|  |
| --- |
| Trường Đại Học Sư Phạm Kỹ Thuật TPHCMTRƯỜNG ĐẠI HỌC  SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  HCMC University of Technology and Education |
| NHẬN DẠNG NGÔN NGỮ TIẾNG VIỆT QUA VĂN BẢN WEB,HÌNH ẢNH |
| GVHD:  PGS.TS Nguyễn Trường Thịnh  SVTH:  NGUYỄN NHẬT TIẾN – 19146273  TP Hồ Chí Minh, ngày 23 tháng 6 năm 2022 |

**1.GIỚI THIỆU**

Với nhu cầu muốn tìm hiểu rõ hơn về trí tuệ nhân tạo và đặc biệt là về NLP (Neuro – Linguistic – Programming) thì tôi đã chọn đề tài nhận dạng ngôn ngữ tiếng việt sử dụng machine learning + NLP với những kiến thức đã được học ở lớp và tự tìm hiểu khi ở nhà nhầm tạo ra một model có thể dự đoán ngôn ngữ tiếng việt thông qua web hoặc hình ảnh văn bản.

Xử lý văn bản được lấy từ một link website bất kì hoặc xử lý hình ảnh có chứa văn bản từ người dùng nhập vào rồi từ đó đtôi đi so sánh với mô hình đã được huấn luyện sẵn. Tiếp đến mô hình sẽ xử lý là dự đoán là dữ liệu người dùng nhập vào có

phải là tiếng việt hay không rồi sau đó trả ra kết quả.

Tất cả sẽ được tương tác trên một website tự tạo.

Đề tài chỉ dừng lại ở việc nhận biết ngôn ngữ tiếng việt chứ không phải các dạng ngôn ngữ khác.

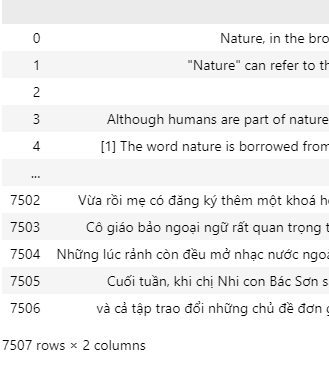
Đề tài sẽ chỉ sử dụng CNN (mạng nơron tích chập) và NLP cơ bản để xử lý nên độ chính xác sẽ không cao như khi dùng các model đã được pretrain như YOLO,VGG16,…

Với việc không có sẵn dữ liệu để huấn luyện mô hình nên dữ liệu tự thêm có thể không được hiệu quả và tối ưu cho mô hình nhận diện

**2.CHUẨN BỊ DỮ LIỆU**

Ban đầu ta sẽ có 2 bộ dữ liệu thô gồm một bộ dữ liệu cho việc xử lý văn bản tiếng việt và một bộ xử lý dữ liệu dành cho việc nhận dạng văn bản tiếng việt thông qua hình ảnh.

Với dữ liệu sẽ được chọn lọc từ trên internet đối với bộ dữ liệu xử lý văn bản thì dữ liệu chủ yếu sẽ được lấy từ các bài báo, văn bản mẫu, công văn,… và được tổng hợp vào một file csv gồm 7506 dòng văn bản.



Hình 1. 1 Dữ liệu file csv

Đối với dữ liệu hình ảnh sẽ là hình ảnh được cắt trực tiếp từ nhiều nguồn trên internet như các trang bài báo vn express, new york times, BCC,…

Hình ảnh sẽ được chứa trong hai thư mục TV và NTV (335 TV và 270 tấm NTV)

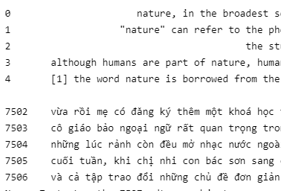


Hình 1. 2 Hai thư mực chứa ảnh văn bản

**3. TIỀN XỬ LÝ DỮ LIỆU VĂN BẢN TIẾNG VIỆT**

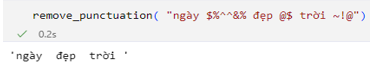
Với việc tổng hợp được 7506 dòng văn bản để phục vụ cho việc huấn luyện mô hình ta tiến hành

xử lý bằng cách cho tất cả chữ in hoa trong tập dữ liệu thành chữ thường để đồng hoá dữ liệu về một dạng giúp quá trình học trở nên hiệu quả hơn



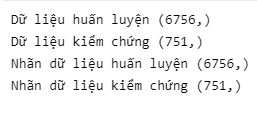
Hình 1. 3 Biến chữ hoa thành chữ thường

Tiếp đến là loại bỏ các kí tự đặc biệt trong đoạn văn bản để văn bản khi đưa vào sẽ trở nên sạch hơn. Ta tạo một function trả về một bản sao của chuỗi ban đầu trong đó các ký tự đã được thay thế sử dụng dictionary hoặc mapping table.



Hình 1. 4 Ví dụ về việc loại bỏ kí tự đặc biệt trong văn bản

Sau khi qua hai bước xử lý trên ta tiến hành phần chia dữ liệu để tiến hành huấn luyện

****

Hình 1. 5 Dữ liệu chia từ 7506 dòng văn bản

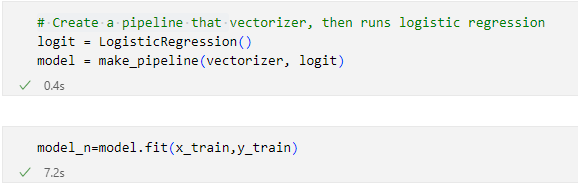
Cuối cùng trong việc xử lý dữ liệu chính là sử dụng thuật toán TfidVectorize để đánh giá mức độ quan trọng của các cụm từ trong tập dữ liệu và biến nó thành một vector đặc trưng được sử dụng để huấn luyện mô hình dự đoán ngôn ngữ

**4. XÂY DỰNG MÔ HÌNH DỰ ĐOÁN TIẾNG VIỆT QUA VĂN BẢN**

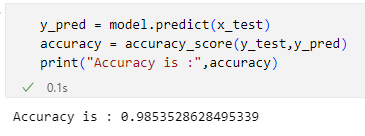
Đối với mô hình này sẽ sử dụng pipeline trong machine learning

Pipeline là một phần không thể thiếu trong một model machine learning vì trong một hệ thống machine learning thường sẽ có rất nhiều thành phần như xử lý dữ liệu, áp dụng thuật toán huấn luyện mô hình, gán nhãn dữ liệu, đánh giá mô hình,v.v. Nếu ta không xây dựng một pipeline hoàn chỉnh với từng phần xử lý tách biệt hoàn toàn thì có nhiều vấn đề có thể xảy ra nên việc tách chúng ra và ghép lại thành một pipeline hoàn chỉnh giúp ta dễ dàng hơn khi huấn luyện mô hình cũng như tìm lỗi cho nó

Sử dụng hàm make\_pipeline() của sklearn để tạo nên một quy trình là xử lý dữ liệu với TfidVectorize là lấy một cụm gồm 3 từ có mức độ quan trọng cao trong từng đoạn văn bản trong tập dữ liệu và biến nó thành một vector đặc trưng và sẽ lấy vector đó là dữ liệu đầu vào để huấn luyện mô hình nhận dạng. Sau đó sẽ áp dụng thuật toán hồi quy Logisic để huấn luyện mô hình

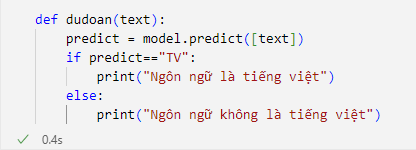


Hình 1. 6 Tạo pipeline và huấn luyện mô hình

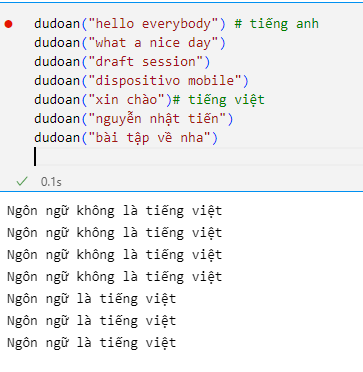


Hình 1. 7 Độ chính xác của mô hình

**5. KIỂM NGHIỆM MÔ HÌNH DỰ ĐOÁN TIẾNG VIỆT QUA VĂN BẢN**



Hình 1. 8 Hàm dùng để dự đoán



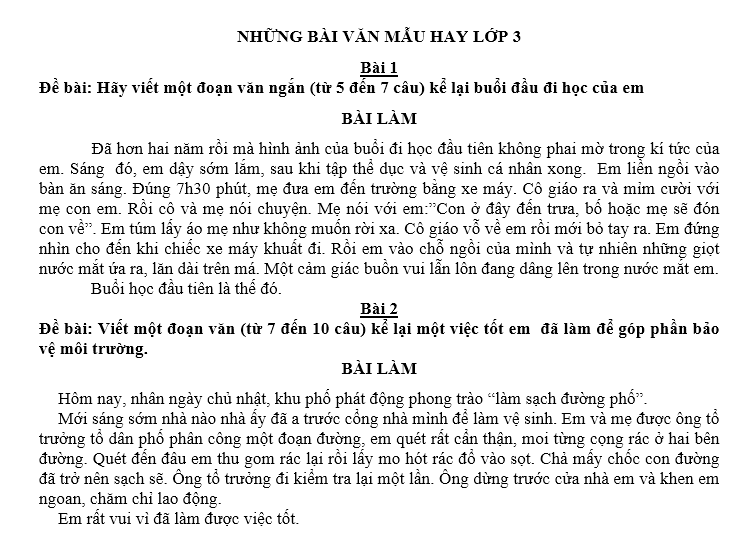
Hình 1. 9 Kết quả dự đoán

Nhận xét: Mô hình dự đoán tương đối chính xác với độ chính xác 98% là một mô hình ta có thể tin cậy được.

**6.** **TIỀN XỬ LÝ DỮ LIỆU HÌNH ẢNH VĂN BẢN TIẾNG VIỆT**

Việc quan trọng trong việc xử lý hình ảnh là cân bằng màu sắc ảnh vì mỗi một điểm ảnh (pixel) của một bức ảnh được cấu tạo từ một ma trận. Mỗi bức ảnh sẽ được chia thành ba lớp màu là đỏ, xanh lá, xanh dương và mức độ đậm nhạt của từng màu sẽ khác nhau từ mức độ 0 đến 255. Nếu chúng ta để mặc định màu từ 0 đến 255 thì mô hình sẽ thiên vị cho các màu sắc có giá trị lớn hơn với những sự thay đổi màu sắc nhỏ sẽ không làm ảnh hưởng dến mô hình làm giảm hiệu năng của mô hình. Vì thế điều đầu tiên khi xử lý hình ảnh là đưa về khoảng giá trị 0 đến 255 về 0 đến 1 bằng cách chia tất cả dữ liệu cho 255.

Đối về đề tài thì do là hình ảnh văn bản nền trắng mực đen nên việc chia cho 255 cũng gọi là không thực sự cần thiết.



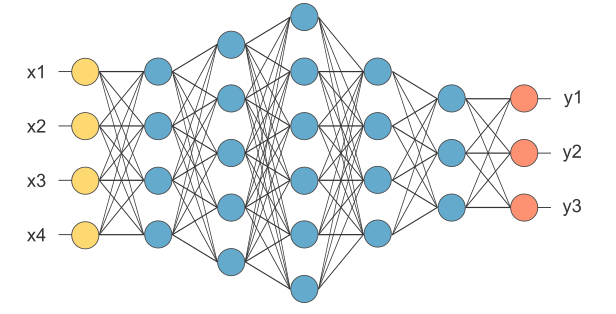
Hình 2. 1 Dạng văn bản tiếng việt dùng để huấn luyện

Tiếp đến ta sẽ tạo độ đa dạng cho hình ảnh bằng cách thay đổi hình ảnh như xoay một góc 40 độ thay dổi kích thước hình ảnh cho lật ngang lật dọc phóng to ảnh,….

Và số ảnh sẽ được dùng cho việc train là 478 ảnh và 119 ảnh còn lại sẽ dùng cho việc validation.

**7. XÂY DỰNG MÔ HÌNH NHẬN DIỆN NGÔN NGỮ TIẾNG VIỆT BẰNG HÌNH ẢNH**

Ý tưởng của mô hình này là sử dụng CNN còn hay gọi là mạng nơ ron thần kinh tích chập.

Về mặt kỹ thuật, trong mô hình học sâu CNN, mô hình ảnh đầu vào sẽ chuyển nó qua một loạt các lớp tích chập với các bộ lọc, sau đó đến lớp Pooling, rồi tiếp theo là các lớp được kết nối đầy đủ (FC — fully connected layers) và cuối cùng áp dụng hàm softmax để phân loại một đối tượng dựa trên giá trị xác suất trong khoản từ 0 đến 1.

Hình 2. 2 Mạng thần kinh nơron

Convolution (lớp tích chập) là lớp đầu tiên trích xuất các đặc tính từ hình ảnh. Tham số lớp này bao gồm một tập hợp các bộ lọc có thể học được. Các bộ lọc đều nhỏ thường có kích cỡ hai chiều khoảng 3x3 hoặc 5x5, .... và có độ sâu bằng với độ sâu của đầu vào đầu vào. Bằng cách trượt dần bộ lọc theo chiều ngang và dọc trên ảnh, chúng thu được một Feature Map chứa các đặc trưng được trích xuất từ trên hình ảnh đầu vào.

Pooling layer thường được dùng giữa các convolutional layer, để giảm kích thước dữ liệu nhưng vẫn giữ được các thuộc tính quan trọng. Kích thước dữ liệu giảm giúp giảm việc tính toán trong model. Trong quá trình này, quy tắc về stride và padding áp dụng như phép tính convolution trên ảnh.

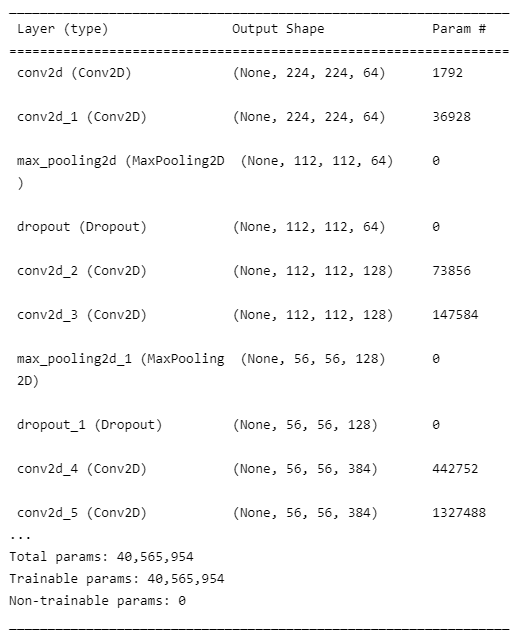
Fully connected layer: Sau khi ảnh đã truyền qua các lớp tích chập và pooling để xử lý thì lúc này mô hình đã được học tương đối các đặc điểm của ảnh thì output của layer cuối cùng sẽ được làm phẳng (Flatten) từ một ma trận thành một vector để đưa vào một lớp được kết nối như một mạng nơron, cuối cùng dùng softmax hoặc sigmoid để phân loại đầu ra.

Với ảnh đầu vào là 224x224x3 ta sẽ sử dụng 64 kernel với bộ lọc ma trận là 3x3 để trượt qua từng ảnh gốc làm nổi bật lên các đặc trưng đồng thời sử dụng thêm các lớp padding để giữ lại kích thước gốc ban đầu của ảnh sau đó ta sẽ thêm một activision (relu) phi tuyến để có thể sấp sỉ được các phân bố phức tạp và không làm thay đổi kích thước ban đầu của ảnh.Tiếp đến ta thêm các lớp MaxPooling 2D, lớp này dùng để chọn ra những đặc trưng nổi bật làm tăng độ chính xác cũng như làm giảm kích thước ảnh.

Tiếp đến là lớp DropOut để qua một 1 nút mạng trong quá trình đạo tạo một cách ngẫu nhiên giúp việc học trở nên mạnh mẽ, tăng gấp đôi số eposch cần để hội tụ,…

Với các Block tiếp theo sẽ tương tự nhưng tăng lượng kernel lên là 128 và 384.

Ở Block cuối cùng để tiến hành phân loại đầu vào thì ta cần biến output của ảnh lúc này là một ma trận về một vector để có thể làm điều này từ ta sẽ cho qua một lớp Flatten. Sau đó thu lại số lớp phù hợp cho số lượng cần dự đoán là 2 tương ứng với hai lớp tiếng việt và không phải tiếng việt.Dùng softmax để tiến hành phân loại đầu ra.



Hình 2. 3 Mô hình mạng thần kinh nơron CNN sau khi được xây dựng

Tiếp đến chính là quá trình xây dựng cho mô hình các hàm loss và thuật toán tối ưu.

Hàm mất mát là một phương pháp đánh giá độ hiệu quả của thuật toán “học” cho mô hình trên tập dữ liệu được sử dụng. Hàm mất mát trả về một số thực không âm thể hiện sự chênh lệch giữa hai đại lượng: a, nhãn được dự đoán và y, nhãn đúng. Hàm mất mát, bản thân chính là một cơ chế thưởng-phạt, mô hình sẽ phải đóng phạt mỗi lần dự đoán sai và mức phạt tỉ lệ thuận với độ lớn sai sót. Trong mọi bài toán học có giám sát, mục tiêu luôn bao gồm giảm tổng mức phạt phải đóng. Trong trường hợp lý tưởng a = y, loss function sẽ trả về giá trị cực tiểu bằng 0 . Hai hàm mất mát thường xuyên được sử dụng trong mạng nơ-ron: MSE (Mean Squared Error) và Cross Entropy.

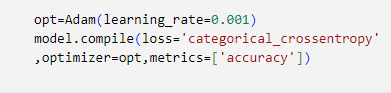
Trong thuật toán học máy nói chung và kĩ thuật học sâu nói riêng, thuật toán tối ưu hóa là một khâu quan trọng không thể thiếu. Quá trình tối ưu hóa thực hiện xác định hàm mất mát (loss function) và sau đó tối thiểu hóa hàm trên bằng cách sử dụng hàm tối ưu. Cụ thể, thông qua việc cập nhật các tham số của mô hình (w, b) và đánh giá lại hàm mất mát với một tỉ lệ học (learning rate) xác định, quá trình tối ưu giúp mô hình tương thích tốt hơn với tập dữ liệu được đào tạo.

Trong mô hình sẽ sử dụng thuật toán Adam (Adaptive Moment Estimation) là thuật toán tối ưu. Adam được xtôi như là sự kết hợp của RMSprop và Stochastic Gradient Descent với động lượng. Adam là một phương pháp tỉ lệ học thích ứng, nó tính toán tỉ lệ học tập cá nhân cho các tham số khác nhau. Adam sử dụng ước tính của khoảng thời gian thứ nhất và thứ hai của độ dốc để điều chỉnh tỉ lệ học cho từng trọng số của mạng nơ-ron.

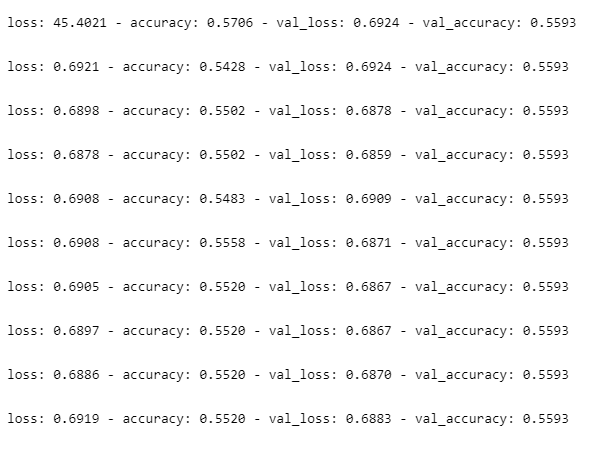
Với tỉ lệ học (learning\_rate) sẽ là 0.001 để tối ưu trọng số và tham số ngưỡng – bias trong quá trình học để giúp mô hình ổn định.

Hàm mất mát sẽ là hàm được thường xuyên trong mạng nơ ron cho bài toán phân loại là categorical\_crossentropy với thước đo đánh giá là accuracy

Hình 2. 6 Huấn luyện mô hình

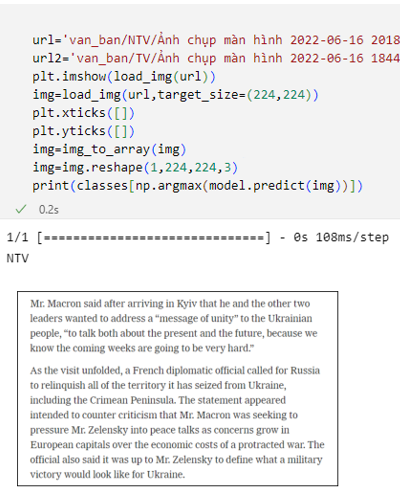


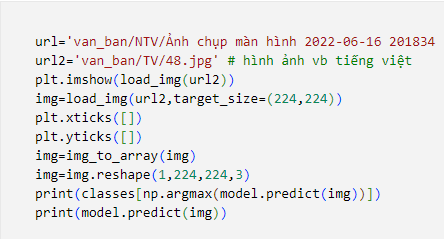
Hình 2. 4 Thuật toán tối ưu và hàm mất mát cho mô hình

Bước cuối cùng chính là huấn luyện mô hình với dữ liệu là đầu vào là hình ảnh văn bản đã được tạo ra từ ImageDataGenerator ở phần trước với batch\_size (số lần lấy mẫu trong một lần học) là 64 với số lần học (espoch) là 10 .

**Nhận xét:** Với sai số của tập dữ liệu huấn luyện là 0.6919 với độ chính xác là 0.552, sai số của tập kiểm chứng là 0.6883 với độ chính xác là 0.56

**Đánh giá:** Độ chính xác còn thấp chưa thể đtôi đi áp dụng thực tế cần cải thiện nhiều hơn về dữ liệu, tiền xử lý và thuật toán phù hợp.

**8. KIỂM NGHIỆM MÔ HÌNH NHẬN DIỆN NGÔN NGỮ TIẾNG VIỆT BẰNG HÌNH ẢNH**

****

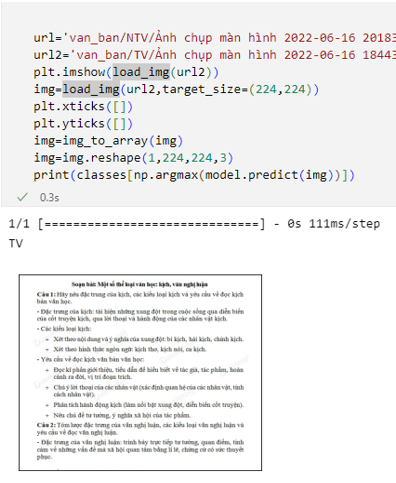
Hình 2. 7 Code cho việc dự đoán

Đầu tiên ta sẽ tạo mapping gồm 2 class là NTV và TV để phục vụ cho việc gắn nhãn dự đoán.

Sau đó chúng ta cho hiện ảnh mà ta chọn để dự đoán với scale với kích thước ảnh thành 224x224

Với việc ảnh là một ma trận điểm ảnh thì sau khi scale lai kích thước 224x 224 thì ta sẽ chuyển thành một mảng 224 hàng 224 cột 3 mã màu

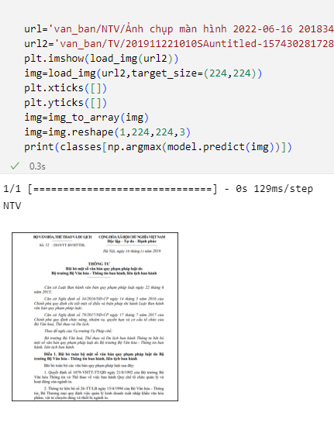
Cuối cùng là việc in là kết quả dự đoán kết hợp giữa mapping mà ta đã tạo với function argmax ( trả về chỉ số của phần tử lớn nhất trong một trục cụ thể)

Hình 2. 8 Đối với hình ảnh văn bản không phải tiếng việt

Hình 2. 9 Đối với hình ảnh văn bản là tiếng việt

Tuy nhiên mô hình còn đoán sai nhiều

do độ chính xác còn thấp cụ thể là 56%

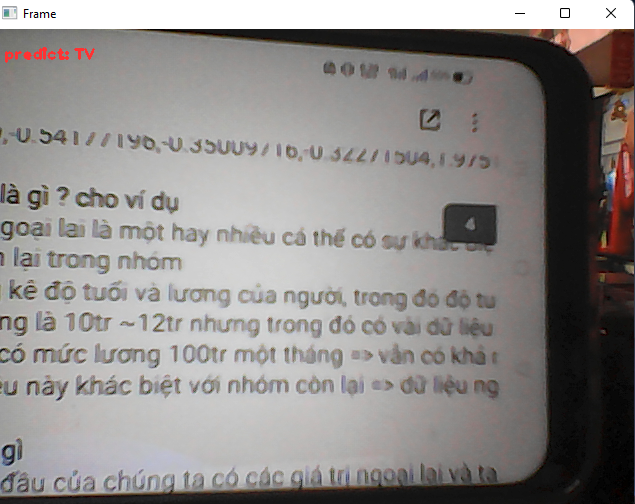


Hình 3. 1 Hình ảnh văn bản đoán sai

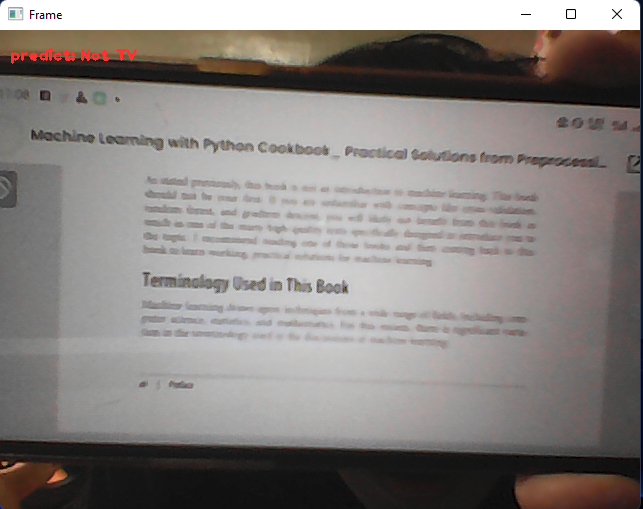
**9.Real time detection**



Hình 3. 2 Thử cam laptop



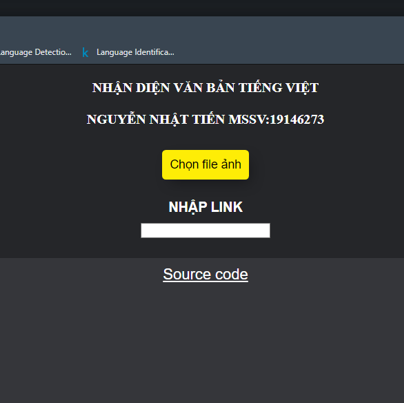
Hình 3. 3 Với hình ảnh văn bản tiếng việt



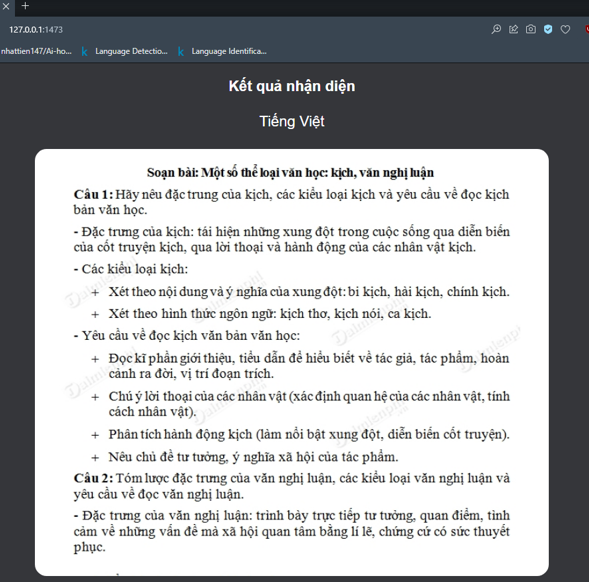
Hình 3. 4 Với hình ảnh văn bản ngôn ngữ khác

**10. Thiết kế giao diện Web**

Giao diện tương tác người dùng khá đơn giản được dựng lên từ ngôn ngữ HTM và môi trường Flask



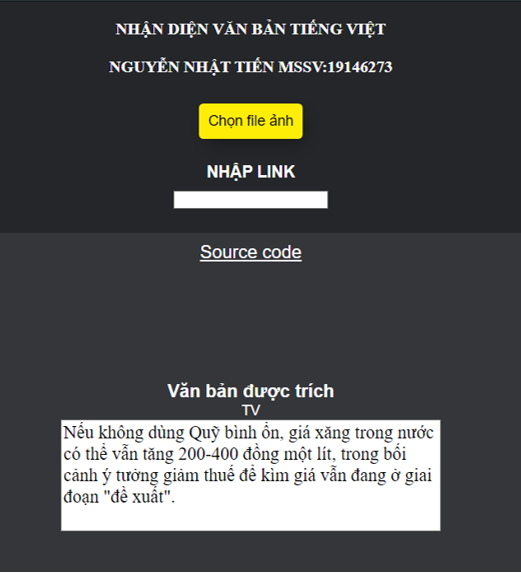
Hình 3. 5 Giao diện web

****

Hình 3. 6 Dự đoán hình ảnh được đưa vào

Ta có thể dự đoán ngôn ngữ văn bản web với việc đưa link vào ô NHẬP LINK để mô hình tiến hành đưa ra dự đoán

Link bài báo: <https://vnexpress.net/gia-xang-ngay-mai-co-the-tang-tiep-4477629.html>



Hình 3. 7 Dự đoán với việc nhập một link bài báo bất kì

**11. KẾT LUẬN**

Sau một thời gian thực hiện thì tôi đã cơ bản dựng lên được hai mô hình dự đoán ngôn ngữ tiếng việt thông qua văn bản và hình ảnh văn bản

Những điều đã làm được:

-Áp dụng dược những kiến thức đã được học trên lớp trí tuệ nhân tạo kết hợp với internet để thực hiện đề tài

-Tìm hiểu được thêm các ngôn ngữ khác như HTML, python

-Tiếp xúc được với môi trường Flask framework

-Huấn luyện thành công tương đối hai mô hình nhận dạng văn bản ngôn ngữ tiếng việt và nhận dạng văn bản hình ảnh ngôn ngữ tiếng việt

-Thiết kế giao diện website tương tác người dùng cơ bản

Những điều chưa làm được:

-Mô hình nhận dạng ngôn ngữ tiếng việt bằng hình ảnh văn bản còn khá thấp chỉ đạt được 56%

-Dữ liệu còn ít 607 tấm với lại chưa đuọc chọn lọc kĩ càng

**Định hướng**

Sẽ cải thiện mô hình bằng cách cải thiện dữ liệu đầu vào với nhiều hình ảnh và chọn lọc kĩ càng nhất có thể ngoài ra sẽ tính toán thiết kế lại mạng nơron để tối ưu hoá việc học cũng như dự đoán đạt được độ chính xác cao nhất

Ngoài ra sẽ tìm hiểu nhiều hơn nữa các tài liệu về trí tuệ nhận tạo các thuật toán để có sử dụng thuật toán tốt nhât cho mô hình

**12. TÀI LIỆU THAM KHẢO**

**[1]** Chris Albon. 2018, Python Machine Learning Cookbook

**[2]** Quing Kong, Timmy Siauw, Alexandre M. Bayen. 2020, Python Programming And Numerical Medthods

**[3]** Kaggle, Github

**[4]** Mì Ai

**[5]** Text detection using CNN by Manolis Delakis and Christophe Garcia

**[6]** Deep Learning Based OCR for Text in the Wild