image)

3.4. Trình bày, giải thích cho các mô hình đạt độ chính xác 100% khi kết hợp

Dưới đây là phần giải thích cho danh sách các cặp mô hình kết hợp xử lý văn bản – hình ảnh đạt độ chính xác 100%, của các giá trị alpha = 0.4, 0.5, 0.6

Nearest Neighbors (Text) – Neural Net (Image): Ở alpha = 0.4, phần hình ảnh (Neural Net) được ưu tiên hơn (60%). Đây là lựa chọn hợp lý vì Neural Net có khả năng khai thác hiệu quả đặc trưng ảnh trích xuất từ VGG19. Trong khi đó, KNN là mô hình đơn giản nhưng vẫn hoạt động hiệu quả nếu đặc trưng văn bản đã được biểu diễn tốt bằng TF-IDF. Sự kết hợp này cho thấy Neural Net có thể bù đắp cho khả năng phân loại phi tuyến kém của KNN, dẫn đến kết quả tối ưu.

Linear SVM (Text) – Neural Net (Image): Cặp mô hình này duy trì hiệu quả cao ở cả alpha = 0.4 và 0.6, chứng tỏ tính ổn định và bổ trợ tốt. Linear SVM xử lý dữ liệu văn bản tuyến tính tốt, trong khi Neural Net học được đặc trưng hình ảnh phi tuyến mạnh mẽ từ VGG19. Khi alpha tăng (0.6), phần văn bản được nhấn mạnh nhưng độ chính xác vẫn đạt tối

đa nhờ sức mạnh của SVM. Khi alpha giảm (0.4), Neural Net đóng vai trò chính. Sự cân bằng và mạnh mẽ của cả hai mô hình là chìa khóa cho hiệu quả vượt trội này.

Linear SVM (Text) – Naive Bayes (Image): Tại alpha = 0.6, Linear SVM được ưu tiên. Mặc dù Naive Bayes là mô hình đơn giản trong xử lý ảnh, nhưng vẫn đạt 100% khi kết hợp với Linear SVM nhờ đặc trưng ảnh đã được biểu diễn tốt từ VGG19. Sự đơn giản và tính tuyến tính cao của cả hai mô hình mang lại kết quả ổn định, đặc biệt là khi SVM đóng vai trò chính trong quyết định phân loại.

Decision Tree (Text) – Neural Net (Image): Với alpha = 0.4, phần hình ảnh được ưu tiên. Neural Net khai thác tốt đặc trưng ảnh, trong khi Decision Tree thực hiện các quyết định phân loại theo luật rõ ràng. Tuy nhiên, Decision Tree vốn dễ overfit và không ổn định với dữ liệu chiều cao như văn bản. Nhờ Neural Net mạnh mẽ, mô hình ảnh đã hỗ trợ tốt để khắc phục điểm yếu này, dẫn đến hiệu suất tối đa trong trường hợp ưu tiên hình ảnh.

AdaBoost (Text) – Linear SVM (Image): Cặp mô hình này đạt 100% ở cả ba giá trị alpha (0.4, 0.5, 0.6), cho thấy tính ổn định và khả năng bổ sung mạnh mẽ. Linear SVM phân loại hiệu quả ảnh từ VGG19, trong khi AdaBoost là một kỹ thuật tăng cường kết hợp nhiều mô hình yếu để cải thiện hiệu suất xử lý văn bản. Dù văn bản hay hình ảnh được ưu tiên, cả hai mô hình đều thực hiện tốt phần việc của mình, dẫn đến kết quả ổn định.

Naive Bayes (Text) – Linear SVM (Image): Cặp mô hình này đạt hiệu suất tối đa ở alpha = 0.4. Trong trường hợp này, Linear SVM – mô hình mạnh trong phân loại ảnh – đóng vai trò chính. Naive Bayes, dù đơn giản, vẫn hoạt động tốt với dữ liệu văn bản đã chuẩn hóa như TF-IDF. Sự đơn giản, nhanh và dễ huấn luyện của cả hai mô hình giúp chúng bổ sung cho nhau hiệu quả.

Naive Bayes (Text) – Neural Net (Image): Với alpha = 0.5 và 0.6, kết quả đều đạt 100%. Khi alpha = 0.5, văn bản và hình ảnh có trọng số ngang nhau, cho thấy khả năng phối hợp tốt. Ở alpha = 0.6, phần văn bản (Naive Bayes) được ưu tiên. Dù đơn giản, Naive Bayes hoạt động tốt nhờ dữ liệu văn bản đã chuẩn hóa tốt, còn Neural Net vẫn khai thác hiệu quả phần ảnh. Mô hình đơn giản – phức tạp phối hợp tạo nên sự cân bằng giữa khả năng khái quát và độ chính xác.

RBF SVM (Text) – Naive Bayes (Image): Tại alpha = 0.6, phần văn bản (RBF SVM) được ưu tiên. RBF SVM là một mô hình mạnh, có khả năng học phi tuyến rất tốt từ văn bản với TF-IDF. Trong khi đó, Naive Bayes – tuy đơn giản – vẫn đạt hiệu quả cao nhờ đặc trưng ảnh từ VGG19. Cặp mô hình này cho thấy khả năng tổng quát hóa tốt khi kết hợp mô hình mạnh cho văn bản và mô hình nhẹ cho ảnh.

Random Forest (Text) – Neural Net (Image): Ở alpha = 0.6, văn bản được nhấn mạnh hơn. Tuy nhiên, Neural Net vẫn đóng vai trò quan trọng trong xử lý ảnh với đặc trưng phức tạp. Random Forest là mô hình mạnh có khả năng khái quát hóa tốt, đặc biệt khi xử lý đặc trưng văn bản có chiều cao như TF-IDF. Sự phối hợp với Neural Net giúp đảm bảo độ chính xác cao, dù mỗi mô hình phụ trách một miền dữ liệu khác nhau.

Neural Net (Text) – Random Forest (Image): Cặp mô hình này đạt 100% ở alpha = 0.6, trong đó Neural Net xử lý phần văn bản và được ưu tiên. Neural Net giúp học sâu các đặc trưng văn bản, còn Random Forest xử lý ảnh một cách ổn định. Khi đặc trưng ảnh đã được VGG19 biểu diễn tốt, Random Forest không cần quá phức tạp mà vẫn có thể phân loại hiệu quả, tạo nên sự phối hợp tối ưu với phần văn bản học sâu.

Neural Net (Text) – Neural Net (Image): Với alpha = 0.6, văn bản được ưu tiên, nhưng cả hai phần đều do Neural Net xử lý. Điều này cho phép hệ thống khai thác tối đa khả năng học sâu ở cả hai miền – văn bản và hình ảnh. Đặc biệt, khi đặc trưng ảnh đã được VGG19 biểu diễn rõ ràng, Neural Net chỉ cần học mối quan hệ giữa các đặc trưng đó. Mô hình này cho thấy khả năng kết hợp đồng nhất giữa hai nhánh học sâu, dẫn đến độ chính xác tối đa.

Nhìn chung, những mô hình hình ảnh như Linear SVM và Neural Net đều cho kết quả tốt với hầu hết các mô hình văn bản khác, đặc biệt khi đặc trưng ảnh được trích xuất từ VGG19 – một mạng học sâu hiệu quả trong biểu diễn đặc trưng hình ảnh. Đáng chú ý, ngay cả Naive Bayes - mô hình đơn giản , cũng đạt độ chính xác cao khi kết hợp đúng cách, chứng tỏ tính hiệu quả của phương pháp kết hợp này. Sự kết hợp của các mô hình này, giúp nâng cao hiệu quả phân loại nhờ vào sự bổ sung giữa tính tuyến tính, khả năng học sâu và tổng quát hóa tốt từ hai nguồn dữ liệu.

CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. Kết luận

Các mô hình phân loại khi áp dụng trên dữ liệu văn bản và hình ảnh đều đạt được độ chính xác cao và thời gian xử lý ở mức có thể chấp nhận được. Đặc biệt, với dữ liệu văn bản, hầu hết các mô hình đều hoạt động tốt, trong đó Bayes thơ ngây (Naive Bayes) nổi bật nhờ hiệu quả và tốc độ nhanh. Tuy nhiên, mô hình cây quyết định (Decision Tree) và rừng ngẫu nhiên (Random Forest) lại thể hiện hiệu suất thấp hơn so với các mô hình khác.

Đối với tập dữ liệu hình ảnh, hai mô hình thể hiện nổi trội là Linear SVM và mạng nơ-ron nhân tạo (Neural Net), đạt được sự cân bằng tốt giữa độ chính xác cao và thời gian xử lý hợp lý.

Khi kết hợp cả dữ liệu văn bản và hình ảnh, phần lớn các mô hình đều duy trì hiệu suất cao. Tuy nhiên, với cây quyết định (Image) khi kết hợp hai loại dữ liệu cho kết quả không như mong đợi, có thể do tính chất phức tạp của đặc trưng hình ảnh gây ra hiện tượng quá khớp (overfitting). Điều này cho thấy không phải mọi mô hình đều phù hợp để xử lý dữ liệu đa phương thức.

Tổng thể, mỗi mô hình sẽ phát huy thế mạnh trên từng dạng dữ liệu cụ thể: Bayes thơ ngây phù hợp với văn bản nhờ giả định độc lập giữa các đặc trưng, trong khi mạng nơ-ron và SVM thường hiệu quả hơn với dữ liệu hình ảnh nhờ khả năng khai thác đặc trưng phức tạp. Kết quả này cung cấp cơ sở quan trọng để lựa chọn mô hình phù hợp cho các bài toán xử lý đa phương thức trong thực tế.

2. Kết quả đạt được

Đã thực hiện đánh giá hiệu suất của các mô hình máy học trên ba loại dữ liệu: văn bản, hình ảnh và sự kết hợp giữa hai loại này.

Xác định và phân tích mô hình đạt độ chính xác cao nhất trong từng trường hợp.

Xác định phương pháp kết hợp dữ liệu văn bản và hình ảnh tối ưu nhằm nâng cao hiệu quả phân loại.

3. Hướng phát triển

Tối ưu hóa các mô hình hiện có thông qua việc tinh chỉnh siêu tham số, điều chỉnh tốc độ học, tăng chiều sâu mạng nơ-ron, và áp dụng các kỹ thuật như dropout để hạn chế overfitting, sẽ giúp cải thiện đáng kể hiệu suất.

tăng cường dữ liệu huấn luyện bằng các phương pháp biến đổi ảnh và văn bản sẽ nâng cao khả năng khái quát hóa của mô hình. Sử dụng các phương pháp như xoay ảnh, cắt, lật, thêm nhiễu hoặc biến dạng để cải thiện tính đa dạng và độ khái quát của mô hình.

Khám phá tiềm năng của các kiến trúc mạng sâu hiện đại như Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) và các biến thể khác trong xử lý cả văn bản và hình ảnh.

Ứng dụng học chuyển tiếp (Transfer Learning) với các mô hình tiền huấn luyện như VGG, ResNet hoặc BERT và tinh chỉnh lại trên tập dữ liệu cụ thể để tăng hiệu quả sẽ giúp tiết kiệm thời gian và tài nguyên tính toán đáng kể.

Áp dụng kỹ thuật Ensemble để kết hợp nhiều mô hình khác nhau (voting, trung bình có trọng số, stacking) nhằm tạo ra một mô hình tổng hợp có độ chính xác và khả năng khái quát cao hơn.

TÀI LIỆU THAM KHẢO

[1] H. M. P. S. D. Hand, Principles of Data Mining, The MIT Press, 2001.

[2] H. W. D. B. Y. B. K. G. Gongde Guo, "KNN Model-Based Approach in Classification," trong *OTM Confederated International Conferences: CoopIS, DOA, and ODBASE*, Catania, Italy, 2003.

[3] F. Sebastiani, "Machine Learning in Automated Text Categorization," *ACM Computing Surveys,* tập 34, p. 1–47, March 2002.

[4] T. E. a. M. Pontil, "SUPPORT VECTOR MACHINES: THEORY AND APPLICATIONS," *Lecture Notes in Computer Science,* 2001.

[5] A. G. A. C. Malti Bansal, "A comparative analysis of K-Nearest Neighbor, Genetic, Support Vector Machine, Decision Tree, and Long Short Term Memory algorithms in," *ScienceDirect,* 2022.

[6] H. C. H. H.] Qiong Ren, ""Research on Machine Learning Framework Based on Random Forest Algorithm," trong *AIP Conference Proceedings*, 2017.

[7] Zupan, Jure, "Introduction to Artificial Neural Network (ANN) Methods: What They Are and How to Use Them," *Acta Chimica Slovenica,* tập 41, số 3, pp. 327-352, 1994.

[8] H. T. P. Thảo, "Ứng dụng mạng nơron nhân tạo để đánh giá mức độ ảnh hưởng của các nhân tố đến sự thỏa mãn công việc," *Tạp chí Khoa học và Công nghệ, Đại học Đà Nẵng,* số 11(72), pp. 60-65, 2013.

[9] L. X.Li, "A Study of AdaBoost with SVM Based Weak Learners," trong *Proceedings of International Joint Conference on Neural Network*, 2005.