BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐÀO TẠO CHẤT LƯỢNG CAO

TIỂU LUẬN CHUYÊN NGHÀNH

TÌM HIỂU VỀ TRỢ LÝ ẢO SỬ DỤNG HỌC SÂU VÀ VIẾT ỨNG DỤNG MINH HỌA

GVHD : **TS. Trần Nhật Quang**

SVTH : **Nguyễn Lê Bảo Thanh** **19110019**

Huỳnh Nguyễn Tấn Nhạc **19110252**

Lớp 19110CLST3

TP. Hồ Chí Minh, 10 tháng 12 năm 2022

ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA CHẤT LƯỢNG CAO

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

Họ và tên Sinh viên 1: Nguyễn Lê Bảo Thanh MSSV 1: 19110019

Họ và tên Sinh viên 2: Huỳnh Nguyễn Tấn Nhạc MSSV 2: 19110252

Ngành: Công nghệ Thông tin

Tên đề tài: Tìm hiểu về trợ lý ảo sử dụng học sâu và viết ứng dụng minh họa

Họ và tên Giảng viên hướng dẫn: TS. Trần Nhật Quang

NHẬN XÉT

1. Về nội dung đề tài khối lượng thực hiện:

.....

.....

2. Ưu điểm:

.....

.....

3. Khuyết điểm

.....

.....

4. Đánh giá loại :

5. Điểm :

TP. Hồ Chí Minh, ngày …. tháng 12 năm 2022

Giảng viên hướng dẫn

(ký & ghi rõ họ tên)

ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA CHẤT LƯỢNG CAO

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN PHẢN BIỆN

Họ và tên Sinh viên 1: Nguyễn Lê Bảo Thanh MSSV 1: 19110019

Họ và tên Sinh viên 2: Huỳnh Nguyễn Tấn Nhạc MSSV 2: 19110252

Ngành: Công nghệ Thông tin

Tên đề tài: Tìm hiểu về trợ lý ảo sử dụng học sâu và viết ứng dụng minh họa

Họ và tên Giảng viên phản biện: TS. Nguyễn Thiên Bảo

NHẬN XÉT

1. Về nội dung đề tài khối lượng thực hiện:

.....

.....

2. Ưu điểm:

.....

.....

3. Khuyết điểm

.....

.....

4. Đánh giá loại :

5. Điểm :

TP. Hồ Chí Minh, ngày …. tháng 12 năm 2022

Giảng viên hướng dẫn

(ký & ghi rõ họ tên)

MỤC LỤC

DANH MỤC HÌNH ẢNH

LỜI CẢM ƠN

Để hoàn thành tốt đề tài và bài báo cáo này, chúng em xin gửi lời cảm ơn chân thành đến giảng viên, thầy Trần Nhật Quang, người đã trực tiếp hỗ trợ chúng em trong suốt quá trình làm đề tài. Chúng em cảm ơn thầy đã đưa ra những lời khuyên từ kinh nghiệm thực tiễn của mình để định hướng cho chúng em đi đúng với yêu cầu của đề tài đã chọn, luôn giải đáp thắc mắc và đưa ra những góp ý, chỉnh sửa kịp thời giúp chúng em khắc phục nhược điểm và hoàn thành tốt cũng như đúng thời hạn đã đề ra.

Chúng em cũng xin gửi lời cảm ơn chân thành các quý thầy cô trong khoa Đào tạo Chất Lượng Cao nói chung và ngành Công Nghệ Thông Tin nói riêng đã tận tình truyền đạt những kiến thức cần thiết giúp chúng em có nền tảng để làm nên đề tài này, đã tạo điều kiện để chúng em có thể tìm hiểu và thực hiện tốt đề tài. Cùng với đó, chúng em xin được gửi cảm ơn đến các bạn cùng khóa đã cung cấp nhiều thông tin và kiến thức hữu ích giúp chúng em có thể hoàn thiện hơn đề tài của mình.

Đề tài và bài báo cáo được chúng em thực hiện trong khoảng thời gian ngắn, với những kiến thức còn hạn chế cùng nhiều hạn chế khác về mặt kĩ thuật và kinh nghiệm trong việc thực hiện một dự án phần mềm. Do đó, trong quá trình làm nên đề tài có những thiếu sót là điều không thể tránh khỏi nên chúng em rất mong nhận được những ý kiến đóng góp quý báu của các quý thầy cô để kiến thức của chúng em được hoàn thiện hơn và chúng em có thể làm tốt hơn nữa trong những lần sau. Chúng em xin chân thành cảm ơn.

Cuối lời, chúng em kính chúc quý thầy, quý cô luôn dồi dào sức khỏe và thành công hơn nữa trong sự nghiệp trồng người. Một lần nữa chúng em xin chân thành cảm ơn.

TP. Hồ Chí Minh, ngày . . . tháng 12 năm 2022

Nhóm sinh viên thực hiện

PHẦN I: ĐẠO VĂN

1. Định nghĩa về đạo văn

Đạo văn có thể hiểu đơn giản là hành động cố ý hoặc vô tình lấy, sao chép thành quả, tài liệu, nghiên cứu của người khác và xem như đó là của mình.

Theo định nghĩa được đưa ra trong Từ điển bách khoa toàn thư của New Webster về ngôn ngữ tiếng Anh năm 1997 (the 1997 New Webster's Encyclopedic Dictionary of the English Language), đạo văn là hành vi sử dụng trái phép ngôn ngữ và suy nghĩ của một tác giả khác và thể hiện chúng như là của riêng bạn.

2. Những trường hợp đạo văn phổ biến

Đạo văn không chỉ đơn thuần là sao chép một tài liệu của người khác và xem như đó là của mình. Đạo văn có rất nhiều biến hóa khác được tóm tắt sơ lược như sau:

- Clone (Bản sao): Sao chép toàn bộ tài liệu, nghiên cứu, chất xám của tác giả.
- Copy: Sao chép một phần hoặc toàn bộ tài liệu, nghiên cứu và chất xám của tác giả nhưng có chỉnh sửa, thay đổi nhỏ nội dung nhưng không trích dẫn nguồn.
- Remix: Vừa sao chép nội dung từ nhiều nguồn, vừa thay đổi nội dung để tạo sự mạch lạc và không trích dẫn nguồn cụ thể.
- Hybrid: Kết hợp giữa tài liệu không trích nguồn và tài liệu trích nguồn để tránh việc đạo văn
- Aggregator: Giống như Clone nhưng Aggregator có trích dẫn nguồn, tuy nhiên việc sao chép toàn bộ vẫn không được khuyến khích.
- Mashup: Trộn lẫn những nội dung từ nhiều nguồn khác nhau để tạo ra một tài liệu hoàn chỉnh.

3. Cách phòng tránh đạo văn

Với việc các nguồn thông tin ngày càng dồi dào, cùng với việc được tiếp xúc với lượng kiến thức đồ sộ từ nhiều nguồn khác nhau, việc tránh đạo văn thường sẽ rất khó khăn đối với sinh viên. Nhưng không vì vậy mà chúng ta lại sử dụng lí do đó để bao dung cho hành động này. Để tránh đạo văn khi sử dụng, trích dẫn tài liệu của một tác giả, nguồn thông tin nào đó, chúng ta phải luôn đảm bảo việc trích dẫn và ghi rõ nguồn cụ thể. Tuy vậy, nên tránh việc sao chép quá nhiều dù cho có trích dẫn cụ thể, việc làm này vẫn được xem là hình thức của việc đạo văn. Chỉ nên sử dụng việc trích dẫn, lấy ý tưởng như là một công cụ hỗ trợ giúp cho chúng ta trong việc đưa ra quan diểm cá nhân

4. Lời cảm kết

Chúng em xin cam đoan đồ án này do các thành viên trong nhóm thực hiện. Chúng em không sao chép, sử dụng bất kỳ tài liệu, mã nguồn… của người khác mà không ghi nguồn. Chúng em xin chịu hoàn toàn trách nhiệm nếu vi phạm đạo văn.

PHẦN II: LÝ THUYẾT

1. Ngôn ngữ lập trình python

1.1. Python là gì?

Python là một ngôn ngữ lập trình được sử dụng trong việc phát triển các ứng dụng web, phần mềm, máy học và khoa học dữ liệu.

Những lợi ích mà Python mang lại:

- Các nhà phát triển có thể dễ dàng đọc và hiểu một chương trình Python vì ngôn ngữ này có cú pháp cơ bản giống tiếng Anh.
- Python giúp cải thiện năng suất làm việc của các nhà phát triển vì so với những ngôn ngữ khác, họ có thể sử dụng ít dòng mã hơn để viết một chương trình Python.
- Python có một thư viện tiêu chuẩn lớn, chứa nhiều dòng mã có thể tái sử dụng cho hầu hết mọi tác vụ. Nhờ đó, các nhà phát triển sẽ không cần phải viết mã từ đầu.
- Các nhà phát triển có thể dễ dàng sử dụng Python với các ngôn ngữ lập trình phổ biến khác như Java, C và C++.
- Cộng đồng Python tích cực hoạt động bao gồm hàng triệu nhà phát triển nhiệt tình hỗ trợ trên toàn thế giới. Nếu gặp phải vấn đề, bạn sẽ có thể nhận được sự hỗ trợ nhanh chóng từ cộng đồng.
- Trên Internet có rất nhiều tài nguyên hữu ích nếu bạn muốn học Python. Ví dụ: bạn có thể dễ dàng tìm thấy video, chỉ dẫn, tài liệu và hướng dẫn dành cho nhà phát triển.
- Python có thể được sử dụng trên nhiều hệ điều hành máy tính khác nhau, chẳng hạn như Windows, macOS, Linux và Unix.

Python được sử dụng trong những lĩnh vực như:

- Phát triển web phía máy chủ
- Tự động hóa các tập lệnh Python
- Khoa học dữ liệu và máy học
- Phát triển phần mềm
- Tự động hóa kiểm thử phần mềm

Những đặc điểm của Python:

- Python là một ngôn ngữ thông dịch
- Python là một ngôn ngữ dễ sử dụng
- Python là một ngôn ngữ linh hoạt
- Python là một ngôn ngữ cấp cao

- Là một ngôn ngữ lập trình hướng đối tượng

Một số thư viện phổ biến của Python: Matplotlib, Pandas, Numpy, Requests, OpenCV, Keras

SDK Python là một tập hợp các công cụ phần mềm mà các nhà phát triển có thể sử dụng để tạo ra những ứng dụng phần mềm bằng một ngôn ngữ cụ thể.

1.2. Cú pháp trong python

Python là một ngôn ngữ dễ đọc, dễ hiểu. Định dạng của nó rất gọn gàng về mặt trực quan, và nó thường sử dụng các từ khoá tiếng Anh trong khi các ngôn ngữ khác lại sử dụng các dấu câu.

Python sử dụng thụt lề bằng khoảng trắng hoặc ký tự tab thay vì dùng ngoặc nhọn hay các từ khoá để giới hạn khối lệnh. Lề thường được thụt vào sau một câu lệnh và thụt ra để đánh dấu kết thúc khối lệnh hiện tại.

Một số câu lệnh trong python:

- Dấu = là một câu lệnh gán
- Câu lệnh if (if – else) thực thi khối lệnh nếu thỏa mãn điều kiện
- Câu lệnh for lặp qua các đối tượng, gán mỗi phần tử và một biến cục bộ để sử dụng trong khối lệnh của vòng lặp.
- Câu lệnh while thực thi khối lệnh khi điều kiện đúng
- Câu lệnh try thực thi và bắt các ngoại lệ (except), dọn dẹp trong (finally).
- Câu lệnh break thoát ra khỏi vòng lặp
- Câu lệnh class thực thi một khối lệnh và gắn không gian tên cục bộ của nó vào một lớp, để dùng trong lập trình hướng đối tượng.
- Câu lệnh def định nghĩa hàm hoặc một phương thức
- Câu lệnh return trả lại một giá trị từ một hàm hoặc một phương thức nào đó
- Câu lệnh import nhập các module khác nhau
- ...

1.3. Kiểu dữ liệu

Một số kiểu dữ liệu trong python là: bool, bytearray, bytes, complex, dict, float, frozenset, int, list, str, tuple, range, set, NoneType,…

2. Tìm hiểu về AI (Artificial Intelligence)

2.1. AI là gì?

AI hay **trí truệ nhân tạo** là một nghành kỹ thuật khoa học chế tạo máy móc thông minh, các chương trình máy tính thông minh.

AI được thực hiện bằng cách nghiên cứu cách suy nghĩ của con người, cách con người học hỏi, quyết định và làm việc trong khi giải quyết một vấn đề nào đó sau đó sử dụng lại kết quả nghiên cứu này từ đó tìm ra các giải pháp, nền tảng phát triển các phần mềm thông minh, hệ thống thông minh.

Mục đích của AI:

- Tạo ra hệ thống có thể hiểu, suy nghĩ và học hỏi từ con người
- Tạo ra các hệ thống chuyên gia – các hệ thống này giúp giải quyết được vấn đề ở một vấn đề phức tạp cụ thể nào đó.

AI được áp dụng vào nhiều lĩnh vực trong đời sống như Khoa học máy tính, Toán học, Sinh học, kỹ thuật,…

Trong thực tế, chúng ta nhận thấy rằng các nguồn dữ liệu có một số tính chất không mong muốn như khối lượng rất lớn, không có định dạng hay cấu trúc đủ tốt và dữ liệu thay đổi liên tục → Kỹ thuật AI được biết đến như là một cách tổ chức và sử dụng kiến thức để khắc phục những tình trạng không tốt của dữ liệu.

2.2. Một số ứng dụng của AI

- Quản trị: Các hệ thống AI trợ giúp các công việc hành chính hàng ngày, để giảm thiểu lỗi của con người và tối đa hóa hiệu quả.
- Điều trị từ xa: Đối với các tình huống không khẩn cấp, bệnh nhân có thể liên hệ với hệ thống AI của bệnh viện để phân tích các triệu chứng của họ, nhập các dấu hiệu quan trọng của họ và đánh giá xem có cần phải chăm sóc y tế hay không. Điều này làm giảm khối lượng công việc của các chuyên gia y tế bằng cách chỉ đưa các trường hợp quan trọng đến họ.
- Hỗ trợ chuẩn đoán: Thông qua thị giác máy tính và mạng lưới thần kinh tích chập, AI hiện có khả năng đọc quét hình ảnh cộng hưởng từ để kiểm tra khối u và sự phát triển ác tính khác của nó, với tốc độ nhanh hơn so với các bác sĩ x-quang và sai số thấp hơn đáng kể.
- Phẫu thuật có sự trợ giúp của robot: Robot phẫu thuật có sai số rất nhỏ và có thể thực hiện phẫu thuật suốt ngày đêm mà không bị kiệt sức.
- Giám sát các chỉ số quan trọng
- Ngoài ra còn rất nhiều những ứng dụng trong các lĩnh vực khác trong đời sống như nhận diện khuôn mặt, nhận diện giọng nói, ô tô tự lái…

3. Tìm hiểu về Machine Learning

3.1. Machine Learning là gì?

Machine Learning là một ứng dụng của trí tuệ nhân tạo (AI), nhờ vào Machine Learning hệ thống trí tuệ nhân tạo có khả năng tự học hỏi và cải thiện kinh nghiệm mà

không cần phải lập trình rõ ràng, chi tiết. Machine Learning được cho là thường tập trung vào các chương trình máy tính có thể truy cập dữ liệu và dùng nó để tự học.

3.2. Mục tiêu của Machine Learning?

Mục tiêu của machine learning được cho là một công cụ giúp máy tính dễ dàng tìm hiểu về quy trình một cách tự động khi không có sự tác động nào từ phía con người hoặc sẽ hỗ trợ điều chỉnh hành động hiệu quả hơn. Các thuật toán machine learning này được giám sát để dễ dàng cho việc áp dụng và tìm hiểu về dữ liệu mới trong quá khứ với ví dụ được gắn nhãn để dự đoán các sự kiện tương lai.

3.3. Các phương pháp Machine Learning

- **Supervised Learning (học tập có giám sát):** chúng ta có thể hiểu là máy tính sẽ được cung cấp các ví dụ (data) đầu vào được gắn nhãn với đầu ra mong muốn của chúng. Mục đích của phương pháp này là thuật toán có thể học (được dạy học) bằng cách so sánh kết quả đầu ra với kết quả thực tế - tìm và sửa lỗi, sửa đổi mô hình cho phù hợp.
- **Unsupervised Learning (học tập không giám sát):** chúng ta có thể hiểu dữ liệu ở học tập không giám sát không cần gắn nhãn cho nên thuật toán học được để lại điểm chung giữa các dữ liệu đầu vào. Mục tiêu của học tập loại này có thể đơn giản như là khám phá các mẫu ẩn trong tập dữ liệu, bên cạnh đó, nó cũng có thể có mục tiêu học tập tính năng, cho phép máy tính tự động khám phá các biểu diễn cần thiết để phân loại dữ liệu.

4. Tìm hiểu về Deep Learning

4.1. Deep learning (học sâu) là gì?

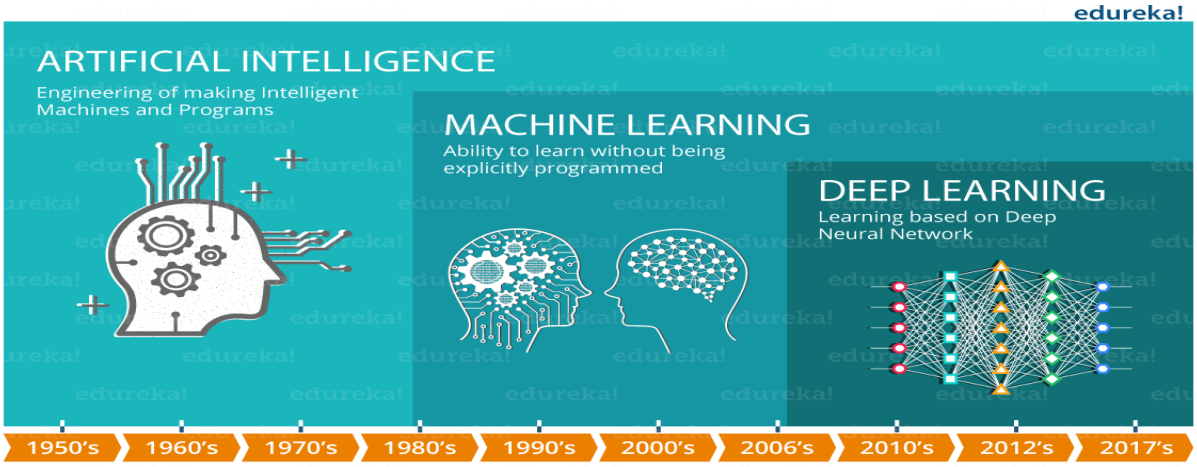
Deep learning hay học sâu được biết đến như là một lĩnh vực con của học máy nó thường liên quan đến các thuật toán lấy cảm hứng từ cấu trúc và chức năng của não và được gọi là mạng thần kinh nhân tạo (artificial neural networks).[1]

Theo **Wikipedia,** các kiến trúc học sâu như mạng nơ-ron sâu, mạng niềm tin sâu, học tăng cường sâu, mạng nơ-ron lặp lại, mạng nơ-ron phức hợp và máy biến áp đã được áp dụng cho các lĩnh vực bao gồm thị giác máy tính, nhận dạng giọng nói, xử lý ngôn ngữ tự nhiên, dịch máy, tin sinh học, thiết kế thuốc , phân tích hình ảnh y tế, khoa học khí hậu, kiểm tra vật liệu và các chương trình trò chơi trên bàn cờ, nơi chúng đã tạo ra kết quả tương đương và trong một số trường hợp vượt qua hiệu suất của chuyên gia con người.

[1] Deep learning (học sâu) là gì? Link: https://daco.vn/san-pham/cong-nghe-deep-learning-hoc-sau-la-gi-ung-dung-thuc-te-va-moi-lien-he-giua-hoc-sau-hoc-may-va-tri-tue-nhan-tao-7903

Mạng nơ-ron nhân tạo (ANN) được lấy cảm hứng từ quá trình xử lý thông tin và các nút giao tiếp phân tán trong các hệ thống sinh học. ANN có nhiều điểm khác biệt so với bộ não sinh học. Cụ thể, mạng lưới thần kinh nhân tạo có xu hướng tĩnh và tượng trưng, trong khi bộ não sinh học của hầu hết các sinh vật sống là động (nhựa) và tương tự

4.2. Lịch sử phát triển của học sâu? [1] [2] [3]



Hình 1: Lịch sử phát triển

(src: https://www.edureka.co/blog/wp-content/uploads/2017/05/AI-Timeline-What-is-Deep-Learning-Edureka-1.png)

Khái niệm học sâu lần đầu tiên được đề cập bởi Rina Dechter trong một công trình công bố năm 1986. Đến năm 1989, Lecun và cộng sự đã công bố kết quả đề xuất một kiến trúc mạng nơ ron học sâu (gọi là LeNet) áp dụng các thuật toán truyền ngược tiêu chuẩn nhằm xử lý nhận dạng chữ viết và kết quả thực nghiệm đạt độ chính xác cao.

Năm 2006, Geoffrey Hinton công bố một cách thức huấn luyện mạng nơ ron nhiều lớp mới được gọi là mạng học sâu niềm tin DBN. DBN đã vượt qua tất cả các thuật toán học máy khác trong việc phân loại chính xác bộ chữ số viết tay MNIST.

Năm 2012, cũng tại một cuộc thi thường niên có tên ImageNet Large Scale Visual Recognition Challenge - ILSVRC, mạng AlexNet đạt kết quả tốp 5 đánh giá theo chỉ số lỗi (16%). Mạng AlexNet có kiến trúc mạng tương tự với LeNet nhưng sử dụng một số lượng các lớp, số bộ lọc và số nơ ron lớn hơn rất nhiều. Sau AlexNet, tất cả các mô hình giành giải cao trong các năm tiếp theo đều là các mạng học sâu. Với những thành công này, học sâu trở thành một lĩnh vực nghiên cứu được đặc biệt quan tâm trong trí tuệ nhân tạo, khoa học máy tính.

Là một xu hướng nóng trong công nghệ thông tin, học sâu không những là chủ đề được cộng đồng nghiên cứu khoa học máy tính quan tâm hàng đầu mà đã vượt ra khuôn khổ của các phòng, dự án nghiên cứu, để trở thành công nghệ được ứng dụng trong thực

tiễn. Một số ứng dụng nổi bật của học sâu có thể kể đến: Trợ lý ảo (Alexa, Siri, Cortana). dịch thuật, chatbots, thiết bị không người lái, nhận dạng đối tượng, xác định danh tính người (qua khuôn mặt, hình dáng). các hệ thống không người lái, chẩn đoán bệnh và các hệ thống hỗ trợ y tế, gian lận điện tử, thương mại điện tử và cá nhân hóa người dùng,...

Những năm gần đây, kỹ thuật học sâu đang trở thành một trong những lĩnh vực được quan tâm nghiên cứu và ứng dụng đặc biệt trong lĩnh vưc khoa học máy tính. Kỹ thuật học sâu đã đạt được những kết quả khả quan với độ chính xác vượt trội so với cách tiếp cận truyền thống, đồng thời thúc đẩy tiến bộ trong đa lĩnh vực như nhận dạng đối tượng, dịch tự động, nhận dạng giọng nói, các trò chơi thông minh và những bài toán khó trong trí tuệ nhân tạo.

BusinessWire, thuộc tập đoàn Bershire Hathaway, dự đoán thị trường liên quan đến công nghệ học sâu trên toàn cầu dự tính đạt khoảng 4 tỉ USD vào năm 2025.

Các hãng công nghệ tập trung đầu tư và được hưởng lợi nhuận nhiều nhất trong việc bán các sản phẩm hỗ trợ học sâu có thể kế đến NVIDIA (chuyên về sản xuất GPU). Google (sử dụng học sâu trong các công cụ tìm kiếm, phân tích thông tin). Amazon (thương mại điện tử) …

Nhiều nhà khoa học trong lĩnh vực Khoa học máy tính chuyển sang các tập đoàn công nghệ lớn trong đó có Geoffrey Hinton (Google). Yann Lecunn (Phó chủ tịch, giám đốc AI của Facebook). Andrew Ng (Baidu) … đều xuất thân là các nhà nghiên cứu chuyên sâu, tiên phong về công nghệ học sâu.

Những thông tin và phân tích trên cho thấy học sâu vẫn sẽ là xu hướng phát triển nóng của ngành công nghệ thông tin, tiếp tục nhận được sự đầu tư lớn từ các tập đoàn công nghệ.

Các chuyên gia trí tuệ nhân tạo và học sâu đều có nhận định rằng để phát triển tốt lĩnh vực này trong cả nghiên cứu lẫn công nghiệp, vấn đề quan trọng là hình thành các cơ sở dữ liệu đủ lớn và đủ tốt dùng trong huấn luyện các mô hình học sâu. Những cơ sở dữ liệu lớn như vậy về ảnh y tế, tiếng nói, tín hiệu điện tim, điện não, ảnh giao thông… đang dần được xây dựng bởi các tập đoàn công nghệ, cộng đồng nghiên cứu trong các trường, viện nghiên cứu dưới sự bảo trợ của Chính phủ.

4.3. Các kiến trúc của học sâu? [1] [2] [3]

Có một số kiến trúc mạng nơ ron trong học sâu.

- Mạng nơ ron sâu (Deep Neural Network - DNN) là một dạng cụ thể của lĩnh vực học sâu. Mạng nơ ron sâu là một mạng nơ ron nhân tạo nhưng có kiến trúc phức tạp và "sâu" hơn nhiều so với kiến trúc của mạng nơ ron truyền thống (mạng nơ ron nông). Nghĩa là nó có số nút trong mỗi lớp và số lớp ẩn lớn hơn rất nhiều và cách thức hoạt động của nó phức tạp hơn so với kiến trúc mạng nơ ron truyền thống.
    - Mạng nơ-ron tích chập. Mạng nơ ron tích chập (Convolutional neural network - CNN) là một dạng cụ thể của mạng nơ ron sâu. Mạng nơ ron tích chập có một lớp vào, một lớp ra và nhiều lớp ẩn khác nhau. Các lớp ẩn gồm các loại như: lớp tích chập (convolution). lớp giảm kích thước (pooling). lớp sửa dữ liệu (ReLU). lớp chuẩn hóa (normalization). lớp kết nối đầy đủ (full connection)... Trong đó, lớp tích chập được sử dụng nhằm tạo liên kết giữa các lớp liền kề trong phạm vi nhỏ, giới hạn trong "vùng" cục bộ. Điều này giúp giảm đáng kể các việc tính toán các hàm truyền giữa các lớp mà vẫn duy trì được mối liên hệ giữa các nơ ron để trích xuất đặc trưng của dữ liệu ở các lớp sau đó của mạng.
- Mạng học sâu niềm tin. Mạng học sâu niềm tin (Deep belief net-DBN) là một mô hình mạng nơ-ron nhân tạo nhiều lớp. Quá trình huấn luyện mạng DBN gồm hai pha: tiền huấn luyện (pre-training) và hiệu chỉnh trọng số (fine-tuning). Trong pha tiền huấn luyện, máy học Boltzman được sử dụng để khởi tạo trọng số tốt nhất cho mô hình với dữ liệu không cần được gán nhãn. Trong pha tiếp theo hiệu chỉnh trọng số, DBN tiếp tục được huấn luyện bằng phương pháp lan truyền ngược cổ điển với dữ liệu được gán nhãn.
- Mạng tự mã hóa. Để huấn luyện mạng nơ-ron thường sử dụng học có giám sát, trong đó sử dụng các tập mẫu có gán nhãn. Mạng nơ-ron tự mã hóa thưa (sparse autoencoder) là một thuật toán học không giám sát sử dụng thuật toán lan truyền ngược, đặt giá trị đầu ra bằng với đầu vào trên dữ liệu không gán nhãn.

Một số thư viện liên quan đến học sâu:

- Tensor Flow: là thư việc mã nguồn mở được xây dựng và phát triển bởi Google Brain. Thư việc sử dụng đồ thị luồng dữ liệu (data flow graphọc sâu) để tính toán, hỗ trợ API cho python, C++… TensorFlow hỗ trợ các nền tảng lập trình trên hầu hết các hệ điều hành phổ biến như Linux, macOS, Windows, Android và iOS.
- Caffe: là nền tảng học sâu được phát triển bởi Berkeley AI Research và cộng đồng. Nó hỗ trợ nhiều loại kiến trúc học sâu khác nhau. Caffe hỗ trợ các thư viện có tốc độ tính toán nhanh toán, hoạt động tốt trên GPU và CPU.
- Torch: là framework hỗ trợ các tính toán khoa học, thuật toán học máy dựa trên ngôn ngữ Lua. Torch được hỗ trợ phát triển bởi Facebook, Google, DeepMind, Twitter…

- Pytorch: là thư viện học máy mã nguồn mở, đặc biệt mạng học sâu thực thi trên GPU, được phát triển bởi Facebook. PyTorch được viết bằng Python, C và CUDA.
- Keras: là thư viện về mạng học sâu được viết bằng Python, thư viện có thể thực thi trên GPU và CPU. thân thiện với người dùng.

4.4. Ứng dụng của học sâu

Một số ứng dụng của học sâu:

- Xử lý ngôn ngữ tự nhiên (Nature Language Processing)
- Nhận dạng hình ảnh
- Khám phá dược phẩm và độc chất học
- Quản lý quan hệ khách hàng (CRM)
- Các hệ thống khuyến cáo (gợi ý)
- Tin sinh học

PHẦN III: PHÂN TÍCH HỆ THỐNG

1. Tìm hiểu về trợ lý ảo

1.1. Tổng quan về trợ lý ảo

Chúng ta biết rằng thông thường có ba cách để truyền tải thông điệp hay thông tin của mình đến người khác đó chính là viết, dùng hành động hoặc là dùng lời nói để truyền tải thông tin. Vậy, rõ ràng chúng ta thấy rằng việc "nói" chính là cách truyền tải thông tin tiết kiệm công sức nhất.

Ngày nay với sự phát triển nhanh chóng của công nghệ nói chung và AI cũng như máy học nói riêng đã giúp con người rất nhiều trong việc phát triển công nghệ nhận dạng giọng nói, công nghệ này càng đi sâu vào mọi lĩnh vực trong cuộc sống. Từ những điều này chúng ta sẽ cùng nhau phân tích, chứng minh tại sao công nghệ điều khiển giọng nói hay thực hiện các chuỗi hành động thông qua giọng nói được coi là xu thế công nghệ hiện tại không chỉ bởi các ông lớn về IT hay những người có niềm đam mê với AI mà các doanh nghiệp, tập đoàn lớn họ cũng đang tìm cách phát triển, tối ưu hệ thống của mình bằng giọng nói hay "**Trợ lý ảo**".

1.2. Trợ lý ảo là gì?

Trợ lý ảo (có thể được gọi là trợ lý kỹ thuật số, trợ lý giọng nói hay là trợ lý AI) là một ứng dụng lập trình hướng nhiệm vụ, nhận dạng giọng nói của con người và thực hiện các lệnh được phát âm bởi người dùng. Nền tảng của nó là AI và năng suất của nó dựa vào việc lưu trữ hàng triệu từ và hàng triệu cụm từ. Vào những năm trước đây chúng ta có hai loại phần mềm trợ lý giọng nói phổ biến:

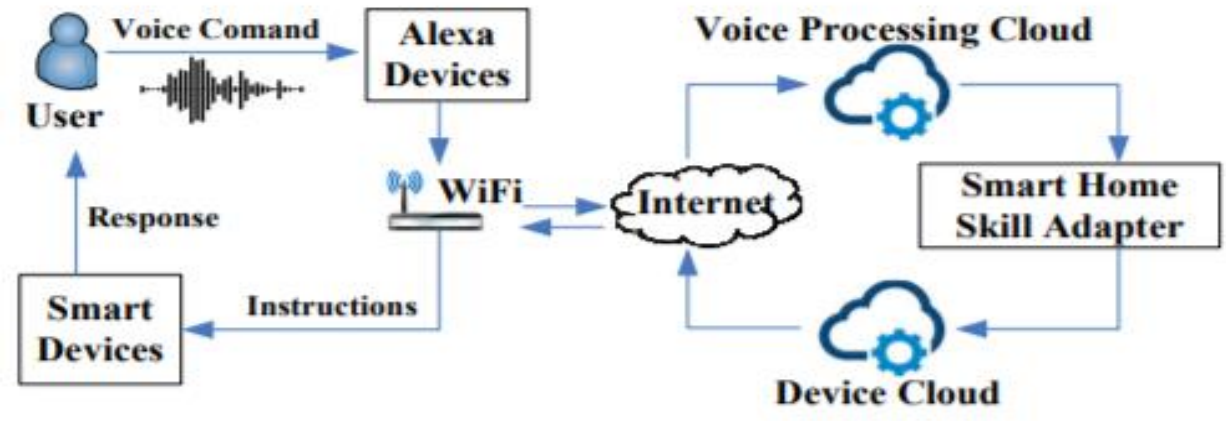
- Trợ lý ảo thông minh tại nhà tương tự như việc chúng ta có thể sử dụng giọng nói để điều khiển ngôi nhà của mình như bật đèn, bật nhạc, bật lò sưởi và nhiều thiệt bị điện tử khác được lắp trong ngôi nhà của chúng ta. Việc này đều phải đòi hỏi kết nối internet và một phần của Internet vạn vật.
- Trợ lý ảo thông minh tại cơ sở làm việc.

Ngoài ra ngày nay chúng ta còn thấy nhiều loại phần mềm trợ lý giọng nói phổ biến như trợ lý lái xe, trợ lý công việc,…

Ngày nay có hai ứng dụng công nghệ trợ lý ảo lớn được phát triển bởi Apple (Siri) và Google (Google Assistant) là 2 ứng dụng trợ lý ảo được rất nhiều người biết đến và sử dụng rộng rãi nhất hiện nay. Thêm vào đó không quá xa lạ đến với những người dùng sử dụng hệ điều hành Windows đó là trợ lý ảo Microsoft Cortana.

1.3. Cấu trúc hệ thống trợ lý ảo

Để hiểu một hệ thống trợ lý ảo hoạt động như thế nào, cần phải hiểu được cấu trúc và cấu tạo của một hệ thống trợ lý ảo. Sơ đồ của một hệ thống trợ lý ảo: (lấy ví dụ trợ lý ảo Alexa)



Hình 2: Sơ đồ chung cho hệ thống trợ lý ảo

(src: https://i.imgur.com/L1h9fQX.png)

Dựa theo hình, có thể thấy cấu tạo chung của trợ lý ảo ngày nay gồm 2 khối: khối người dùng (user) có đầu vào là giọng nói (voice) và khối xử lý dữ liệu (server) gồm các khối xử lý dữ liệu âm thanh, ngoài ra trong một số phần mềm trợ lý ảo khác nhau còn có thể nhận hình ảnh, văn bản và khối xử lý dữ liệu âm thanh, văn bản.

Truy vấn bắt đầu bằng giọng nói, hình ảnh, văn bản của người dùng thiết bị thông minh như điện thoại, đồng hồ thông minh, thiết bị đeo, kính thông minh. Các file nén âm thanh, hình ảnh hoặc văn bản hay đại loại được gửi tới máy chủ để xử lý.

Tại đây giọng nói của chúng ta sẽ được xử lý bằng giao diện ASR (Automatic Speech Recognition) chuyển câu thoại của người dùng sang văn bản tương đương bằng mô hình thống kê. Sau đó văn bản này sẽ đi qua trình phân loại truy vấn (Query Classifier -QC) và nhờ đó để quyết định xem câu thoại đó có phải là câu hỏi hoặc hành động nào đó hay không. Nếu là hành động, lệnh sẽ được gửi lại cho thiết bị di động để được thực hiện. Nếu không thì hệ thống sẽ hiểu là câu hỏi bằng văn bản thuần túy.

Sử dụng các kỹ thuật xử lý ngôn ngữ tự nhiên NLP (Natural Language Process), dịch vụ trả lời câu hỏi QA (Question – Answer) sẽ nhận được output từ input của người dùng, tìm kiếm trong cơ sở dữ liệu từ đó đưa ra được output tốt nhất.

1.4. Cách thức hoạt động của trợ lý ảo (nhận dạng giọng nói ASR)

Các ứng dụng trợ lý giọng nói hoạt động dựa trên hệ thống Nhận dạng giọng nói tự động (ASR – Automatic Speech Recognition). Các đầu vào cho ASR [4] là các vector đặc trưng đại diện cho các đoạn hội thoại, được tạo ra bởi quá trình tiền xử lý nhanh và trích xuất đặc tính của bài nói. Thành phần của ASR dựa vào sự kết hợp của mô hình Hidden Markov (HMM) và một mô hình hỗn hợp Gaussian (GMM) hoặc một mạng nơ-ron sâu (DNN).

Chúng ta có thể thấy hầu như mọi phần mềm trợ lý ảo ngày nay đều sử dụng ASR. Điều quan trọng là chúng ta cần làm quen với cách ASR hoạt động. Có thể nói ngắn gọn quá trình này, quá trình bắt đầu từ việc thu âm thành từ các thiết bị thu âm thanh. Các dạng song giọng nói đã được ghi lại chuyển sang phân tích, được thực hiện ở ba cấp độ khác nhau:

- Mô hình âm thanh, đại diện cho những âm vị được phát âm và những từ mà các âm vị này hoàn thành là gì;
- Mô hình phát âm, phân tích cách phát âm của âm vị, có bất kỳ trọng âm hoặc đặc thù nào khác của bộ máy phát âm để nắm bắt sự biến đổi ngữ âm của lời nói;
- Mô hình hóa ngôn ngữ, nhằm mục đích tìm kiếm xác suất theo ngữ cảnh tùy thuộc vào âm vị nào được ghi lại.

Dữ liệu được xử lý bằng AI, giảm tỷ lệ xuất hiện lỗi bằng cách sử dụng các modal trong học máy. Dữ liệu được chuyển sang giọng nói và truyền đến bộ giải mã, nơi cuối cùng nó chuyển thành văn bản để sử dụng thêm như lệnh hoặc chính tả.

1.5. Ứng dụng thực tế của trợ lý ảo ngày nay

Ngày nay chúng ta có thể thấy trợ lý ảo trong các lĩnh vực như:

- Gửi thông tin cập nhật về các chủ đề mà bạn quan tâm mà không cần bạn tìm kiếm chúng;
- Dự báo thời tiết;
- Thêm các sự kiện và cuộc họp vào lịch của một nhóm hoặc từng thành viên riêng biệt;
- Đặt báo thức và nhắc nhở mọi việc thứ diễn ra theo đúng lịch trình;
- Trả lời câu hỏi chung bằng giọng nói (thay vì mở liên kết để bạn tìm kiếm câu trả lời);
- Tạo và điền vào danh sách To-do list;
- Thực hiện dịch thuật thời gian thực;

- Cập nhật cho bạn về lưu lượng trên lộ trình của bạn (đặc biệt hữu ích cho các hoạt động hậu cần);
- Theo dõi hàng tồn kho trong kho và tự động điền vào danh sách mua sắm với các mặt hàng sẽ được đưa ra ngoài;
- Điều khiển các thiết bị khác từ ánh sáng đến PC;
- Đọc email và các tài liệu khác thành tiếng;
- Ghi lại lời nói chính tả và chuyển nó thành văn bản thay vì gõ thủ công;

Trợ lý ảo hay những lợi ích của nó đối với cuộc sống ngày nay là một điều không thể bàn cãi. Qua đây ta có thể thấy rằng Lĩnh vực trí tuệ nhân tạo đang và sẽ trở thành một trong những nền tảng cốt lõi của thời đại 4.0 hiện nay.

2. Trợ lý ảo sử dụng giọng nói

Trợ lý ảo giọng nói (Voice Assistant – VA) có thể nói là một công cụ, một người bạn khá quen thuộc với chúng ta ngày nay (Iphone có Siri – Android Samsung có Google Assistant) và cả Alexa ở hình ảnh trên cũng là một ví dụ về trợ lý ảo sử dụng giọng nói. Chỉ với một câu nói hay mệnh lệnh, trợ lý giọng nói này giúp chúng ta thực hiện các tác vụ mà chúng ta mong muốn hoặc đơn giản chỉ đưa ra 1 hoặc nhiều câu phản hồi giống như chúng ta đang trò chuyện với một người.

3. Quy trình trợ lý giọng nói hoạt động

- **Trợ lý giọng nói nghe và nhận lệnh từ người dùng:** sau khi nghe được câu gọi khởi động (hoặc nút khởi động) từ người dùng, trợ lý ảo sẽ được kích hoạt bắt đầu phản ứng. Nó bắt đầu tiếp nhận các yêu cầu của người dùng và tiến hành xử lý chúng.
- **Nhận dạng tiếng nói (ASR):** chúng ta có thể hiểu đây là giai đoạn sử dụng AI và học sâu để bắt đầu quá trình chuyển đổi các song âm thanh thành dữ liệu mà máy có thể hiểu được. Các yếu tố ở giai đoạn này bao gồm các đặc tính của tín hiệu tiếng nói như tần số, năng lượng, trường độ,…
- **Xử lý ngôn ngữ tự nhiên (Natural Language Processing):**
    - **Xử lý ngôn ngữ tự nhiên trong quản lý trả lời câu hỏi:**

Theo Wikipedia, NLP (Natural languages processing) là một nhánh của trí tuệ nhân tạo tập trung vào các ứng dụng trên ngôn ngữ con người. Trong trí tuệ nhân tạo thì ngôn

ngữ tự nhiên là một trong những phần khó nhất vì nó liên quan đến việc phải hiểu ý nghĩa ngôn ngữ- công cụ hoàn hảo nhất của tư duy và giao tiếp.

Chúng ta có thể hiểu xử lý ngôn ngữ tự nhiên [5] là một phạm vi lý thuyết các kỹ thuật tính toán để phân tích và mô tả các văn bản xảy ra tự nhiên ở một hoặc nhiều mức độ phân tích ngôn ngữ theo yêu cầu của con người mong muốn.

Mục tiêu của NLP là nhằm thể hiện ý nghĩa thực sự và ý định của người dùng khi thao tác dữ liệu. Điển hình ứng dụng NLP:

- Giải thích văn bản đầu vào.
- Dịch văn bản sang một ngôn ngữ khác.
- Trả lời các câu hỏi về nội dung của một văn bản..
- Thu thập các suy luận từ văn bản.
    - **Các mức xử lý của ngôn ngữ tự nhiên (natural language process):**
        - **Ngữ âm học:** mức này có liên quan tới giải thích các âm thanh nói trong và giữa các từ. Có ba loại quy tắc thường được sử dụng trong phân tích âm vị học: quy tắc âm thanh trong từ, quy tắc âm trong biến thể phát âm khi từ được nói với nhau, quy tắc biến động trong ngữ điệu của một câu. Một hệ thống NLP hỗ trợ đầu vào nói, song âm là phân tích và mã hóa dữ liệu thành tín hiệu số hóa để giải thích các quy tắc khác nhau hoặc bằng việc so sánh với các ngôn ngữ cụ thể được sử dụng.
        - **Hình thái học:** liên quan tới bản chất cấu thành của từ bao gồm đơn vị nhỏ nhất của ý nghĩa. Ví dụ từ preregistratrion có thể được phân tích thành tiền tố, gốc “registra” và hậu tố. Ví ý nghĩa của mỗi hình thái vẫn được giữ nguyên qua các từ, con người có thể phân chia động từ không rõ thành các hình thái cấu thành để hiểu ý nghĩa của nó. Tương tự như trong NLP có thể nhận ra ý nghĩa.
        - **Từ vựng:** con người hay hệ thống NLP diễn giải ý nghĩa của từng từ.
        - **Thuật ngữ:** tập trung vào việc phân tích các từ trong một câu dê khám phá ra ngữ pháp câu trúc của câu. Điều này đòi hỏi cả ngữ pháp và trình độ phân tíchcú pháp. Kết quả của việc này là đại diện của một cu. Các môi quan hê phụ thuộc câu trúc giữa các từ. Có nhiêu ngù pháp khác nhau có thể được sử dụng và do dó sẽ ảnh huởng đến sự lựa chọn của một trình phân tích cú pháp. Ví dụ có câu "con chó đuổi con mèo" và "con mèo đuổi theo con chó" khác nhau về ý nghĩa.

- **Ngữ nghĩa:** mức độ mà hàu hết mọi người nghĩ rằng ý nghĩa được xác dịnh tuy nhiên chúng ta có thể xem trong xác dinh ở trên của các cấp, đó là tất cả các cấp có ý nghia góp phần vào. Xử lý ngữ nghĩa xác định ý nghĩa của một câu bằng cách tập trung vào tương tác giữa các ý nghĩa cấp từ trong câu. Mức độ này bao gồm việc định hướng ngữ nghĩa của các từ với nhiều giác quan, theo cách tương tự để cách phân định cú pháp của các từ có thể hoạt động như nhiều phần câu bài phát biểu là hoàn thành các cấp cú pháp. Ví du trong các nghĩa khác "file" là một danh từ có thê có nghĩa là một thư mục hoặc một công cụ để tạo.
- **Đàm luận:** mặc dù ngữ pháp và ngữ nghĩa làm việc với các đơn vị câu, mức độ diễn đạt của NLP làm việc với các đơn vị văn bản dài hơn một câu. Nghĩa là nó không giải thích văn bản như các câu ghép nối, mỗi câu có thể được giải thích đơn lẻ. Thay vào đó bài diễn thuyết tập trung vào tính chất của văn bản tập trung kết nối giữa các thành phần câu.
- **Thực dụng:** Liên quan đến việc sử dụng có mục đích ngôn ngữ trong các tình huống và sử dụng bối cảnh trên các nội dung văn bản để hiểu muc đích là để giải thích làm thế nào để thêm ý nghĩa được đọc vào văn bản. Điều này đòi hỏi nhiều kiến thức bao gồm sự hiểu về ý định, kế hoạch và những mục tiêu

Hệ thông NLP ngày nay có khuynh hướng thực hiện thành các mô dun để đạt được mức độ yêu cầu. Mức thấp thì sử dụng được mô-đun thấp, mức cao thì sử dụng được mô-đun kết hợp với nhau.

- **Hiểu ngôn ngữ (NLU):** chúng ta đều thấy rằng mỗi loại ngôn ngữ đều có cách sử dụng khác nhau và cả cách dùng khác nhau tùy vào mỗi cá nhân sử dụng ngôn ngữ vậy nên trợ lý giọng nói nhiều khi sẽ gặp khó khăn trong việc phân biệt các câu lệnh cần thực hiện. Xử lý ngôn ngữ tự nhiên là thao tác quan trọng (thứ mà đã nhắc đến ở trên) để giúp trợ lý giọng nói có thể hiểu được mệnh lệnh từ người dùng.
- **Truy xuất thông tin:** sau khi đã xử lý lệnh thông qua nhận dạng tiếng nói và đã hiểu được ngôn ngữ → truy xuất thông tin. Tại đây phần mềm, hệ thống bắt đầu truy cập vào các nguồn thông tin hay dữ liệu khác nhau để xây dựng câu trả lời và phản hồi theo đầu vào từ người dùng.

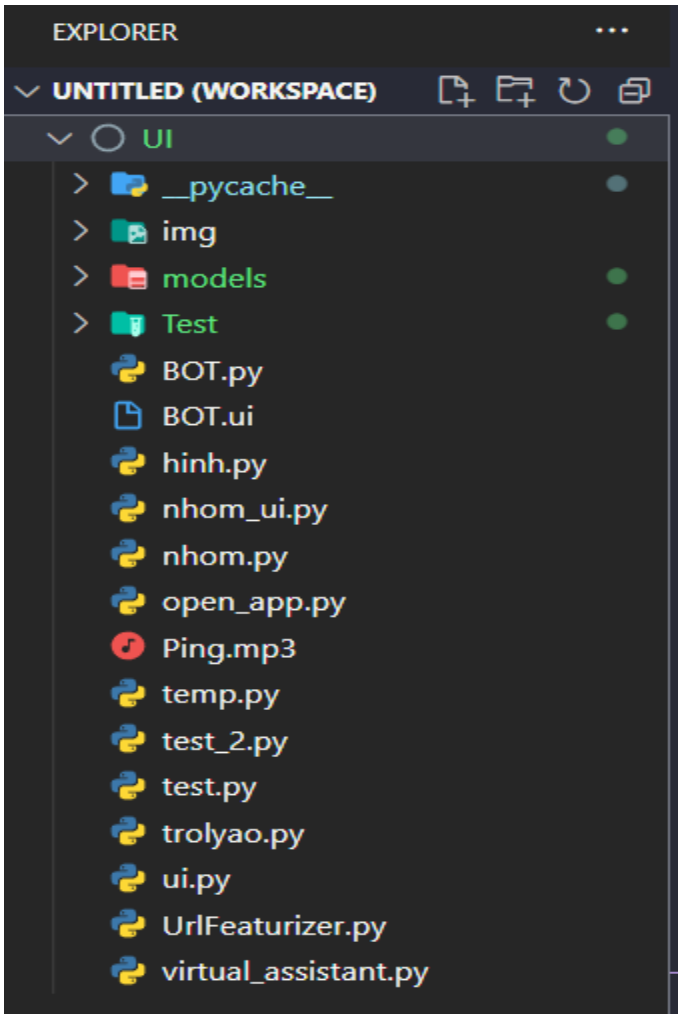
- **Trả ra câu trả lời và thực thi:** sau cùng người dùng sẽ nhận được phản hồi bằng âm thanh (hoặc cả âm thanh và đoạn text nội dung là câu trả lời) và đôi khi còn có cả các tác vụ hỗ trợ tích hợp tùy theo các tính năng của trợ lý giọng nói.

PHẦN IV: ỨNG DỤNG MINH HỌA

1. Cài đặt ứng dụng

Tải về project tại link github: https://github.com/Bao-Thanh/Virtual-assistant

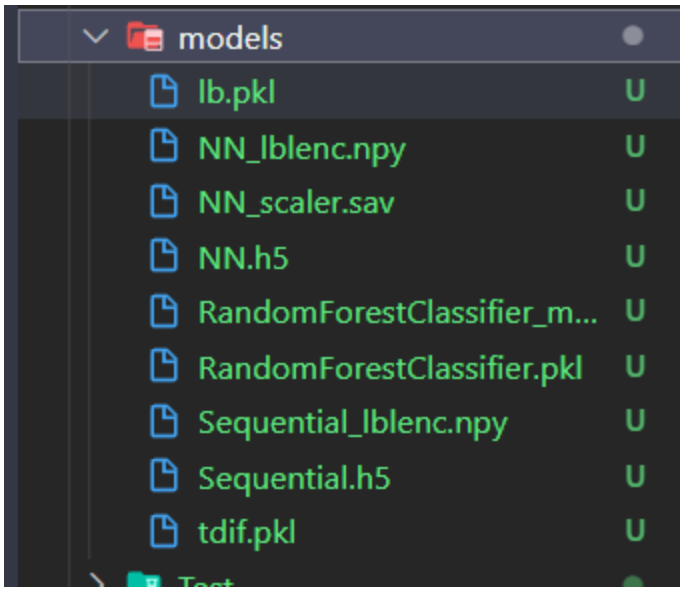
Khuyến khích dùng VSCode add thư mục UI vào workspace → tạo thêm thư mục models (cây thư mục sẽ như sau):



Hình 3: cây thư mục được hiển thị

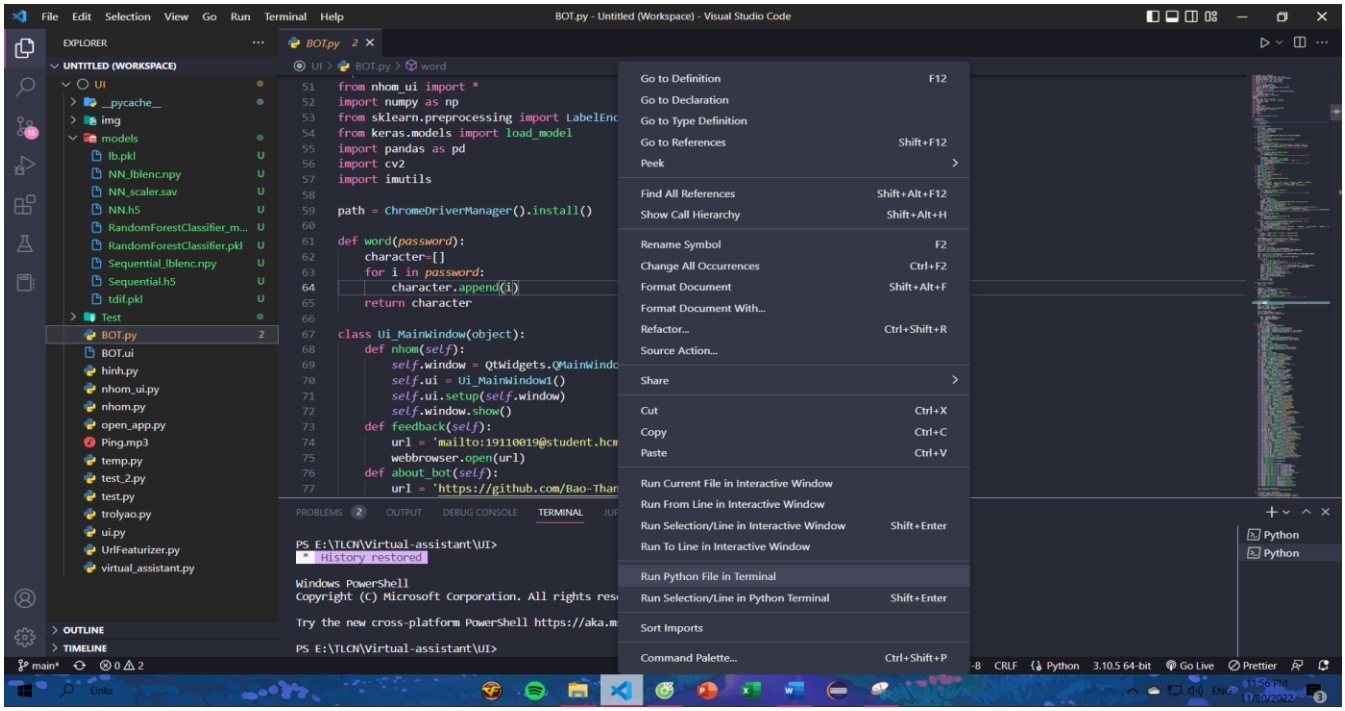
Tại link drive:
https://drive.google.com/drive/folders/1bqyoV8N_MyXW0ycvRq3VVldMwpBZdEbJ

Tải về toàn bộ các models được train trước đó và di chuyển vào thư mục models trong project (lúc này cây thư mục trong folder models sẽ như sau):

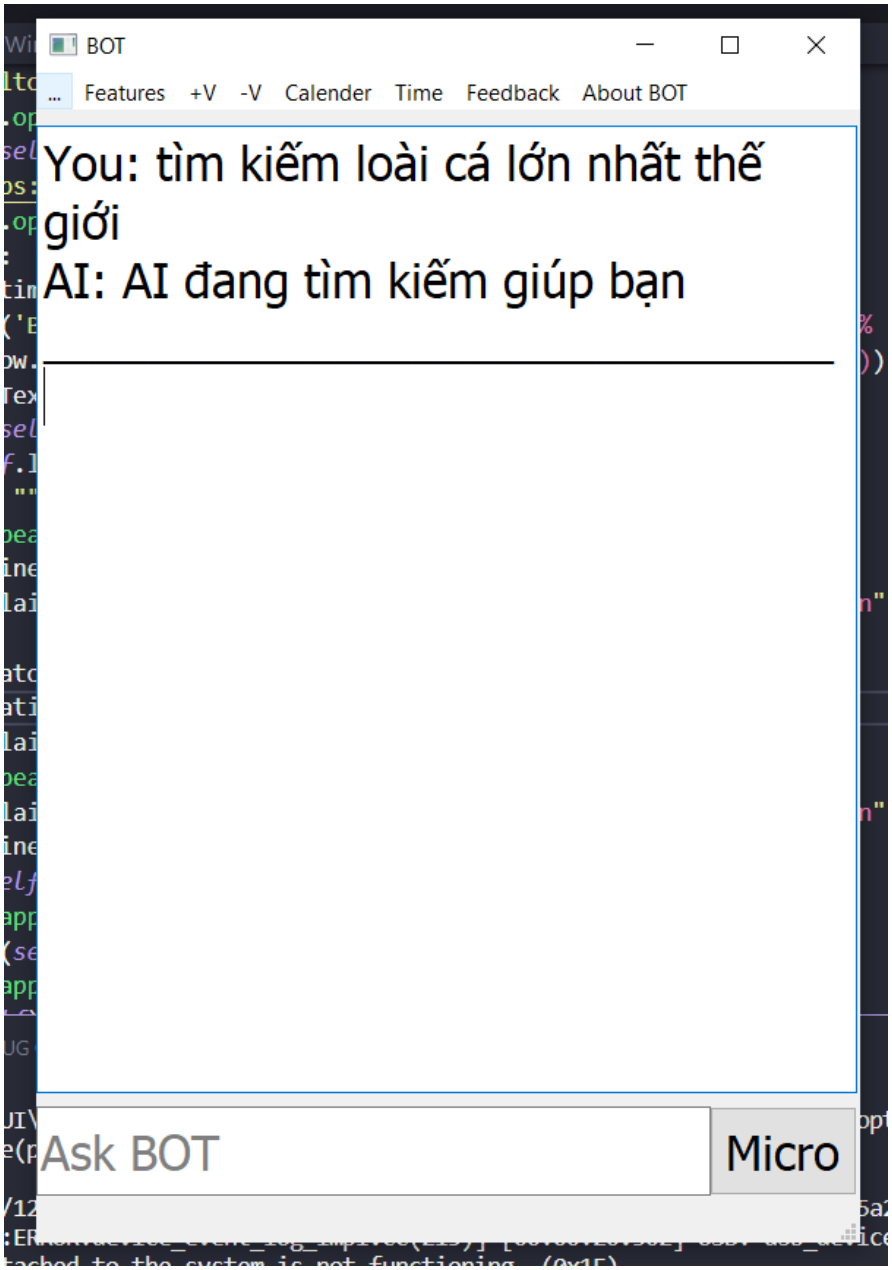


Hình 4: Các file .pkl trong folder models

Tiếp theo tại file BOT.py tiến hành chạy project



Giao diện của Virtual Assistant sẽ như sau nghĩa là chạy thành công:



Hình 5: Giao diện của BOT (Virtual Assistant)

*** Notes: Đảm bảo đã cài đặt các thư viện cần thiết để chạy project bằng pip install**

2. Mô tả ứng dụng

2.1. Tìm hiểu một số thư viện chính được sử dụng trong đồ án

- **Tensorflow**

Tensorflow là một thư viện toán học ký hiệu sử dụng luồng dữ liệu và lập trình có thể phân biệt để thực hiện các nhiệm vụ khác nhau, tập trung vào đào tạo và suy luận các mạng nơ-ron sâu (deep neural network) [6].

    - Cách hoạt động của tensorflow: chúng cho phép chúng ta xây dựng các biểu đồ và cấu trúc luồng dữ liệu (hay còn gọi là dataflow graph), các cấu trúc này mô tả cách dữ liệu có thể di chuyển qua một biểu đồ bằng cách lấy đầu vào là một mảng đa chiều được gọi là Tensor.
    - Kiến trúc của tensorflow: hoạt động trong ba phần là xử lý trước dữ liệu, xây dựng mô hình, đào tạo và ước tính mô hình.
    - Thành phần của tensorflow: Tensor, Graphs
- **Sklearn**

Sklearn là một thư viện hỗ trợ các thuật toán hiện đại như KNN, XGBoost, random forest, SVM và một số thuật toán khác. Scikit-learn giúp tiền xử lý, giảm chiều dữ liệu (lựa chọn tham số), phân loại, hồi quy, phân cụm và model selection [7].

Sau đâu là một số nhóm thuật toán được xây dựng bởi thư viện scikit-learn:

- Clustering: Nhóm thuật toán Phân cụm dữ liệu không gán nhãn. Ví dụ thuật toán KMeans
- Cross Validation: Kiểm thử chéo, đánh giá độ hiệu quả của thuật toán học giám sát sử dụng dữ liệu kiểm thử (validation data) trong quá trình huấn luyện mô hình.
- Datasets: Gồm nhóm các Bộ dữ liệu được tích hợp sẵn trong thư viện. Hầu như các bộ dữ liệu đều đã được chuẩn hóa và mang lại hiêu suất cao trong quá trình huấn luyện như iris, digit, ...
- Dimensionality Reduction: Mục đích của thuật toán này là để Giảm số lượng thuộc tính quan trọng của dữ liệu bằng các phương pháp như tổng hợp, biểu diễn dữ liệu và lựa chọn đặc trưng. Ví dụ thuật toán PCA (Principal component analysis).
- Ensemble methods: Các Phương pháp tập hợp sử dụng nhiều thuật toán học tập để có được hiệu suất dự đoán tốt hơn so với bất kỳ thuật toán học cấu thành nào.
- Feature extraction: Trích xuất đặc trưng. Mục đích là để định nghĩa các thuộc tình với dữ liệu hình ảnh và dữ liệu ngôn ngữ.
- Feature selection: Trích chọn đặc trưng. Lựa chọn các đặc trưng có ý nghĩa trong việc huấn luyện mô hình học giám sát.
- Parameter Tuning: Tinh chỉnh tham số. Các thuật toán phục vụ việc lựa chọn tham số phù hợp để tối ưu hóa mô hình.
- Manifold Learning: Các thuật toán học tổng hợp và Phân tích dữ liệu đa chiều phức tạp.
- Supervised Models: Học giám sát. Mảng lớn các thuật toán học máy hiện nay. Ví dụ như linear models, discriminate analysis, naive bayes, lazy methods, neural networks, support vector machines và decision trees.
- **Numpy**

Numpy là một thư viện lõi phục vụ cho khoa học máy tính của Python, hỗ trợ cho việc tính toán các mảng nhiều chiều, có kích thước lớn với các hàm đã được tối ưu áp dụng lên các mảng nhiều chiều đó. Numpy đặc biệt hữu ích khi thực hiện các hàm liên quan tới Đại Số Tuyến Tính [8].

Với Numpy ta có thể thực hiện các thao tác sau:

- Các phép toán toán học và logic trên mảng.
- Các biến đổi Fourier và các quy trình để thao tác shape.
- Các phép toán liên quan đến đại số tuyến tính. NumPy tích hợp sẵn các hàm cho đại số tuyến tính và tạo số ngẫu nhiên.

Mảng trong Numpy:

Mảng là một cấu trúc dữ liệu chứa một nhóm các phần tử. Thông thường, tất cả các phần tử này có cùng kiểu dữ liệu, chẳng hạn như số nguyên hoặc chuỗi. Chúng thường được sử dụng trong các chương trình để sắp xếp dữ liệu để một bộ giá trị liên quan có thể dễ dàng được sắp xếp hoặc tìm kiếm. Khi nói đến NumPy, một mảng là một cấu trúc dữ liệu trung tâm của thư viện.

Sự khác biệt giữa Python List và Numpy Array:

- Python List có thể chứa các phần tử với các kiểu dữ liệu khác nhau trong khi các phần tử của Numpy Array luôn đồng nhất (cùng một kiểu dữ liệu).
- Python Array nhanh hơn và nhỏ gọn hơn Python List:
    - NumPy Array sử dụng bộ nhớ cố định để lưu trữ dữ liệu và ít bộ nhớ hơn Python List.
    - Cấp phát bộ nhớ liền kề trong NumPy Array

Đối tượng kiểu dữ liệu

Một đối tượng kiểu dữ liệu mô tả diễn giải khối bộ nhớ cố định tương ứng với một mảng, tùy thuộc vào các khía cạnh sau:

- Loại dữ liệu (số nguyên, đối tượng float hoặc Python)
- Kích thước của dữ liệu
- Thứ tự Byte (little-endian hoặc big-endian)
- Trong trường hợp kiểu có cấu trúc, tên của các trường, kiểu dữ liệu của từng trường và một phần của khối bộ nhớ được lấy bởi từng trường.
- Nếu kiểu dữ liệu là một mảng con, hình dạng và kiểu dữ liệu của nó

Array Indexing: NumPy cung cấp một số cách để truy xuất phần tử trong mảng

Boolean array indexing: Cho phép bạn chọn ra các phần tử tùy ý của một mảng, thường được sử dụng để chọn ra các phần tử thỏa mãn điều kiện nào đó

- **Matplotlib**

Matplotlib là một trong những thư viện Python phổ biến nhất được sử dụng để trực quan hóa dữ liệu. Nó là một thư viện đa nền tảng để tạo các đồ thị 2D từ dữ liệu trong các mảng. Matplotlib được viết bằng Python và sử dụng NumPy, phần mở rộng toán học của Python. Ta thường dùng Matplotlib để thực hiện các suy luận thống kế cần thiết [9].

Một Matplotlib figure có thể được phân loại thành nhiều phần như dưới đây:

- Figure: Như một cái cửa sổ chứa tất cả những gì bạn sẽ vẽ trên đó.
- Axes: Thành phần chính của một figure là các axes (những khung nhỏ hơn để vẽ hình lên đó). Một figure có thể chứa một hoặc nhiều axes. Nói cách khác, figure chỉ là khung chứa, chính các axes mới thật sự là nơi các hình vẽ được vẽ lên.
- Axis: Chúng là dòng số giống như các đối tượng và đảm nhiệm việc tạo các giới hạn biểu đồ.
- Artist: Mọi thứ mà bạn có thể nhìn thấy trên figure là một artist như Text objects, Line2D objects, collection objects. Hầu hết các Artists được gắn với Axes.

- **Pandas**

Pandas là một thư viện hỗ trợ đắc lực trong thao tác dữ liệu. Đây cũng là bộ công cụ phân tích và xử lý dữ liệu mạnh mẽ của ngôn ngữ lập trình python. Thư viện này sử dụng một cấu trúc dữ liệu riêng là Dataframe. Pandas cung cấp rất nhiều chức năng xử lý và làm việc trên cấu trúc dữ liệu này [10].

Tại sao chúng ta sử dụng Pandas?

- DataFrame đem lại sự linh hoạt và hiệu quả trong thao tác dữ liệu và lập chỉ mục;
- Là một công cụ cho phép đọc/ ghi dữ liệu giữa bộ nhớ và nhiều định dạng file: csv, text, excel, sql database, hdf5;
- Liên kết dữ liệu thông minh, xử lý được trường hợp dữ liệu bị thiếu. Tự động đưa dữ liệu lộn xộn về dạng có cấu trúc;
- Dễ dàng thay đổi bố cục của dữ liệu;
- Tích hợp cơ chế trượt, lập chỉ mục, lấy ra tập con từ tập dữ liệu lớn.
- Có thể thêm, xóa các cột dữ liệu;
- Tập hợp hoặc thay đổi dữ liệu với group by cho phép bạn thực hiện các toán tử trên tập dữ liệu;
- Hiệu quả cao trong trộn và kết hợp các tập dữ liệu;
- Lập chỉ mục theo các chiều của dữ liệu giúp thao tác giữa dữ liệu cao chiều và dữ liệu thấp chiều;

- Tối ưu về hiệu năng;
- Pandas được sử dụng rộng rãi trong cả học thuật và thương mại. Bao gồm thống kê, thương mại, phân tích, quảng cáo,…

- **Seaborn**

Seaborn là một thư viện Python được sử dụng để tạo biểu đồ trực quan hóa cho tập dữ liệu. Nó được phát triển để giúp chúng ta dễ dàng tạo ra các kiểu biểu đồ phổ biến nhất. Biểu đồ bên dưới đây có thể được tạo ra chỉ với một vài dòng mã thông qua thư viện Seaborn.

Trực quan hóa dữ liệu được thực hiện dễ dàng trong Seaborn và đây là cách quy trình của nó có thể như sau [11]:

- Dữ liệu từ nhiều nguồn khác nhau: Dữ liệu cần thiết để thực hiện trực quan hóa và phân tích có thể đi vào kiến trúc từ nhiều nguồn khác nhau, chẳng hạn như đơn vị lưu trữ cục bộ, máy chủ, cấu trúc đám mây, v.v.
- Trực quan hóa dữ liệu: Đây là nơi dữ liệu được chuyển đổi từ trạng thái số của nó thành một đối tác trực quan đẹp mắt về mặt thẩm mỹ. Seaborn đóng vai trò chính ở đây.
- Phân tích dữ liệu : Kết quả của trực quan hóa dữ liệu là xem xét dữ liệu theo cách bạn chưa làm trước đây. Phân tích chỉ giúp làm điều này để tiết lộ thông tin chi tiết và xu hướng mà không thể được phát hiện bằng cách khác.

Sự phụ thuộc Seaborn: 4 thư viện khác được sử dụng cùng với Seaborn hầu hết thời gian là Pandas, Numpy, SciPy, Matplotlib

- **Keras**

Keras là một thư viện mạng nơ ron được viết bằng python năm 2015 bởi 1 kỹ sư deep learning của google. Ta có thể kết hợp keras với các thư viện deep learning. (Ta có thể kết hợp sử dụng keras và tensorflow) [12].

Một số ưu điểm của Keras là:

- Dễ sử dụng, dùng đơn giản hơn Tensor, xây dựng model nhanh.
- Run được trên cả CPU và GPU.
- Hỗ trợ xây dựng CNN , RNN hoặc cả hai. Với những người mới tiếp cận đến Deep như mình thì mình chọn sử dụng Keras để build model vì nó đơn giản,dễ nắm bắt hơn các thư viện khác.

- **Pickle và Joblib**

Pickle được sử dụng để thực hiện chuyển đổi các cấu trúc đối tượng Python sang một dạng byte để có thể được lưu trữ trên ổ đĩa hoặc được gửi qua mạng. Sau đó, luồng ký tự

này sau đó có thể được truy xuất và chuyển đổi trở lại sang dạng đối tượng ban đầu trong Python.

Joblib là một phần của hệ sinh thái SciPy, nó cũng hỗ trợ việc lưu ML model thành file rất dễ dàng, sử dụng cấu trúc dữ liệu của NumPy. Ưu điểm của viêc sử dụng joblib so với pickle là nó hoạt động khá nhanh, đặc biệt với những model có kích thước lớn.

2.2. Các tính năng AI

Danh sách các tính năng AI của ứng dụng

STT	Tên chức năng	Mô tả ngắn công dụng	Ghi chú (nếu có)*
1	Chào hỏi	BOT trả lời chào theo buổi trong ngày (chào buổi sáng, trưa, chiều,...)	
2	Xem thời tiết	Xem thời tiết theo vị trí*	Đơn vị hành chính cấp tỉnh
3	Xem thời gian	Xem giờ:phút:giây Ngày/tháng/năm Thứ trong tuần - Toàn bộ các thông tin trên	
4	Mở ứng dụng	Mở các ứng dụng đã cài đặt trong máy tính	
5	Mở website	Mở webiste theo dạng <tên website>.<tên miền>*	VD: wikipedia.org
6	Tìm kiếm trên Internet	Mở webbrowser* và tìm kiếm theo ý người dùng	Webbrowser là default browser trong máy
7	Tìm kiếm trên youtube	Tìm kiếm bài hát, phim,.. trên youtube	
8	Tìm kiếm trên wikipedia	Tìm kiếm bài vết trên wikipedia tiếng Việt	
9	Tính toán	Tính toán số học với các phép tính cơ bản như +, -, * , /, ước chung lớn nhất, giai thừa,..*	Nguồn lấy kết quả phép toán trong ứng dụng: https://www.calculator.net/big-number-calculator.html
10	Tăng/giảm âm lượng	Tăng/giảm âm lượng trong máy tính*	Một click tương ứng với tăng/giảm một (1) âm lượng
11	Translate	Dịch câu tiếng Việt sang tiếng Anh	
12	Calculator	Mở phần mềm calculator trong máy tính	
13	Learn English	Mở website học tiếng Anh*	Nguồn website dùng cho chức năng:https://www.tienganh123.com/

14	Math Formulas	Mở website danh sách các công thức toán*	Nguồn website dùng cho chức năng:https://www.cuemath.com/math-formulas/
15	Check URL	Kiểm tra độ an toàn của website dựa vào URL của nó	Mục **2.3.1 – Tính năng DL**
16	Check password	Kiểm tra độ mạnh của mật khẩu	Mục **2.3.2 – Tính năng ML**
17	Image to Text	Convert các chữ trong ảnh thành chữ dạng text (dạng có thể copy được)	Mục **2.3.3 – Tính năng DL**
18	Audio Book	Đọc tất cả nội dung có trong file PDF	
19	Feedback	Mở email với tiêu đề Feedback	
20	Clear Screen	Xóa toàn bộ kết quả hiển thị trên màn hình ứng dụng	
21	Now	Xem thông tin thời gian hiện tại	
22	Calendar	Mở phần mềm calendar trong máy tính	
23	Setting	Mở phần mềm setting trong máy tính	
24	About BOT	Mở link github toàn bộ source web	
25	About us	Xem thông tin tác giả	
26	Exit	Thoát ứng dụng	
27	Tìm kiếm mở rộng	Gợi ý tìm kiếm theo từ khóa nhận từ người dùng mà ứng dụng chưa được set up trước	

2.3. Các tính năng ML & DL

2.3.1. Check URL

2.3.1.1. URL là gì?

URL (Uniform Resource Locator) chúng ta có thể hiểu đơn giản đây là 1 địa chỉ web hay là một đường dẫn liên kết đến website, tham chiếu tới các tài nguyên trên mạng Internet. Đường dẫn URL này là một đoạn văn bản có thể đọc được thay cho địa chỉ IP mà máy tính sử dụng để liên hệ với server.

Phân loại URL: thông thường mọi website đều có thể có 2 loại

- URL động (ví dụ: http: //www.example.com/?name=…)
- URL tĩnh (ví dụ: http: //www.example.com/index.html)

Các phần cơ bản của URL: thông thường một URL cơ bản gồm

- Các giao thức: FTP, http, https,…
- World Wide Web: www (mạng lưới toàn cầu)
- Tên miền (domain): https://www.example.com/

- Cổng giao tiếp (port): 80, 2222, 8484, 8888,…

Scheme (giao thức) trong URL: Scheme này giúp chúng ta biết được trình duyệt web sử dụng phương thức nào để giao tiếp với server. Các loại scheme thường gặp bao gồm:

- HTTP (port 80 – giao tiếp): xác định các hành động của máy chủ với thao tác của người dùng trên trình duyệt web bằng các lệnh nhất định.
- HTTPS (port 433 – truyền dữ liệu): sử dụng SSL (Secure Socket Layer) để đảm bảo truyền dữ liệu an toàn giữa web server - trình duyệt website.
- FTP: chuyển đổi file (trình duyệt – server)

Authority (nhà cung cấp) của url: gồm nhiều thành phần như

- Tên miền (chúng ta thường thấy nhất những tên miền): .com, .vn, .org, .us, .jp
- Tên miền phụ (subdomain): ví dụ vào Wikipedia tiếng anh ta thấy website có url: https://en.wikipedia.org/wiki/URL vây thì en chính là tên miền phụ.
- Thông tin người dùng.
- Số cổng

Các thành phần bổ sung của URL:

- Path (đường dẫn): ví dụ www.example.com/folder/subfolder/filename.html
- Query (truy vấn): ví dụ https://www.google.com.vn/search?q=singing&sxsrf=ALiCzsbfT5KkGN9cZlWZnxf64dc3Jb9X4Q%3A1669733931847&ei=Kx6GY4-uM8LbhwPd_7qQDQ&ved=0ahUKEwjP74HP09P7AhXC7WEKHd2_DtIQ4dUDCA8&uact=5&oq=singing&gs_lcp=Cgxnd3Mtd2l6LXNlcnAQAzILCAAQgAQQsQMQgwEyCAgAEIAEELEDMggILhDUAhCABDIFCAAQgAQyBQgAEIAEMggILhCABBDUAjIFCAAQgAQyBQgAEIAEMgUIABCABDIFCAAQgAQ6BwgjEOoCECc6BAgjECc6BwguENQCEEM6BAguEEM6EQguEIAEELEDEIMBEMcBENEDOgQIABBDOgcIABCxAxBDOgQIABADOggILhCABBCxAzoOCC4QgAQQsQMQxwEQ0QNKBAhBGABKBAhGGABQAFjYM2C4NGgBcAF4AIABeIgBgQWSAQMwLjaYAQCgAQGwAQrAAQE&sclient=gws-wiz-serp

Fragment (phân mảnh): bắt đầu bằng dấu # và được sử dụng để xác định trang vị trí cụ thể của trang web.

2.3.1.2. Phân loại URL dựa trên các loại tấn công: begign, defacement, malware, phising, spam

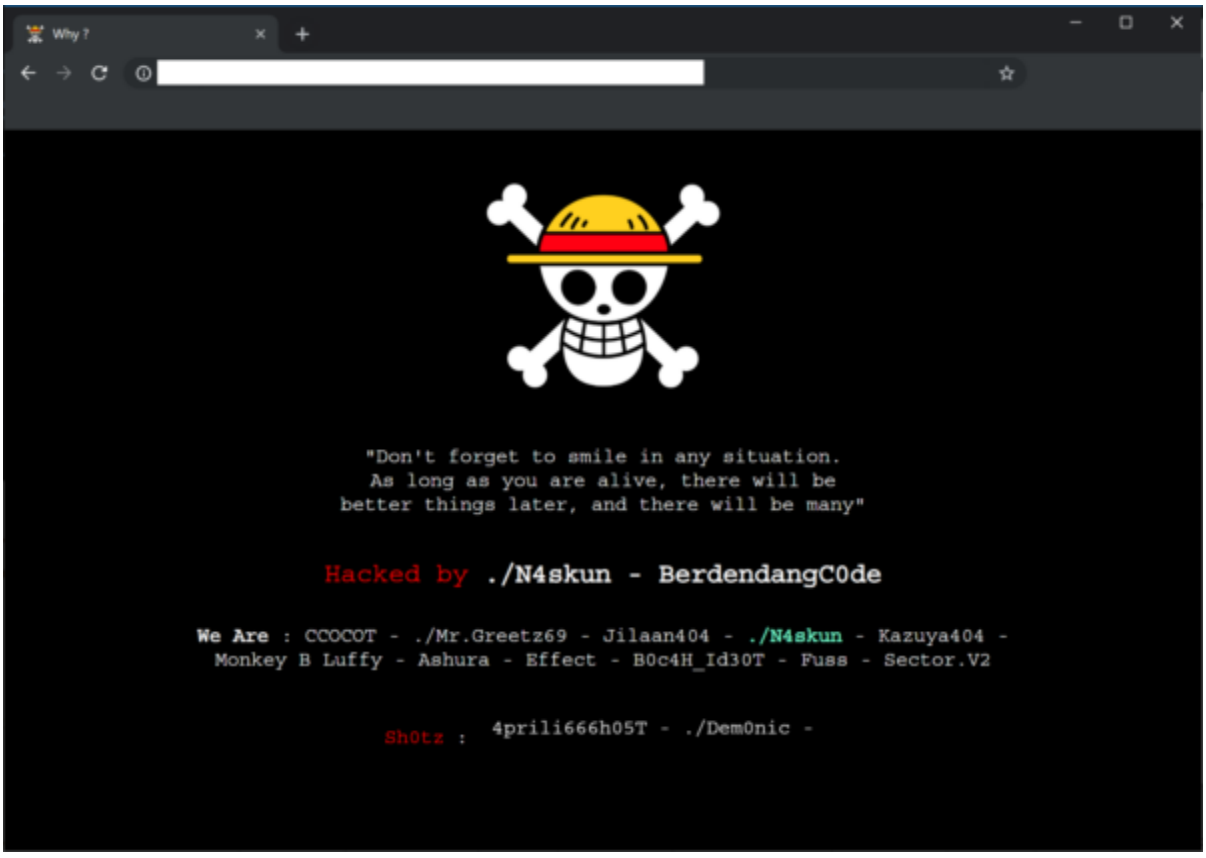
Benign: các loại url này thường được nhắc đến như là các url dẫn đến các website vô hại (không nhằm mục đích tấn công người dùng hoặc ngược lại với độc hại).

Ví dụ: https://www.google.com.vn/

Defacement: chúng ta có thể hiểu tấn công loại này nghĩa là kẻ tấn công sẽ xâm nhập vào một trang web và thay thế nội dung trong trang web bằng nội dung của họ. Các nội dung này có thể là những nội dung như thông điệp chống phá liên quan tới chính trị, tôn giáo, nội dung thô tục hoặc không phù hợp khác. Một số nguyên nhân dẫn đến các cuộc tấn công defacement:

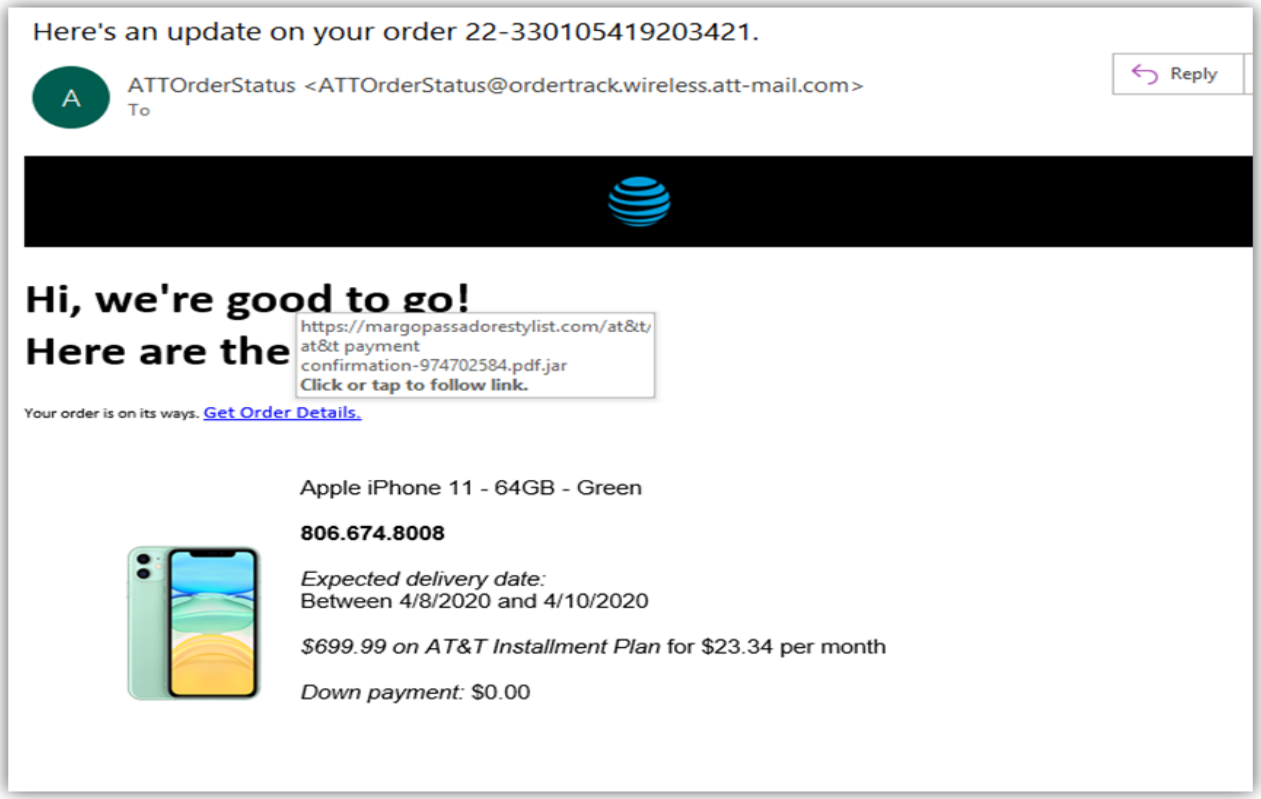
- Truy cập trái phép
- SQL Injection
- Cross-site scription (Tấn công XSS)
- Chiếm quyền điều khiển DNS
- Malware infection

Ví dụ về một trang web bị defacement: ta có thể hiểu 1 trang web nào đó liên quan đến y tế chẳng hạn, khách hàng có thể lên đó và đặt lịch khám, xem các loại thuốc, thông tin,… nhưng bỗng dung một ngày khi truy cập vào trang web chúng ta thấy 1 nội dung hoàn toàn khác lạ hay chỉ là một screen thông báo nào đó (có thể như hình) nghĩa là website này (url này) đã bị defacement attack



Hình 6: Ví dụ về website (hay url) đã bị defacement attack

Malware: chúng ta có thể hiểu tấn công kiểu này được tạo với mục đích phân phối phần mềm độc hại như ransomware. Chúng thường chứa trong thư rác, thư lừa đảo và email lừa đảo. Thông thường, chúng được ngụy trang bằng các công cụ rút ngắn URL như Bit.ly hoặc một siêu liên kết đã sửa đổi. Hay ví dụ như chúng ta đang tải một tệp gì đó hay xem phim lậu chẳng hạn thì bỗng dung click vào một ví trí bất kỳ và một tab mới tự động mở ra liên quan trong đó nội dung liên quan đến cờ bạc, cá độ hay các thông tin thông báo rằng bạn trúng thưởng để lừa bạn → đây rất có thể là 1 website chứa malware.



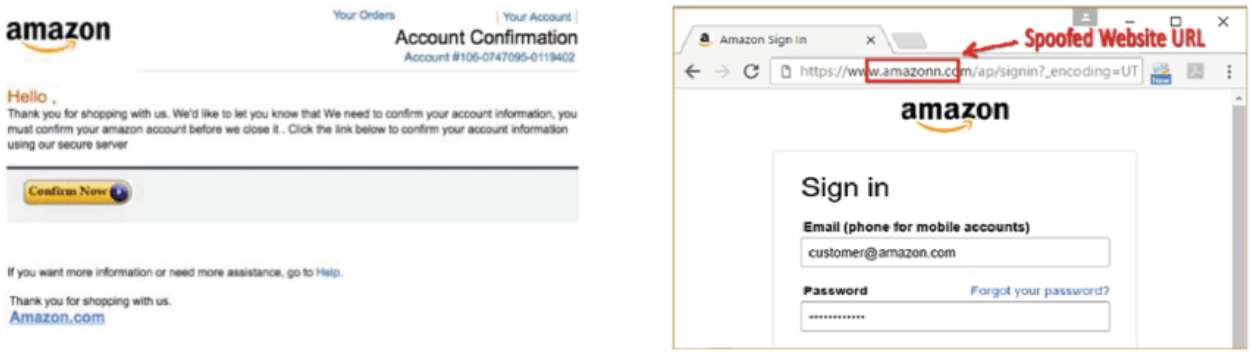
Hình 7: Ví dụ về một website có thể chứa các malware

Phising: chúng ta có thể hiểu một Phising URL là một hoạt động lừa đảo bắt chước các liên kết internet quen thuộc để xúi giục nạn nhân nhấp vào chúng. Các liên kết này thường dẫn đến các trang web độc hại, chứa phần mềm độc hại nhằm đánh cắp thông tin đăng nhập của một người, đặc biệt là thông tin ngân hàng và mật khẩu. Các trang web này có thể là các liên kết trong email hoặc các website giả mạo nhìn trông giống như các website nổi tiếng như Twitter, Facebook, Google,…

Cách xác định một Phising URL:

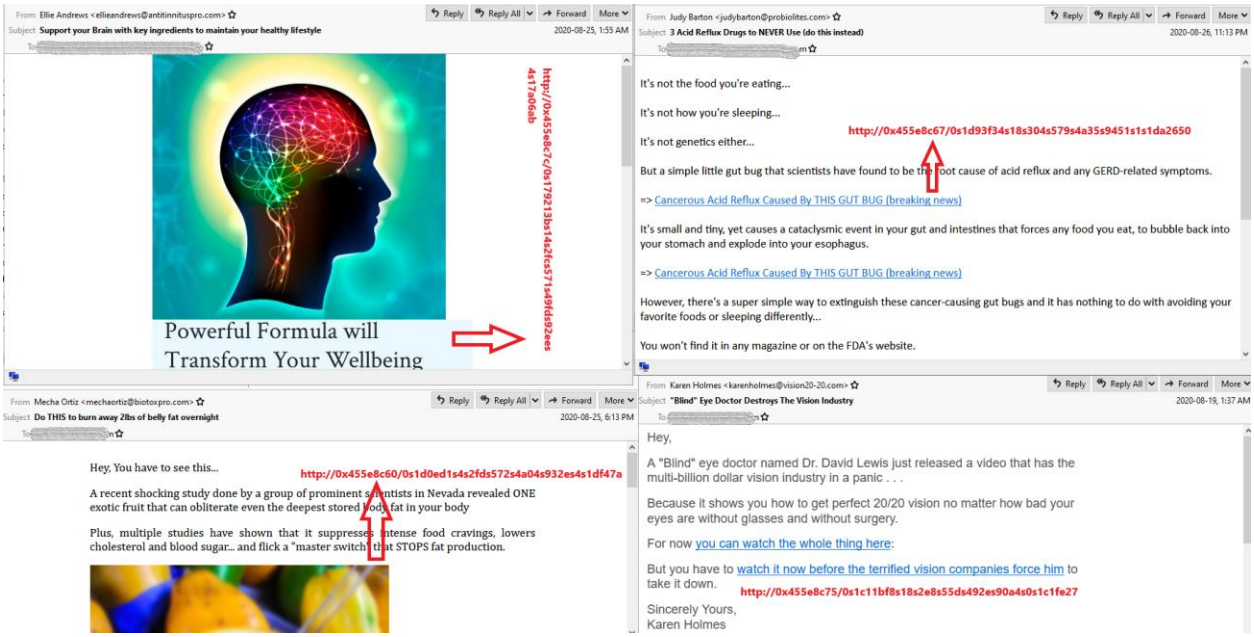
- Giao thức có đúng không (“ https://”)?
- Liên kết có tên miền phụ không và nó nằm ở đâu?

- Liên kết có chuyển hướng bạn qua Google Tìm kiếm hoặc các trang web khác không?
- Tên miền có đúng và có đuôi phù hợp không (.com, .net, v.v.)?
- Liên kết được hiển thị có ẩn một liên kết khác khi bạn di chuột qua liên kết đó không?
- Địa chỉ nguồn của email trông có đáng tin cậy không



Hình 8: Ví dụ về Phising URL (trông giống như website amazon nhưng thực tế thì không)

Spam: chúng ta có thể hiểu các cuộc tấn công spam được định nghĩa là việc sử dụng một ứng dụng có tổ chức và trái phép để gửi hàng nghìn tin nhắn cho người dùng. Những tin nhắn này được gửi bởi các hồ sơ giả mạo hoặc bị tấn công và thường bao gồm các quảng cáo và liên kết không có thật mà người dùng thực được yêu cầu nhấp vào (đôi khi các website spam này có chứa cả malware và những thứ độc hại khác).

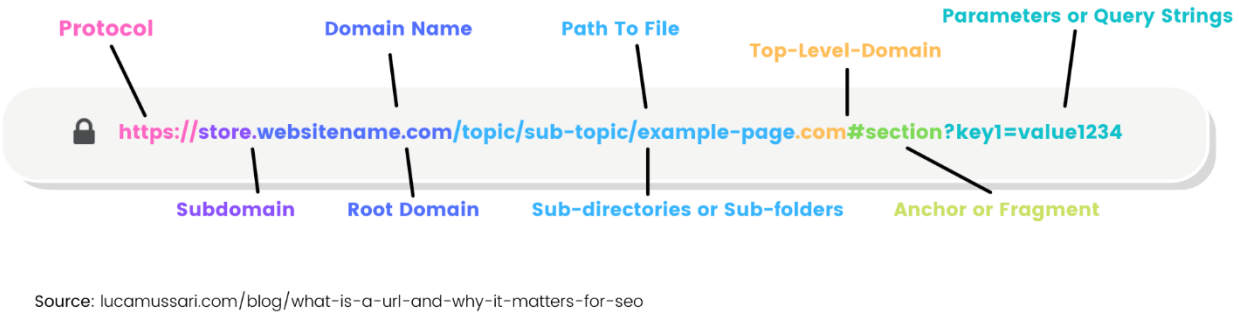


Hình 9: Ví dụ về các spam url

2.3.1.3. Build Model

2.3.1.3.1. Data Processing

Thành phần của một URL có dạng như sau:



Hình 10: Thành phần của một URL

Xử lý URL feature

Feature Name	Feature Group	Ý nghĩa
URL Entropy	URL String characteristics	Entropy của URL
numDigits	URL String characteristics	Tổng số lượng chữ số trong URL
URL length	URL String characteristics	Độ dài của URL
numParameters	URL String characteristics	Tổng số lượng quey parameter trong URL
numFragments	URL String characteristics	Tổng số lượng fragment trong URL
domainExtension	URL String characteristics	Doamin extension của URL
num_%20	URL String characteristics	Tổng số lượng %20 trong URL
num_@	URL String characteristics	Tổng số lượng @ trong URL
has_ip	URL String characteristics	Kiểm tra sự xuất hiện của ip trong URL
hasHTTP	URL domain features	Website domain có giao thức http
hasHTTPS	URL domain features	Website domain có giao thức https
urlLive	URL domain features	Trang web đang hoạt động

daysSinceRegistration	URL domain features	Số ngày tính từ ngày domain được đăng ký đến ngày hôm nay
daysSinceExpiration	URL domain features	Số ngày tính từ ngày ngày hôm nay đến domain hết hạn đăng ký
bodyLength	URL page feature	Tổng số lượng ký tự trong trang HTML của URL
numTitles	URL page feature	Tổng số lượng thẻ H1 – H6 trong trang HTML của URL
numImages	URL page feature	Tổng số lượng hình ảnh được nhúng trong trang HTML của URL
numLinks	URL page feature	Tổng số lượng link được nhúng trong trang HTML của URL
scriptLength	URL page feature	Tổng số lượng ký tự được nhúng trong thẻ script trong trang HTML của URL
specialCharacters	URL page feature	Tổng số lượng ký tự đặc biệt trong trang HTML của URL
scriptToSpecialCharsRatio	URL page feature	Tỷ lệ giữa số lượng ký tự được nhúng trong thẻ script và số lượng ký tự đặc biệt trong trang HTML của URL
scriptTobodyRatio	URL page feature	Tỷ lệ giữa số lượng ký tự được nhúng trong thẻ script và số lượng ký tự trong trang HTML của URL

Convert các url trong các file csv thành các feature như bảng trên và gộp thành một file csv chung

```
l = ['DefacementSitesURLFiltered.csv', 'phishing_dataset.csv',
     'Malware_dataset.csv', 'spam_dataset.csv', 'Benign_list_big_final.csv']


emp = UrlFeaturizer("").run().keys()
```

```
A = pd.DataFrame(columns=emp)
t = []
for j in l:
    print(j)
    d = pd.read_csv(j, header=None).to_numpy().flatten()
    for i in tqdm(d):
        try:
            temp = UrlFeaturizer(i).run()
            temp["File"] = j.split(".")[0]
            t.append(temp)
        except requests.Timeout as err:
            pass
```

```
A = A.append(t)
A.to_csv("../Dataset/feature.csv")
```

Drop cột thứ tự và map giá trị true là 1 và false là 0

```
data.drop(columns='Unnamed: 0', inplace=True)
data.replace(True, 1, inplace=True)
data.replace(False, 0, inplace=True)
```

Dùng LabelEncoder để encode label

```
encoder = LabelEncoder()
encoder.fit(y)
Y = encoder.transform(y)
```

Dùng MinMaxScaler để scaler các giá trị của dữ liệu trong khoảng từ 0 đến 1

```
scaler = MinMaxScaler(feature_range=(0, 1))
X = scaler.fit_transform(data)
X = pd.DataFrame(X)
```

2.3.1.3.2. Training Model

Khởi tạo model

```
input_dim = len(data.columns)
model = Sequential()
model.add(Dense(256, input_dim=input_dim, activation='relu'))
model.add(Dense(128, activation='relu'))
model.add(Dense(64, activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(16, activation='relu'))
model.add(Dense(5, activation='softmax'))
```

```
model.compile(loss='categorical_crossentropy',
              optimizer='adam', metrics=['accuracy'])
```

Summary model

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 256)               5632

 dense_1 (Dense)             (None, 128)               32896

 dense_2 (Dense)             (None, 64)                8256

 dense_3 (Dense)             (None, 32)                2080

 dense_4 (Dense)             (None, 16)                528

 dense_5 (Dense)             (None, 5)                 85

=================================================================
Total params: 49,477
Trainable params: 49,477
Non-trainable params: 0
_________________________________________________________________
```

Kết quả sau khi training

	precision	recall	f1-score	support
Benign	0.80	0.71	0.76	1998
Defacement	0.86	0.87	0.86	2273
Malware	0.90	0.95	0.93	2167
Phishing	0.72	0.79	0.75	2013
Spam	0.91	0.87	0.89	2456
accuracy			0.84	10907
macro avg	0.84	0.84	0.84	10907
weighted avg	0.84	0.84	0.84	10907

2.3.2. Check Password

2.3.2.1. Password là gì?

Chúng ta biết rằng kể thì khi máy tính ra đời thì khoa học kỹ thuật ngày càng phát triển với tốc độ nhanh chóng, thế nhưng cũng từ đó lại tồn tại rất nhiều rủi ro bảo mật liên quan tới các vấn đề rất quan trọng như lộ thông tin cá nhân. Để phòng chống những việc như vậy thì password xuất hiện với vai trò như là một bí mật, thường là một chuỗi ký tự được sử dụng để xác nhận danh tính người dùng.

Một mật khẩu là một chuỗi các ký tự tùy ý bao gồm chữ cái, chữ số, hoặc các biểu tượng khác. Nếu các ký tự cho phép bị ràng buộc là số, thì bí mật tương ứng đôi khi được gọi là số nhận dạng cá nhân (personal identification number - PIN).

Tuy công nghệ bảo mật bằng Password xuất hiện gần như cùng lúc với máy tính, thế nhưng ban đầu công nghệ này vẫn chưa thật sự an toàn và có thể bị hack dễ dàng. Dạng bảo mật này chỉ yêu cầu người sử dụng nhập các chuỗi ký tự được lưu trữ trước đó, nếu nhập chính xác thì hệ thống sẽ chấp nhận cho bạn truy cập. Và tất nhiên, khi nhập sai quá số lần quy định thì hệ thống sẽ từ chối và thậm chí không cho bạn tiến hành nhập mật khẩu nữa.

Các yếu tố trong hệ thống bảo mật của mật khẩu: bảo mật của hệ thống dựa vào sự hỗ trợ của mật khẩu phụ thuộc vào một số yếu tố. Đó là hệ thống tổng thế phải được thiết kế bảo mật tốt, với khả năng chống lại các loại virus máy tính, các cuộc tấn công trung gian hoặc tương tự. Kế đó là vấn đề an ninh nghĩa là chúng ta phải cẩn thận hạn chế tối đa các trường hợp quay video và copy, keylog từ bàn phím (bắt được hành động gõ phím của người dùng). Mật khẩu nên được chọn sao cho khó đoán, khó phát hiện, hoặc khó khăn

để lấy được mật khẩu kể cả khi người đó dùng một số thủ thuật, hoặc công cụ hỗ trợ nào đó.

Dưới đây là một số vấn đề về quản lý mật khẩu:

- **Tần suất người khác có thể thử đoán mật khẩu**
- **Giới hạn về số lần đoán mật khẩu**
- **Hình thức mật khẩu được lưu trữ**

2.3.2.2. Độ bảo mật mạnh của mật khẩu

Độ mạnh của mật khẩu là một thuật ngữ để chỉ mức độ khó khăn trong việc khám phá ra một mật khẩu nào đó [13].

Chúng ta có thể xem một mật khẩu có mạnh hay không bằng việc nó có thể dễ dàng được đoán ra hay không, nó có liên quan tến bất kỳ sự kiện nào của người tạo ra mật khẩu hay không, nếu bị tấn công bằng các công cụ (…, vét cạn) thì khả năng tìm ra mật khẩu mất bao lâu.

Chúng ta thường chỉ chia ra 2 loại mật khẩu (thực tế có thể chia làm 3):

- **Mật khẩu yếu:** mật khẩu yếu có thể ngắn, dễ đoán, có thể là mặc khẩu mặc định do hệ thống cung cấp, hoặc một mật khẩu có thể đoán ra nhanh bằng một loạt các cuộc tấn công có sử dụng phương pháp vét cạn (thường sử dụng một tập hợp của các mật khẩu liên quan đến tên riêng, thú cưng, sinh nhật, sở thích, tên người yêu,…). Các ví vụ về mật khẩu yếu:
    - admin—quá dễ đoán
    - 1234—quá dễ đoán
    - abc123—quá dễ đoán
    - minh—tên riêng thông thường
    - password—đoán ra dễ dàng, rất thường dùng
    - p@$$\/\/0rd -- leet và mật mã bằng ký tự đơn giản đều đã được lập trình trước trong các công cụ bẻ khóa
    - rover—tên thú nuôi thông thường, cũng là một từ trong từ điển
    - 12/3/75—ngày tháng, có thể quan trọng đối với cá nhân đó
    - December12—Sử dụng ngày bắt buộc phải đổi mật khẩu là rất phổ biến
    - nbusr123—có thể là một tên người dùng, và nếu vậy, cực kỳ dễ đoán
    - asdf—chuỗi ký tự kế nhau trong nhiều loại bàn phím
    - qwerty—một chuỗi ký tự kế nhau trong nhiều loại bàn phím
    - aaaa—ký tự lặp đi lặp lại, dễ đoán ra
    - iloveyou – top các mật khẩu dễ đoán.

- **Mật khẩu mạnh:** mật khẩu mạnh là mật khẩu đủ dài, không dễ đoán, mang tính ngẫu nhiên hoặc chỉ có người tạo ra mật khẩu này mới có thể đoán ra được, mật khẩu mạnh phải là mật khẩu dù kẻ tấn công có sử dụng công cụ đi chăng nữa cũng phải mất rất nhiều thời gian để tìm được mật khẩu ước tính > 70 năm. Các ví dụ về mật khẩu mạnh:
    - t3wahSetyeT4 -- phân biệt chữ thường chữ hoa và chữ số xen kẽ
    - 4pRte!ai@3—phân biệt chữ thường chữ hoa, chữ số xen kẽ, dấu câu và một ký tự "đặc biệt"
    - MoOoOfIn245679—phân biệt chữ thường chữ hoa, chữ số xen kẽ
    - Convert_100£ to Euros!—cụm từ có thể dài, dễ nhớ và có chứa ký hiệu mở rộng để tăng sức mạnh, nhưng một số phương pháp băm mật khẩu yếu hơn có thể phụ thuộc vào phân tích tần số
    - 1382465304H—một chuỗi số kết thúc bằng một ký tự
    - Tp4tci2s4U2g!—Sự pha trộn của các ký tự có kiểu chữ khác nhau, số, và dấu câu. Nó dễ nhớ vì là các chữ bắt đầu của từ "The password for this computer is too strong for you to guess!"
    - 5:*35pm&8/30—Thời gian và ngày tháng điện thoại với hai ký tự "đặc biệt" ngẫu nhiên
    - EPOcsoRYG5%4pp@.djr—sử dụng nhiều yếu tố bao gồm viết hoa và ký tự đặc biệt
    - BBslwys90!—gồm chữ hoa, số, và dấu câu. Cũng dễ nhớ, vì nó đại diện cho "Big Brother is always right (90°)!"

MjinllboyC@@33 – gồm chữ hoa và số và là từ viết tắt của 1 câu nói nào đó ít gặp và bạn yêu thích – bao gồm ký tự đặc biệt và số cuối cùng là ngày sinh và tháng sinh cộng lại

2.3.2.3. Build Model

2.3.2.3.1. Data Processing

Feature Name	Ý nghĩa
Password	Mật khẩu (theo nghĩa tiếng Việt) Bao gồm mật khẩu của tất cả các ứng dụng: website, mobile app, desktop app,... Mỗi mật khẩu có độ dài khác nhau

	Các ký tự trong mật khẩu bao gồm 26 ký tự trong bảng chữ cái tiếng Anh, các số từ 0 đến 9 và các ký tự đặc biệt - Tất cả mật khẩu đều mang tính ngẫu nhiên hoàn toàn không tuân theo bất cứ quy tắc nào

Drop các sample rỗng và map label theo nghĩa 0 là weak, 1 là medium và 3 là strong

```
data = data.dropna()

data["strength"] = data["strength"].map({0: "Weak",
                                         1: "Medium",
                                         2: "Strong"})
```

Dùng TfidVectorizer để tranform feature password

```
tdif = TfidfVectorizer(tokenizer=word)
x = np.array(data["password"])
x = tdif.fit_transform(x)
```

```
def word(password):
    character=[]
    for i in password:
        character.append(i)
    return character
```

2.3.2.3.2. Training Model

Khởi tạo model RandomForestClassifier

```
model = RandomForestClassifier()
model.fit(xtrain, ytrain)
```

Kết quả sau khi training

	precision	recall	f1-score	support
Medium	0.95	0.99	0.97	99519
Strong	0.96	0.91	0.93	16501
Weak	0.96	0.82	0.88	17908
accuracy			0.96	133928
macro avg	0.96	0.91	0.93	133928
weighted avg	0.96	0.96	0.95	133928

2.3.3. Convert image to text

2.3.3.1. Nhận dạng ký tự quang học OCR (Optical Character Recognition)

Để ngắn gọn, chúng ta có thể hiểu rằng đây là một ứng dụng công nghệ giúp chúng ta đọc text ở file ảnh. Được biết đến như là một công cụ scan kỹ thuật số chuyên nhận dạng các ký tự, chữ viết tay, chữ đánh máy, công nghệ này chuyên dùng để truyền tải, nhập liệu dữ liệu. Đặc biệt, ở OCR còn có khả năng kỹ thuật số dưới dạng nhiều tài liệu khác nhau như hóa đơn, hộ chiếu, danh thiếp, tài liệu,…

Cách thức hoạt động của OCR: đối với OCR thì khi nhận diện 1 ảnh viết tay hay 1 ảnh trang in hay trang in thì chúng sẽ được quét và lưu dưới dạng tệp TIF. Chúng ta có thể dễ dàng đọc hình ảnh này dưới màn hình hiển thị nhưng tuy nhiên tùy vào thuộc tính máy tính, nó sẽ tồn tại một loạt hình ảnh có chấm trắng hoặc chấm đen. Từ đó công nghệ nhìn vào từng dòng của hình ảnh để xác định các dấu có khớp nhay hay không.

Chi tiết hơn, một OCR sẽ hoạt động theo các bước như sau:

Thu nhận hình ảnh: một máy quét sẽ đọc tài liệu và chuyển đổi chúng thành dữ liệu nhị phân. Phần mềm OCR phân tích hình ảnh đã quét và phân loại vùng sáng làm nền và vùng tối làm văn bản.

Tiền xử lý, trước tiên, phần mềm OCR sẽ làm sạch hình ảnh và loại bỏ các lỗi để chuẩn bị cho bước đọc. Sau đây là một số kỹ thuật làm sạch của phần mềm OCR:

- Chỉnh thẳng hoặc nghiêng nhẹ tài liệu đã quét để khắc phục lỗi về căn chỉnh trong quá trình quét.
- Khử nhiễu đốm hoặc loại bỏ mọi đốm ảnh kỹ thuật số hay làm mịn các viền của hình ảnh văn bản.
- Làm sạch đường viền khung và đường thẳng trong hình ảnh.
- Nhận dạng chữ viết cho công nghệ OCR đa ngôn ngữ

Nhận dạng văn bản: hai loại thuật toán OCR hoặc quy trình phần mềm chính mà phần mềm OCR sử dụng để nhận dạng văn bản được gọi là so khớp mẫu và trích xuất đặc điểm.

So khớp mẫu: cách thức hoạt động của so khớp mẫu là tách biệt một hình ảnh ký tự, được gọi là hình dạng chữ và so sánh với một hình dạng chữ tương tự được lưu trữ. Tính năng nhận dạng mẫu chỉ hoạt động hiệu quả khi hình dạng chữ được lưu trữ có phông chữ và tỷ lệ tương tự với hình dạng chữ đầu vào. Phương thức này hoạt động tốt đối với hình ảnh quét từ tài liệu được đánh máy bằng phông chữ đã biết.

Trích xuất đặc điểm: trích xuất đặc điểm sẽ chia nhỏ hoặc phân tách hình dạng chữ thành các đặc điểm như nét thẳng, nét vòng khép kín, hướng nét và giao điểm nét. Sau đó, hệ thống sử dụng các đặc điểm này để tìm kết quả phù hợp nhất hoặc kết quả gần đúng nhất trong số các hình dạng chữ khác nhau được lưu trữ.

Hậu xử lý: sau khi phân tích, hệ thống sẽ chuyển đổi dữ liệu văn bản được trích xuất thành tệp trên máy tính. Một số hệ thống OCR có thể tạo các tệp PDF có chú thích bao gồm cả phiên bản trước và sau của tài liệu được quét.

Vì sao chúng ta nên sử dụng OCR và đây là một số lý do:

- Trợ giúp người mắc bệnh mù và khiếm thị: vì sao lại nói OCR có khả năng hỗ trợ người mắc bệnh mù và khiếm thị, những người có ảnh hưởng về thị giác. Bởi, OCR có khả năng quét và đọc các từ trên màn hình. Từ đây, những người gặp vấn đề thị giác có thể dễ dàng hiểu được chúng.
- Tìm kiếm và thực hành dữ liệu: thực tế, OCR có khả năng tạo ra những nội dung văn bản riêng của quét tài liệu giúp chúng có thể dễ dàng tìm kiếm và xác định vị trí tài liệu dựa trên từ khóa. Đồng thời, OCR cũng cho phép nhanh nhẹn hơn trong việc chỉnh sửa và xử lý văn bản.
- Cập nhập dữ liệu nhanh chóng: OCR đảm bảo chức năng cải thiện hiệu quả và nhanh chóng cho công việc văn phòng cũng như năng suất cao. Bởi, hầu hết trong quá trình làm việc ở văn phòng nhu cầu scan (quét) tài liệu ngày một lớn. Điều này sẽ giúp tiết kiệm thời gian, đồng thời cập nhập dữ liệu nhanh, chính xác nhất cho người dùng.

Những hạn chế của OCR

Bên cạnh những lợi ích to lớn mà OCR mang lại, cũng không thể tránh khỏi những hạn chế riêng như:

- Đa số những phần mềm, ứng dụng chứa OCR chỉ có khả năng nhận dạng chính xác khoảng 80-90% dựa vào hình ảnh rõ nét.

- Với những hình ảnh truy cập có màu nền và màu chữ khá tương đồng (không có sự chênh lệch lớn) điều này khiến OCR gặp khó khăn trong nhận dạng. Và tất nhiên, kết quả nhận dạng sẽ không được khả thi cho lắm.

Ngoài ra, ở thời điểm hiện tại khi các ngôn ngữ ngày càng trở nên phong phú thì công nghệ OCR lại chưa thể đáp ứng đa ngôn ngữ. OCR chưa hỗ trợ support cho tất cả ngôn ngữ.

2.3.3.2. Build Model

2.3.3.2.1. Data Processing

Feature Name (Mỗi ký tự tương ứng với 1 folder)	Ý nghĩa
#	Chứa các hình viết tay ký tự # (không dùng)
$	Chứa các hình viết tay ký tự $ (không dùng)
&	Chứa các hình viết tay ký tự & (không dùng)
@	Chứa các hình viết tay ký tự @ (không dùng)
0	Chứa các hình viết tay số 0 (theo dataset số 0 và chữ o là 2 ký tự giống nhau)
1	Chứa các hình viết tay số 1
...	[tương tự với các số từ 2 đến 8]
9	Chứa các hình viết tay số 9
A	Chứa các hình viết tay chữ A
..	[tương tự với các chữ từ B đến M]
N	Chứa các hình viết tay chữ N
P	Chứa các hình viết tay chữ P
...	[tương tự với các chữ từ Q đến Y]
Z	Chứa các hình viết tay chữ Z

Lấy tất cảc hình trong folder Train và resize thành 32 x 32 bỏ vào tập train_data cùng với label (tên folder)

```
dir = "../Dataset/Train/"
train_data = []
img_size = 32
non_chars = ["#","$","&","@"]
for i in os.listdir(dir):
    if i in non_chars:
        continue
    count = 0
    sub_directory = os.path.join(dir,i)
    for j in os.listdir(sub_directory):
        count+=1
        if count > 4000:
            break
        img = cv2.imread(os.path.join(sub_directory,j),0)
        img = cv2.resize(img,(img_size,img_size))
        train_data.append([img,i])
```

Lấy tất cả các hình trong folder Validation và resize thành 32 x 32 bỏ vào tập val_data cùng với label (tên folder)

```
val_dir = "../Dataset/Validation/"
val_data = []
img_size = 32
for i in os.listdir(val_dir):
    if i in non_chars:
        continue
    count = 0
    sub_directory = os.path.join(val_dir,i)
    for j in os.listdir(sub_directory):
        count+=1
        if count > 1000:
            break
        img = cv2.imread(os.path.join(sub_directory,j),0)
        img = cv2.resize(img,(img_size,img_size))
        val_data.append([img,i])
```

Xáo trộn các sample trong tập train_data và val_data

```
random.shuffle(train_data)
random.shuffle(val_data)
```

Tách sample và label ra thành 2 tập

```
train_X = []
train_Y = []
for features,label in train_data:
    train_X.append(features)
    train_Y.append(label)
```

```
val_X = []
val_Y = []
for features,label in val_data:
    val_X.append(features)
    val_Y.append(label)
```

Dùng LabelBinarizer để encode label

```
LB = LabelBinarizer()
train_Y = LB.fit_transform(train_Y)
val_Y = LB.fit_transform(val_Y)
```

Convert kích thước ảnh

```
train_X = np.array(train_X)/255.0
train_X = train_X.reshape(-1,32,32,1)
train_Y = np.array(train_Y)
```

```
val_X = np.array(val_X)/255.0
val_X = val_X.reshape(-1,32,32,1)
val_Y = np.array(val_Y)
```

2.3.3.2.2. Training Model

Khởi tạo model

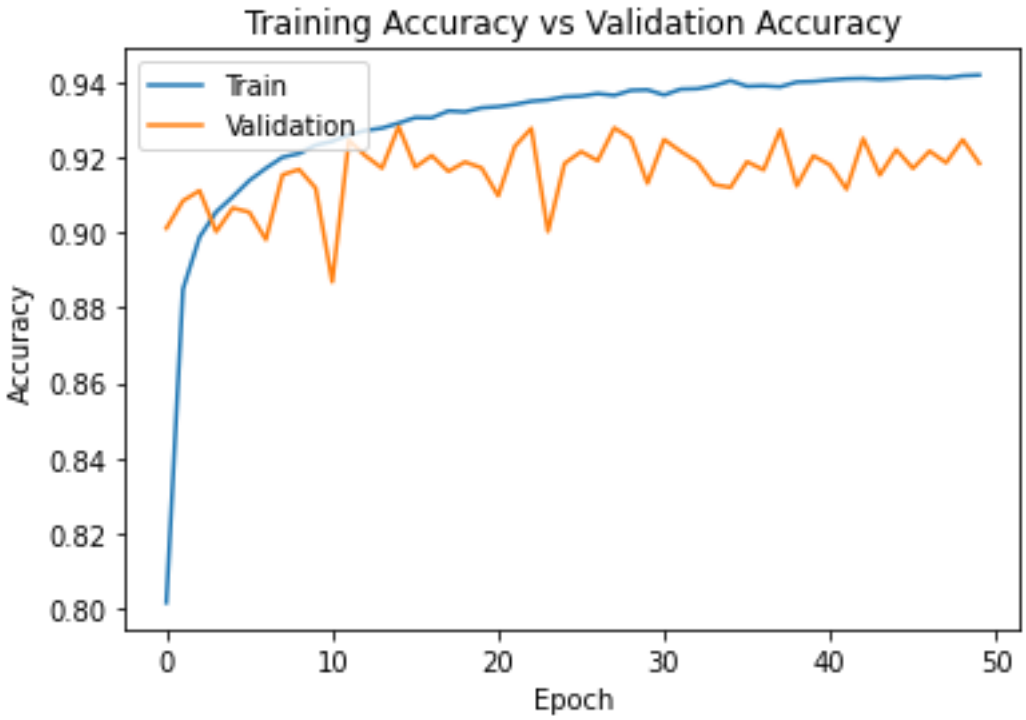
```
model = Sequential()

model.add(Conv2D(32, (3, 3), padding = "same", activation='relu', input_shape=(32,32,1)))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))

model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(35, activation='softmax'))
```

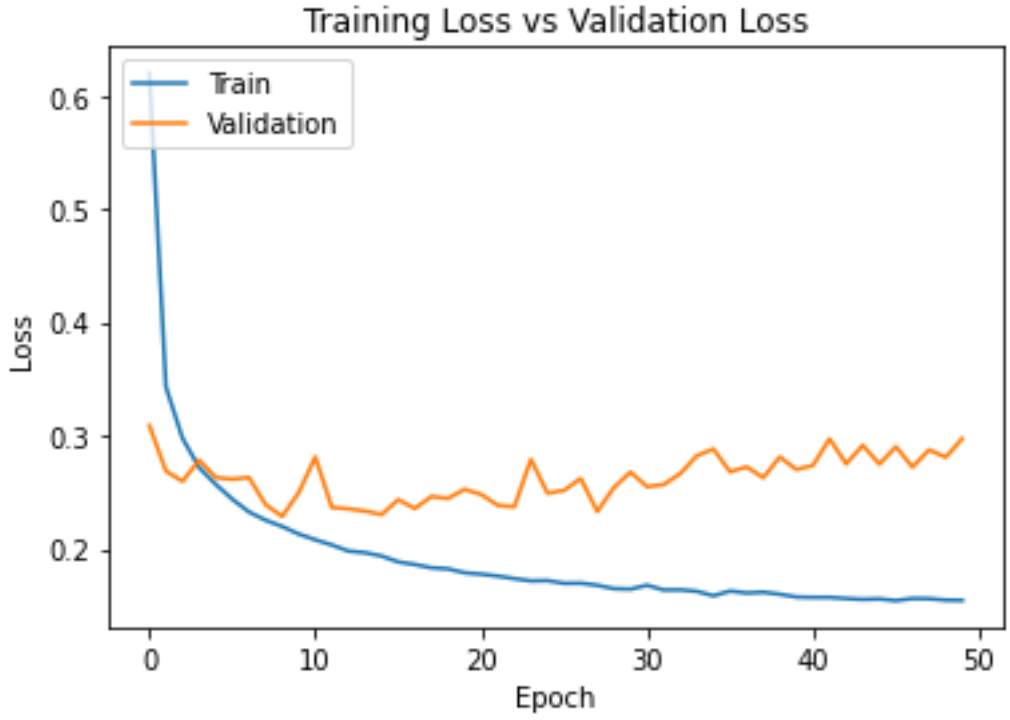
```
model.compile(loss='categorical_crossentropy', optimizer="adam",metrics=['accuracy'])
```

Summary model

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 32, 32, 32)        320

 max_pooling2d (MaxPooling2D  (None, 16, 16, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 14, 14, 64)        18496

 max_pooling2d_1 (MaxPooling  (None, 7, 7, 64)         0
 2D)

 conv2d_2 (Conv2D)           (None, 5, 5, 128)         73856

 max_pooling2d_2 (MaxPooling  (None, 2, 2, 128)        0
 2D)

 dropout (Dropout)           (None, 2, 2, 128)         0

 flatten (Flatten)           (None, 512)               0

 dense (Dense)               (None, 128)               65664

 dropout_1 (Dropout)         (None, 128)               0

 dense_1 (Dense)             (None, 35)                4515

=================================================================
Total params: 162,851
Trainable params: 162,851
Non-trainable params: 0
_________________________________________________________________
```

Kết quả sau khi training





3. Cách sử dụng

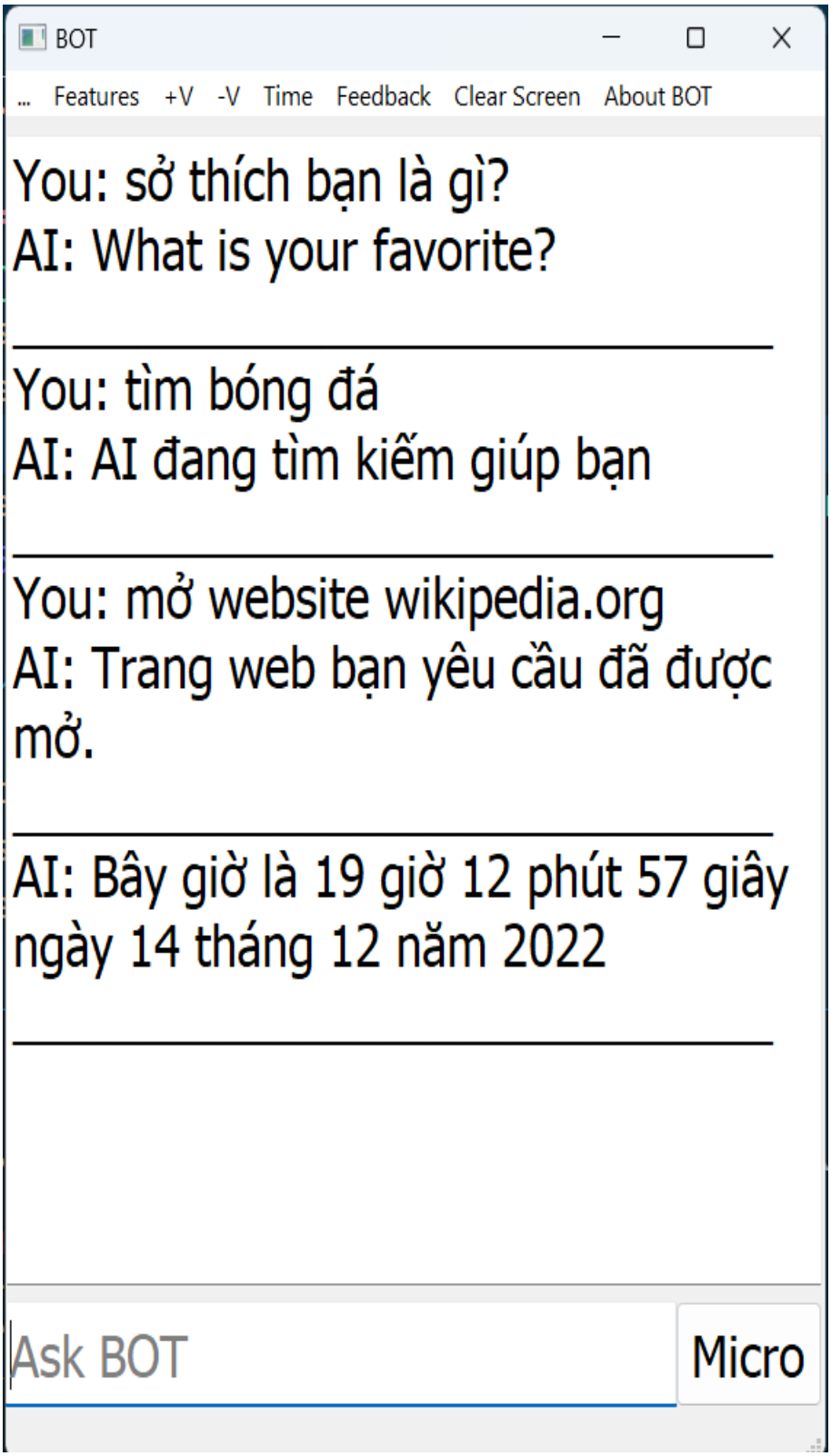
Có 3 cách sử dụng các chức năng trong ứng dụng: qua lời nói, qua gõ bàn phím và click vào nút chức năng

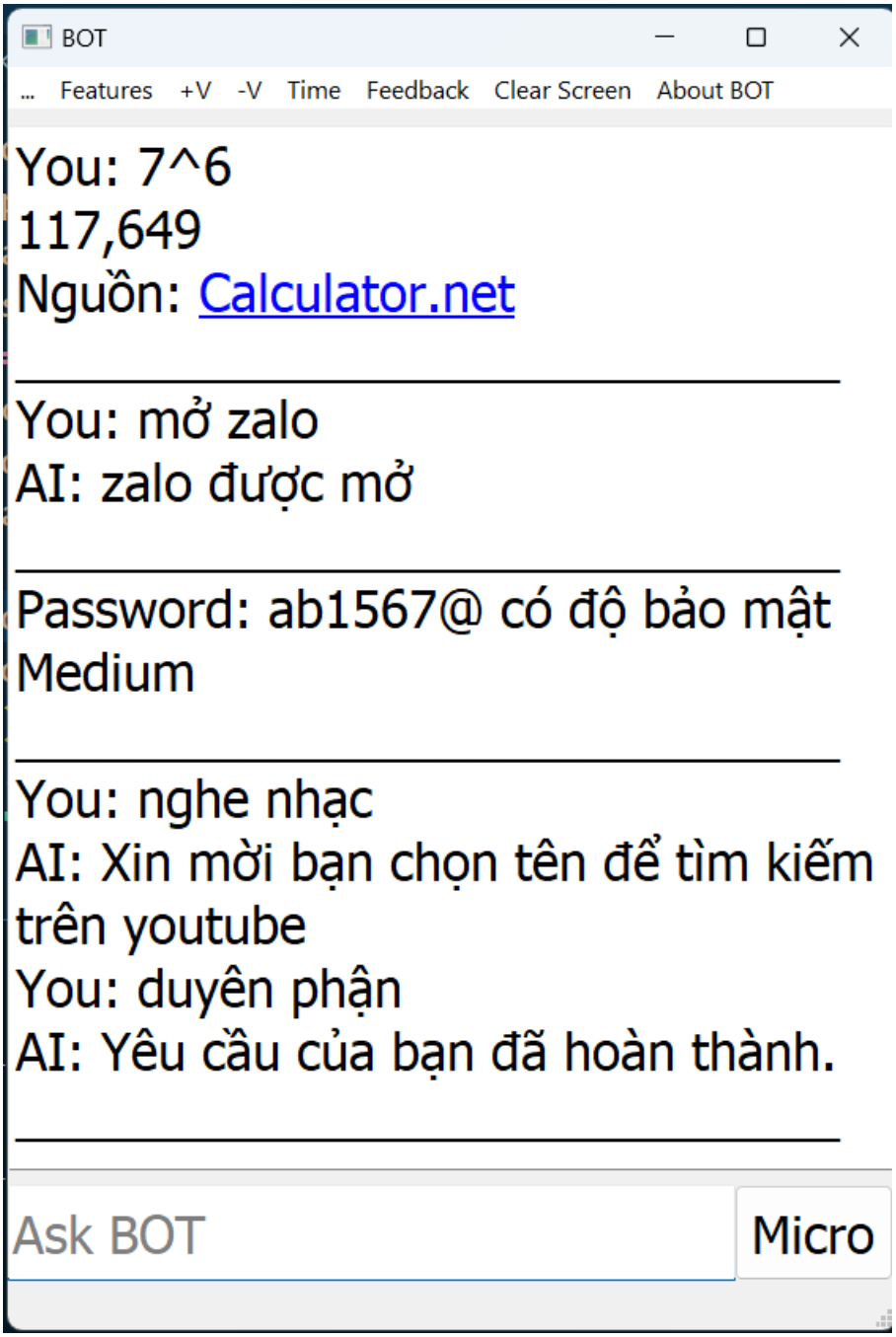
- Qua lời nói và qua gõ phím: khi người dùng bật micro nói và gõ trong ô input [Ask BOT] với những keyword tương ứng trong bảng bên dưới thì các chức năng tương ứng sẽ được thực hiện

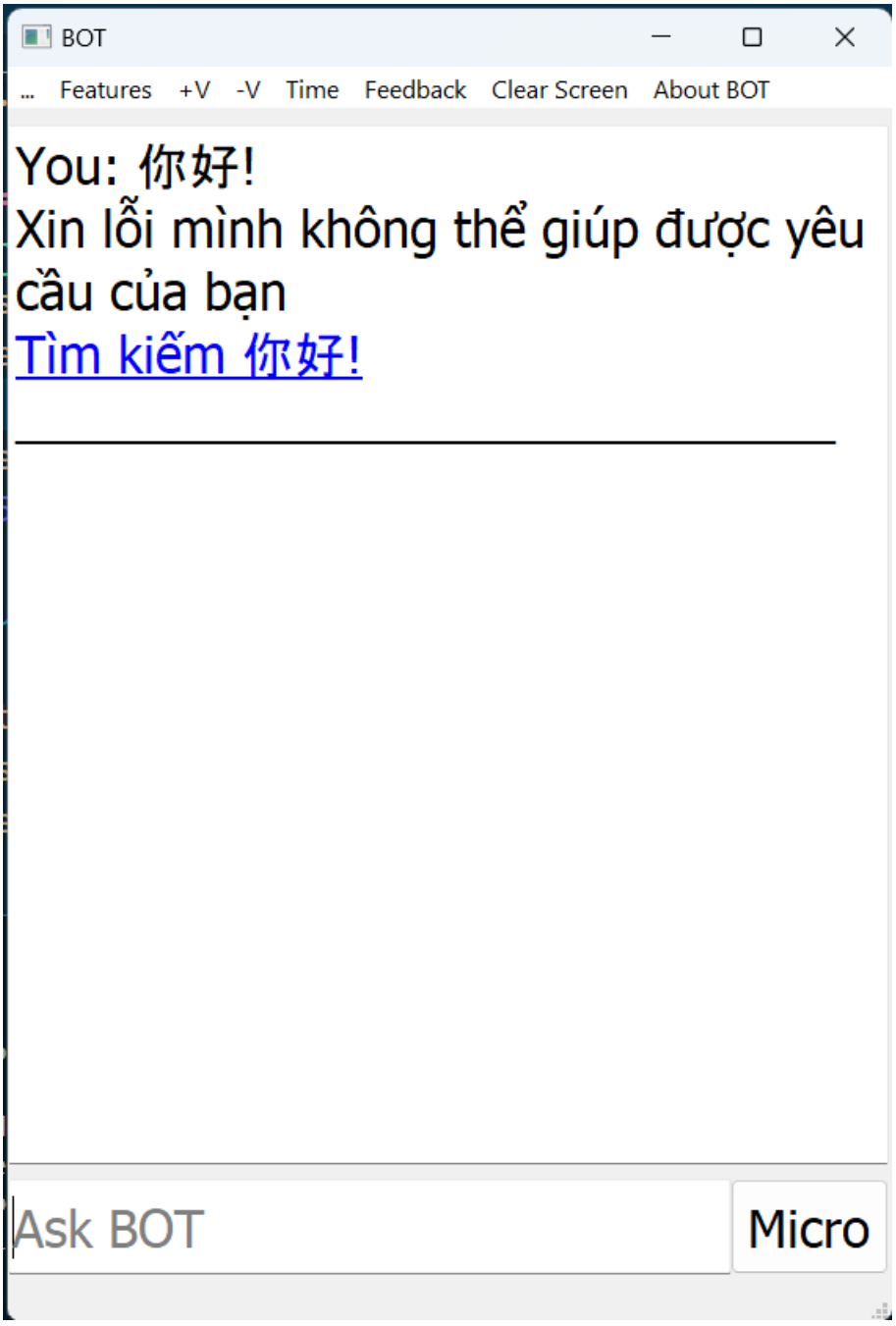
- Click vào nút chức năng: khi click vào nút chức năng, chức năng đó sẽ được thực hiện

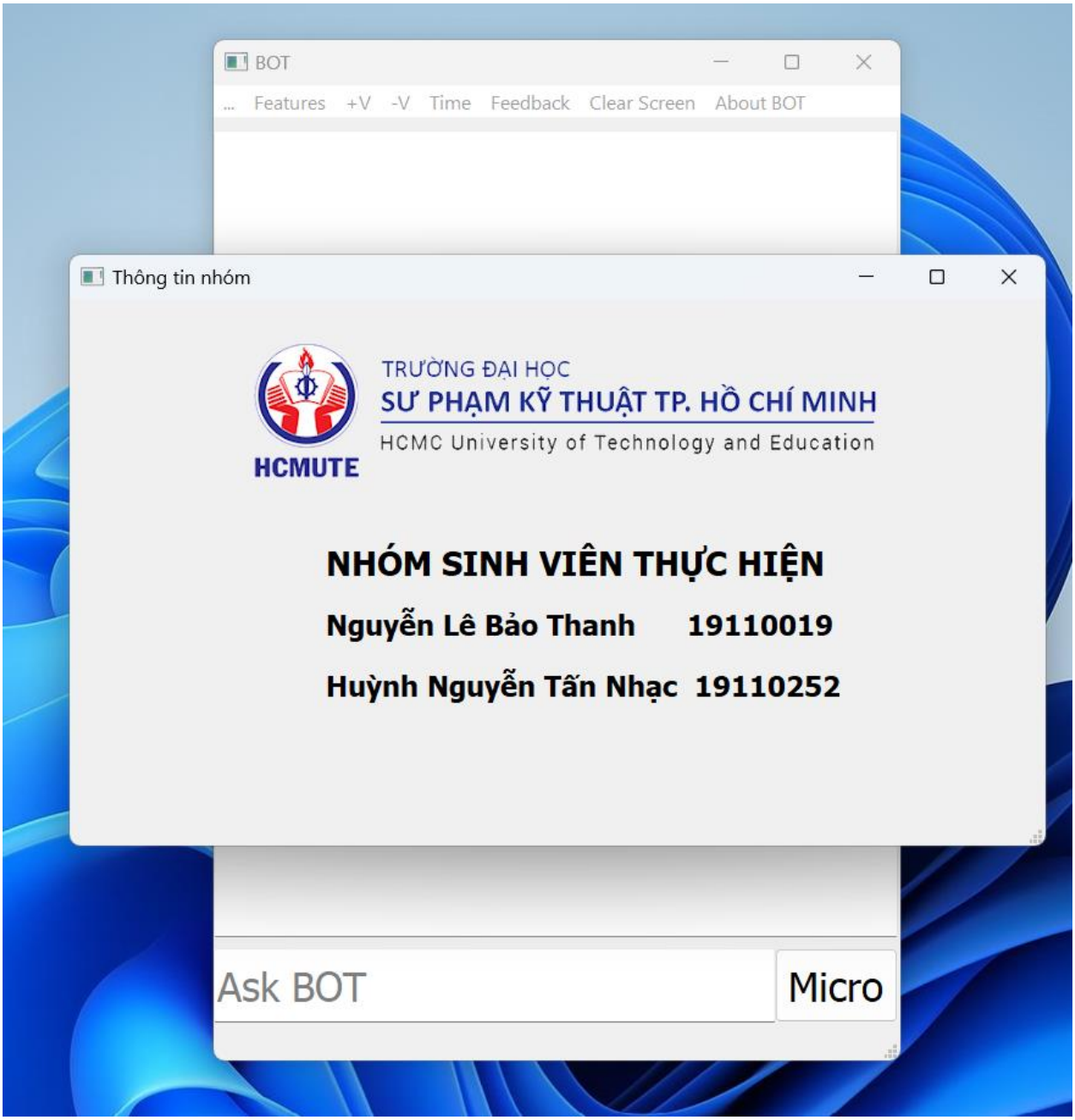
STT	**Tên chức năng**	**Cách sử dụng** qua lời nói (1), qua gõ bàn phím (2): **keyword** qua nút chức năng (3): **tên nút chức năng**	**Ghi chú (nếu có)***
1	Chào hỏi	(1) (2): xin chào, hello	
2	Xem thời tiết	(1) (2): thời tiết	
3	Xem thời gian	(1) (2): ngày mấy, mấy giờ, thứ mấy	
4	Mở ứng dụng	(1) (2): mở ứng dụng, mở phần mềm, mở	
5	Mở website	(1) (2): mở website	
6	Tìm kiếm trên Internet	(1) (2): tìm	
7	Tìm kiếm trên youtube	(1) (2): nghe nhạc, xem phim, mở youtube, bài hát	
8	Tìm kiếm trên wikipedia	(1) (2): là gì, là ai	
9	Tính toán	(2): tính, bằng, =	
10	Tăng/giảm âm lượng	(3): +V, -V	
11	Translate	(3): Translate	
12	Calculator	(3): Calculator	
13	Learn English	(3): Learn English	
14	Math Formulas	(3): Math Formulas	
15	Check URL	(3): Check URL	
16	Check password	(3): Check password	
17	Image to Text	(3): Image to Text	
18	Audio Book	(3): Audio Book	
19	Feedback	(3): Feedback	
20	Clear Screen	(3): Clear Screen	
21	Now	(3): Now	
22	Calendar	(3): Calendar	
23	Setting	(3): Setting	
24	About BOT	(3): About BOT	
25	About us	(3): About us	
26	Exit	(1) (2): hẹn gặp lại, tạm biệt, cảm ơn, thoát	
27	Tìm kiếm mở rộng	(1) (2); bao gồm các keyword không được liệt kê bên trêm	

Một số hình ảnh minh họa cách sử dụng ứng dụng




BOT
... Features +V -V Time Feedback Clear Screen About BOT
You: 7^6
117,649
Nguồn: Calculator.net
You: mở zalo
AI: zalo được mở
Password: ab1567@ có độ bảo mật Medium
You: nghe nhạc
AI: Xin mời bạn chọn tên để tìm kiếm trên youtube
You: duyên phận
AI: Yêu cầu của bạn đã hoàn thành.
Ask BOT
Micro


BOT
... Features +V -V Time Feedback Clear Screen About BOT
You: 你好!
Xin lỗi mình không thể giúp được yêu cầu của bạn
Tìm kiếm 你好!
Ask BOT
Micro



Hình 11: Hình ảnh giao diện thông tin nhóm

TỔNG KẾT

1. Đánh giá kết quả thực hiện

Xây dựng một trợ lý ảo sử dụng giọng nói quả thật là một việc không hề đơn giản nhưng không phải không thể không thực hiện được, chúng ta thấy rõ ràng hầu hết trên các thiết bị điện thoại thông minh ngày ngay đều được tích hợp các trợ lý ảo và chúng ta thấy rằng nó có khả năng giúp được con người khá nhiều.

Vậy việc nhóm cố gắng với đề tài xây dựng trợ lý ảo dưới sự hỗ trợ, hướng dẫn từ thầy Trần Nhật Quang là một đề tài vô cùng cuốn hút và có thể tiến xa hơn nữa trong tương lai, trong đề tài lần này nhóm đã thực hiện được một số tính năng cơ bản nhất định phải có trong trợ lý ảo như tìm kiếm, phản hồi kèm đó là tích hợp một số tính năng sử dụng deep learning như check password, check url, convert image to text. Xong nhóm nhận thấy rằng như vậy vẫn chỉ là những thứ cơ bản nhất và vẫn có một số ưu nhược điểm nhất định từ đó sẽ là khởi nguồn để nhóm có thể khắc phục những nhược điểm và bổ sung nhiều ưu điểm nữa cho đề tài ở khóa luận tốt nghiệp.

2. Ưu điểm và nhược điểm

Một số ưu điểm:

- Trợ lý ảo dễ sử dụng, dễ tiếp cận
- Có được một số tính năng cơ bản
- Đề tài được tính hợp một số tính năng deep learning
- Có thể sử dụng text-chat hoặc có thể sử dụng giọng nói để giao tiếp với trợ lý ảo.

Một số nhược điểm:

- Một trong những tính năng quan trọng là có thể trò chuyện **tốt** (tương tác giao tiếp) với người dùng là điều mà nhóm chưa thực hiện tốt.
- Nhóm còn buâng khuâng việc một số chức năng có thực sự cần thiết.
- Nhóm nhận thấy giao diện chưa được bắt mắt và cần khắc phục.
- Tính năng còn khá hạn chế
- Đôi khi phản hồi sẽ không như mong muốn của người dùng là điều mà nhóm nghĩ hầu hết các trợ lý ảo gặp phải.

3. Hướng phát triển của đề tài

Ở đề tài lần này nhóm có ý tưởng sẽ phát triển lên một trợ lý ảo có thể giao tiếp tốt với người dùng, tạo cảm giác cho người dùng như đang nói chuyện với một con người thực sự có tư duy, phản hồi tốt, kèm theo đó là thiết kế các giao diện nhận đầu vào từ người dùng và phản hồi đẹp hơn. Cuối cùng nhóm cũng muốn nỗ lực tìm ra những tính năng thực sự cần thiết cho người để tích hợp vào trợ lý ảo như, tìm đường tự động, thông báo tình trạng kẹt xe và tự tìm đường. Đây tạm thời là ý tưởng phát triển của nhóm, hi

vọng trong tương lai nhóm sẽ còn nghĩ ra được nhiều ý tưởng hợp để trợ lý ảo có thể đem lại nhiều lợi ích cho người dùng hơn.

TÀI LIỆU THAM KHẢO

[1] J. Brownlee, Deep Learning for Natural Language Processing: Develop Deep Learning Models for your Natural Language Problems, Machine Learning Mastery, 2017.

[2] Hinton, G.E. and Salakhutdinov, R.R., Reducing the dimensionality of data with neural networks, Science, 313(5786), 2006, pp. 504-507.

[3] Yann LeCun, Yoshua Bengio, Geoffrey Hinton, Deep learning, Nature, 521(7553), 2015, pp. 436-444.

[4] Oriol Vinyals, Quoc V Le, A Neural Conversational Model, Jul 22, 2015., pp. 1-8.

[5] E. D.Liddy, Natural Language Processing, Syracuse University 2001, pp. 1-15.

[6] TopDev Team, TensorFlow là gì? Tìm hiểu về TensoFlow từ A đến Z, November 5, 2021. https://topdev.vn/blog/tensorflow-la-gi/#cach-tensorflow-hoat-dong

[7] Admin, Cách Dùng Scikit-Learn – Tự Học TensorFlow, May 3, 2022.

https://tek4.vn/cach-dung-scikit-learn-tu-hoc-tensorflow

[8] Minh.Khai, "Tìm hiểu về thư viện Numpy trong Python," November 17, 2020. https://viblo.asia/p/tim-hieu-ve-thu-vien-numpy-trong-pythonphan-1-Do7542QXZM6#_doi-tuong-kieu-du-lieu-dtype-5

[9] Admin, Giới thiệu Matplotlib - Matplotib Cơ Bản, 2021.

https://vncoder.vn/bai-hoc/gioi-thieu-matplotlib-492

[10] N. V. Hiếu, Pandas Python Tutorial, 2 October,2018.

https://viblo.asia/p/pandas-python-tutorial-XL6lAxaDZek

[11] M. Duc, Seaborn trong Python là gì?, 2021.

https://tecktrending.com/seaborn-trong-python-la-gi/

[12] admin, Introduce about Keras, August 21, 2020.

https://trituenhantao.github.io/2020/08/21/keras-la-gi-gioi-thieu-ve-keras/

[13] Anonymous, "Độ mạnh của mật khẩu," April 9, 2022.

https://vi.wikipedia.org/wiki/Độ_mạnh_của_mật_khẩu