ĐẠI HỌC SƯ PHẠM THÀNH PHỐ HỒ CHÍ MINH

**KHOA CÔNG NGHỆ THÔNG TIN**

**🙡★🙣**

****

**BÁO CÁO ĐỀ TÀI**

**NGHIÊN CỨU KHOA HỌC**

**XÂY DỰNG HỆ THỐNG   
NHẬN DẠNG ĐỐI TƯỢNG   
SỬ DỤNG KHẨU TRANG**

Giảng viên hướng dẫn : ThS. LƯƠNG TRẦN NGỌC KHIẾT

Nhóm thực hiện :

+ LÊ TẤN LỘC 45.01.104.135

+ HUỲNH THANH PHONG 45.01.104.172

+ TRẦN THỊ TỨ LINH 45.01.104.127

+ PHẠM DUY MINH 45.01.104.145

**TP HỒ CHÍ MINH - 202**

**MỤC LỤC**

[**DANH MỤC HÌNH ẢNH** 3](#_Toc67469060)

[**LỜI CẢM ƠN** 1](#_Toc67469061)

[**CHƯƠNG 1. TỔNG QUAN** 2](#_Toc67469062)

[**1.1. Giới thiệu bài toán – Đặt vấn đề** 2](#_Toc67469063)

[Nhận dạng khẩu trang có tên (Face mask detection: FMD – còn được gọi là nhận dạng đối tượng khi đeo khẩu trang) là một bài toán chính thuộc lĩnh vực thị giác máy tính - Computer Vision. Đây là một bài toán với một bài toán lớn và một bài nhỏ, từ bài toán nhỏ là nhận dạng đối tượng dẫn đến bài toán lớn là nhận diện đối tượng khi đeo khẩu trang. Ngày nay, cùng với sự phát triển công nghệ kĩ thuật số mà trong vòng 5 năm trở lại đây, bài toàn đã được cộng đồng nghiên cứu quan tâm và cũng đạt nhiều thành tựu nổi bật. 2](#_Toc67469064)

[**1.2. Xử lý hình ảnh** 3](#_Toc67469065)

[**1.3. Nhận dạng đối tượng đeo khẩu trang.** 4](#_Toc67469066)

[**1.4. Phạm vi đề tài** 5](#_Toc67469067)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT** 6](#_Toc67469068)

[**2.1. Xử lý ngôn ngữ tự nhiên** 6](#_Toc67469069)

[**2.2. Trích xuất thông tin** 11](#_Toc67469070)

[**2.3. Tình hình nghiên cứu ngoài nước về bài toán NER** 12](#_Toc67469071)

[**3.1. Môi trường** 14](#_Toc67469072)

[3.1.1 Ngôn ngữ Python 14](#_Toc67469073)

[3.1.2 Bộ công cụ hỗ trợ NLTK (Natural Language Toolkit) 15](#_Toc67469074)

[3.1.2 Thư viện asyncio 17](#_Toc67469075)

[3.1.3 Thư viện pandas 18](#_Toc67469076)

[3.1.4 Thư viện nest\_asyncio 18](#_Toc67469077)

[**3.2. Dữ liệu** 18](#_Toc67469078)

[3.2.1 Yêu cầu dữ liệu 19](#_Toc67469079)

[3.2.2 Tiền xử lý dữ liệu 22](#_Toc67469080)

[**3.3. Phân tích dữ liệu** 25](#_Toc67469081)

[**CHƯƠNG 4: KẾT LUẬN** 28](#_Toc67469082)

[**4.1. Kết luận** 28](#_Toc67469083)

[**4.2. Hướng phát triển** 28](#_Toc67469084)

[**TÀI LIỆU THAM KHẢO** 29](#_Toc67469085)

**DANH MỤC HÌNH ẢNH**

[Hình 1: Thống kê các công trình nghiên cứu liên quan đến cụm từ “named entity recognition” trên Springer từ 2002 – tháng 11/2017 2](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048721)

[Hình 2: Ví dụ về cây cú pháp 9](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048722)

[Hình 3: Minh họa về khái niệm trích xuất thông tin (Information Extraction) 11](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048723)

[Hình 4: Minh họa về hoạt động của NER (Named Entity Recoginition) 12](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048724)

[Hình 5: IEEE Spectrum The Top Programming languages 2018 14](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048725)

[Hình 6: Trình tải xuống NLTK để chọn gói nào cần được cài đặt 16](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048726)

[Hình 7: Giao diện trang thu thập các văn bản pháp quy 19](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048727)

[Hình 8: Ví dụ về định dạng của một văn bản pháp quy (.htm) 20](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048728)

[Hình 9: : Thêm cấu trúc link download vào Regex trong Settings   
của Link Klipper 21](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048729)

[Hình 10: Giao diện khi mở file step01\_download\_textfile.py bằng Spyder   
(Python 3.7) 22](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048730)

[Hình 11: File "step01\_download\_textfile.py" đang được thực thi 23](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048731)

[Hình 12: Dữ liệu được tải về thành công được lưu trong folder “download” 24](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048732)

[Hình 13: Giao diện khi mở file "step02\_entity\_nlp.py" bằng Spyder (Python 3.7) 25](#_Toc41048733)

[Hình 14: Biểu đồ cột thống kê các từ có số lần xuất hiện nhiều nhất  
 trong văn bản pháp quy tháng 1 – năm 2019 26](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048734)

[Hình 15: Biểu đồ tròn phân tích nội dung của văn bản pháp quy   
tháng 1 - năm 2019 26](#_Toc41048735)

[Hình 16: Biểu đồ cột thống kê các từ có số lần xuất hiện nhiều nhất   
trong văn bản pháp quy tháng 2 – năm 2019 27](file:///E:\NCKH_2104\NCKH_BaoCaoNhom2.docx#_Toc41048736)

[Hình 17: Biểu đồ cột phân tích nội dung trong các văn bản pháp quy   
tháng 4 - 2019 27](#_Toc41048737)

# **LỜI CẢM ƠN**

Nghiên cứu được thực hiện tại Khoa Công nghệ thông tin – Trường Đại học Sư phạm Thành phố Hồ Chí Minh, dưới sự hướng dẫn khoa học của ThS Lương Trần Ngọc Khiết.

Trước tiên chúng em xin bày tỏ lòng biết ơn sâu sắc tới thầy ThS Lương Trần Ngọc Khiết đã đưa chúng em đến với lĩnh vực nghiên cứu này. Thầy đã tận tình giảng dạy, hướng dẫn chúng em tiếp cận và đạt được những kết quả nhất định trong nghiên cứu của mình. Thầy đã luôn tận tâm động viên, khuyến khích và chỉ dẫn giúp chúng em hoàn thành nghiên cứu này.

Chúng em xin bày tỏ lòng biết ơn tới các Thầy Cô thuộc Khoa Công nghệ thông tin và cán bộ Phòng Khoa học Công nghệ, khoa Công nghệ Thông tin – Trường Đại học Sư Phạm Thành phố Hồ Chí Minh đã tạo mọi điều kiện thuận lợi giúp đỡ chúng em trong quá trình học tập và nghiên cứu.

Sự hướng dẫn của thầy Th.s Lương Trần Ngọc Khiết đã tận tình hướng dẫn, động viên, cổ vũ của gia đình, bạn bè là nguồn động lực quan trọng để chúng em thực hiện đề tài nghiên cứu. Do kiến thức còn hạn chế, nên đề tài nghiên cứu của chúng em không tránh khỏi những thiếu sót, kính mong sự thông cảm, chỉ bảo của quý Thầy Cô.

Chúng em xin chân thành cảm ơn.

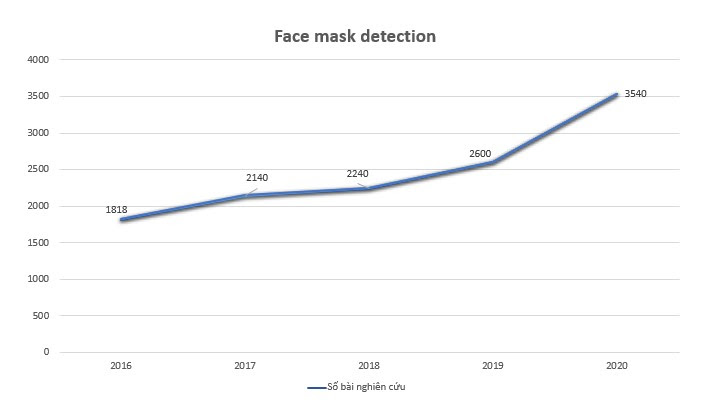
**Thay mặt nhóm thực hiện./**

Lê Tấn Lộc

# **CHƯƠNG 1. TỔNG QUAN**

**1.1. Giới thiệu bài toán – Đặt vấn đề**

Nhận dạng khẩu trang có tên (Face mask detection: FMD – còn được gọi là nhận dạng đối tượng khi đeo khẩu trang) là một bài toán chính thuộc lĩnh vực thị giác máy tính - Computer Vision. Đây là một bài toán với một bài toán lớn và một bài nhỏ, từ bài toán nhỏ là nhận dạng đối tượng dẫn đến bài toán lớn là nhận diện đối tượng khi đeo khẩu trang. Ngày nay, cùng với sự phát triển công nghệ kĩ thuật số mà trong vòng 5 năm trở lại đây, bài toàn đã được cộng đồng nghiên cứu quan tâm và cũng đạt nhiều thành tựu nổi bật.



Hình 1: Thống kê các công trình nghiên cứu liên quan đến cụm từ “Face mask detection” trên Springer từ 2016 – 2020

Face mask detection dần trở thành một chủ đề nóng cho cộng đồng nghiên cứu khoa học, thường xuyên được đề xuất tuong các hội nghị như: ICASSP, IFIP,CVPR,…

và trên các trang tập chí nổi tiếng: …

Chỉ trong vòng 5 năm trở lại đây, các thành tựu về nhận điện đối tượng đeo khẩu trang đã nhận được môt số lượng đáng kể, tuy nhiên bài toán này vẫn còn nhiều hạn chế và thách thức liên quan dữ liệu cực lớn, bảo mật thông tin, nhận diện sai khi có các ý tố khác tác động (vật thể khác che mặt mà không phải khẩu trang, ánh sáng,….)

Nhờ vào sự phát triển vượt bậc về các công nghệ kỹ thuật số, cùng với các công nghệ khác đã tạo cơ hội cho các lĩnh vực về xử lý hình ảnh dần ăn sâu vào đời sống con người về mọi mặt. Việc ứng dụng có hiệu quả các thành tựu này vào đời sống đã góp phần phát triển kinh tế và nâng cao đời sống xã hội, đồng thời nâng cao chất lượng cuộc sống của mọi người. Gần đây dịch covid đã là mối đe dọa đối với toàn nhân loại về vật chất lẫn tinh tha . Vì vậy bài toán nhận diện đối tượng có đeo khẩu trang rất cần thiết trong công cuộc phòng chống sự lây lan của dịch bệnh covid

* **Mục tiêu cụ thể**

+ Ở nghiên cứu này tập trung vào bài toán nhận diện những người có đeo khẩu trang đảm bảo hạn chế sự lây lan của dịch bệnh

+ Mục tiêu cụ thể của đề tài là phân tích, phát triển vấn đề, đề xuất giải pháp cho việc nhận diện các loại đối tượng có đeo khẩu trang thuộc miền dữ liệu trên

+ Xây dựng một chương trình có hiệu quả, đầy đủ và chính xác với mục đích hỗ trợ việc nhận dạng thực thể song xây dựng một hệ thống huấn luyện, thực nghiệm và ứng dụng.

**1.2. Xử lý hình ảnh**

[Xử lý hình ảnh](https://monkeylearn.com/natural-language-processing/) là một lĩnh vực của thị giác máy tính tập trung vào các ứng dụng trên hình ảnh thực. Trong thị giác máy tính thì xử lý hình ảnh là một trong những phần trọng tâm vì nó liên quan đến việc phải xác định, phân tích hình ảnh.

Xử lý hình ảnh là xác định, phân tích và nghiên cứu cấu trúc các điểm ảnh và tạo ra các hệ thống thông minh có khả năng nhận dạng đối tượng từ hình ảnh và video, giúp bạn giải quyết các vấn đề như [phân loại](https://monkeylearn.com/text-classification/" \t "_blank)từng đối tượng. Phân tích hình ảnh là một nhiệm vụ thiết yếu cho phép chúng ta nhận dạng ra các đối tượng ở trong hình ảnh hoặc video.

**1.3. Nhận dạng đối tượng đeo khẩu trang.**

Bài toán nhận dạng đối tượng đeo khẩu trang là bài toán xác định (phát hiện) những đối tượng đeo khẩu trang. Bài toán này sẽ phân tích các điểm ảnh dựa trên các thuộc tính cả việc nhận dạng các điểm ảnh đặc trưng của hình ảnh bằng phương pháp tính chập mạng nơ-ron (CNNs). Ví dụ trong trường hợp cung cấp các bức hình có một hay nhiều người đeo khẩu trang cho chương trình phân tích, sau đó lấy một bức ảnh có người đeo khẩu trang nào đó bất kì đưa cho chương trình kiểm tra, kỳ vọng sau khi kiểm tra là chương trình sẽ phản hồi được là có người đeo khẩu trang trong bước hình đó.

Đối tượng đầu vào cần được nhận diện là một hay nhiều hình ảnh hoặc nhận dạng trực tiếp thông qua các thiết bị ghi hình như camera,…. Một dữ liệu sau khi được kiểm tra sẽ được xếp vào một trong hai trạng thái là có đeo khẩu trang hoặ không đeo khẩu trang. Như vậy, để có thể nhận diện được một người có đeo khẩu trang hay không, thì hình ảnh là dữ liệu cơ bản nhất, cho dù chương trình nhận diện chạy trên bất kì cấu trúc nào.

Nhận diện người đeo khẩu trang góp phần không nhỏ trong việc đảm bảo an ninh trật tự trong công cộng. Chính vì vậy, bài toán này đã nhận được sự quan tâm sâu sắc của giới khoa học, đặc biệt là trong lĩnh vực thị giác máy tính.

**Các ứng dụng trong thế giới thực của nhận diện đối tượng đeo khẩu trang**

**Trước mùa dịch covid:**Trước khi đại dịch covid chưa được phát hiện, thì bài toàn này được sử dụng để hỗ trợ cho các thiết bị ghi hình phát hiện những hành vi bất thường như trộm cướp, hay có ý định thực hiện những hành vị sai trái, vì trên thực tế những hành vi này có xu hướng đeo khẩu trang để tránh bị phát hiện. Chính vì vậy mà bài toán này đã góp phần không nhỏ trong việc đảm bảo an ninh xã hội.

**Trong mùa dịch covid:** Điều này là quá rõ ràng vì sao bài toán nhận diện người đeo khẩu trang là một yêu cầu cấp bách, nếu bạn chưa biết thì đại dịch covid (COVID – 19) là đại dịch truyền nhiễm do vi-rút SARS-CoV-2 gây racó thể lây lan rất nhanh qua đường không khí, chỉ cần một cú hắt hơi người bị nhiễm vi-rút trên có thể dẫn đến cả cộng đồng bị nhiễm bệnh, nếu không phát hiện kịp thời thì nguy cơ tử vọng là rất cao, chính vậy bài toán này đã trở thành điểm nóng trong việc rà soát, phát hiện những người không đeo khẩu trang ở khoảng cách gần tới người bệnh ( gần hơn 2 mét), nhằm can thiệp, ngăn chặn kịp thời cơn đại dịch bùng phát trong cộng đồng. Chính vì thế mà từ đầu 2020 trở lại đây các công trình nghiên cứu liên quan đến nhận diện đối tượng mang khẩu trang đã tăng đột biến, nhận được sự quan tâm cực kì sâu sắc của các nhà nghiên cứu khoa học.

**1.4. Phạm vi đề tài**

Phạm vi nghiên cứu của đề tài là nhận diện được đối tượng có đeo khẩu trang hay không trong video thời gian thực (ghi hình trực tiếp) trên các thiết bị ghi hình.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## **2.1. Nhận diện khuôn mặt**

Xử lý ngôn ngữ tự nhiên (NLP) là một nhánh của Trí tuệ nhân tạo, tập trung vào việc nghiên cứu sự tương tác giữa máy tính và ngôn ngữ tự nhiên của con người. Mục tiêu của lĩnh vực này là giúp máy tính hiểu và thực hiện hiệu quả những nhiệm vụ liên quan đến ngôn ngữ của con người như: tương tác giữa người và máy, cải thiện hiệu quả giao tiếp giữa con người với con người, hoặc đơn giản là nâng cao hiệu quả xử lý văn bản và lời nói.

Xử lý ngôn ngữ tự nhiên ra đời từ những năm 1940, với rất nhiều công trình nghiên cứu theo hai hướng chính là:

- Automation và các mô hình xác suất (probabilistic models) vào những năm 1950

- Các phương pháp dựa trên ký hiệu (symbolic) và các phương pháp ngẫu nhiên (stochastic) vào những năm 1970. Giai đoạn tiếp theo (1970-1983) chứng kiến sự bùng nổ trong nghiên cứu về xử lý tiếng nói và ngôn ngữ. Ngày nay với sự phát triển nhanh chóng, học máy (machine learning) đã trở thành trung tâm của phần lớn các lĩnh vực thuộc khoa học máy tính, bao gồm xử lý ảnh và thị giác máy tính (computer vision), tin sinh học (bioinformatics), các hệ tư vấn (recommender systems), kỹ nghệ phần mềm, và cả xử lý ngôn ngữ tự nhiên.

**Một số bài toán xử lý ngôn ngữ tự nhiên kinh điển:**

Nhận dạng tiếng nói (speech recognition): từ sóng tiếng nói, nhận biết và chuyển chúng thành dữ liệu văn bản tương ứng.

Tổng hợp tiếng nói (speech synthesis): từ dữ liệu văn bản, phân tích và chuyển thành tiếng người.

Nhận dạng chữ viết (optical character recognition, OCR): từ một văn bản in trên giấy, nhận biết từng chữ cái và chuyển chúng thành một tệp văn bản trên máy tính.

Dịch tự động (machine translation): từ một tệp dữ liệu văn bản trong một ngôn ngữ, máy tính dịch và chuyển thành một tệp văn bản trong một ngôn ngữ khác

Tóm tắt văn bản (text summarization): từ một văn bản dài (mười trang chẳng hạn) máy tóm tắt thành một văn bản ngắn hơn (một trang) với những nội dung cơ bản

Tìm kiếm thông tin (information retrieval): từ một nguồn rất nhiều tệp văn bản hay tiếng nói, tìm ra những tệp có nội dung liên quan đến một vấn đề (câu hỏi) ta cần biết (hay trả lời)

Điển hình của công nghệ này là Google, một hệ tìm kiếm thông tin trên Web, mà hầu như chúng ta đều dùng thường xuyên. Cần nói thêm rằng mặc dù hữu hiệu hàng đầu như vậy, Google mới có khả năng cho chúng ta tìm kiếm câu hỏi dưới dạng các từ khóa (keywords) và luôn “tìm” cho chúng ta rất nhiều tài liệu không liên quan, cũng như rất nhiều tài liệu liên quan đã tồn tại thì Google lại tìm không ra

Trích chọn thông tin (information extraction): từ một nguồn rất nhiều tệp văn bản hay tiếng nói, tìm ra những đoạn bên trong một số tệp liên quan đến một vấn đề (câu hỏi) ta cần biết hay trả lời.

Phát hiện tri thức và khai phá dữ liệu văn bản (knowledge discovery and text data mining): Từ những nguồn rất nhiều văn bản thậm chí hầu như không có quan hệ với nhau, tìm ra được những tri thức trước đấy chưa ai biết. Đây là một vấn đề rất phức tạp và đang ở giai đoạn đầu của các nghiên cứu trên thế giới

Còn nhiều bài toán và công nghệ xử lý ngôn ngữ khác, như giao diện người máy bằng ngôn ngữ tự nhiên, các hệ hỏi đáp, các hệ sinh ra ngôn ngữ, …

**Các thuật ngữ cơ bản trong xử lý ngôn ngữ tự nhiên:**

**- Corpus:**

Corpus hiểu một cách đơn giản là các tập hợp dữ liệu văn bản, ngôn ngữ đã được số hóa, nó là các dữ liệu đã được xử lý, nó được dùng để kiểm chứng các quy luật của ngôn ngữ quá trình phân tích thông kê hay kiểm định giả thuyết thống kê của các mô hình dự đoán.

**- Phân tích hình thái (Morphological analysis):**

Trong ngôn ngữ học, hình thái học là môn học xác định, phân tích và miêu tả cấu trúc của hình vị (morpheme) và các đơn vị ý nghĩa khác như từ, phụ tố, từ loại, thanh điệu, hàm ý. Để có thể dễ hiểu ta có thể sử dụng từ vựng để thay thế cho hình vị. Phân tích hình thái trong NLP có thể kể đến các khâu xử lý như:

+ Tách từ (Word segmentation): Đối với một văn bản để xử lý, phân tích được ngôn ngữ thì việc tách từ là vô cùng cần thiết. Để hiểu được một văn bản ta cần hiểu được từng từ trong đoạn văn bản đó. Đối với những ngôn ngữ khác nhau lại có những đặc trưng khác biệt. Lấy ví dụ trong tiếng Anh các từ được phân cách nhau bởi dấu cách, tuy nhiên trong tiếng Thái, tiếng Trung, tiếng Nhật giữa các từ vựng lại không có khoảng trắng đối với tiếng Việt thì khoảng trắng lại chỉ có tác dụng ngăn cách tiếng chứ không phải ngăn cách giữa các từ vì thế việc xử lý tách từ là một công việc quan trọng.

+ Nhận diện riêng

+ Xử lý từ ghép

+ Phục hồi thể nguyên dạng (Lemmantization): Làm trở lại thể nguyên dạng của từ. Đây là việc cần thiết đối với các ngôn ngữ sử dụng hình vị (morpheme) để tạo ra ý nghĩa của từ. Ví dụ trong tiếng Anh sử dụng các phụ tố "s-es", "ed" sau các động từ để biến thể một từ đó đi, tương tự đối với tiếng Nhật. Tuy nhiên trong tiếng Việt lại không sử dụng morpheme để tạo ra ý nghĩa của từ nên công đoạn này không cần thiết.

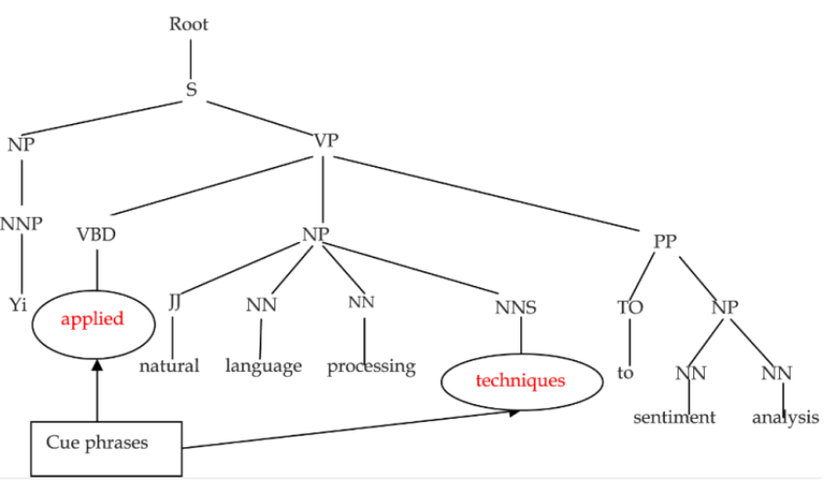
**- Phân tích ngữ pháp (Parser):**

Một số công việc cần làm khi phân tích ngữ pháp như:

+ Xác định từ loại (Part-of-Speech Tagging): Đối với các từ đã được tách công việc cần làm tiếp theo là gán nhãn xác định từ loại của từ, việc xác định từ loại của từ có vai trò quan trọng để hiểu nghĩa của từ và trong việc phân tích ngữ pháp hay các bước xử lý ngôn ngữ khác vì 1 từ có thể thể hiện dưới nhiều dạng từ loại khác nhau trong những trường hợp khác nhau.

+ Gán nhãn danh giới từ (chunking) để phân biệt đâu là điểm bắt đầu, kết thúc của một cụm từ.

+ Gán nhãn cây cú pháp (parse tree): Việc phân tích ngữ pháp của câu giúp cho việc hiểu ý nghĩa của câu trở nên dễ dàng hơn.

Một số ứng dụng thành công của xử lý ngôn ngữ tự nhiên (NLP):

Hình 2: Ví dụ về cây cú pháp

***Truy xuất thông tin*** (Information Retrieval – IR) có nhiệm vụ tìm các tài liệudưới dạng không có cấu trúc (thường là văn bản) đáp ứng nhu cầu về thông tin từ những nguồn tổng hợp lớn. Những hệ thống truy xuất thông tin phổ biến nhất bao gồm các công cụ tìm kiếm như Google, Yahoo, hoặc Bing search. Những công cụ này cho phép tiếp nhận một câu truy vấn dưới dạng ngôn ngữ tự nhiên làm đầu vào và cho ra một danh sách các tài liệu được sắp xếp theo mức độ phù hợp.

***Trích chọn thông tin*** (Information Extraction) nhận diện một số loại thực thể được xác định trước, mối quan hệ giữa các thực thể và các sự kiện trong văn bản ngôn ngữ tự nhiên. Khác với truy xuất thông tin trả về một danh sách các văn bản hợp lệ thì trích chọn thông tin trả về chính xác thông tin mà người dùng cần. Những thông tin này có thể là về con người, địa điểm, tổ chức, ngày tháng, hoặc thậm chí tên công ty, mẫu sản phẩm hay giá cả.

***Trả lời câu hỏi*** (QA) có khả năng tự động trả lời câu hỏi của con người ở dạng ngôn ngữ tự nhiên bằng cách truy xuất thông tin từ một tập hợp tài liệu. Một hệ thống QA đặc trưng thường bao gồm ba mô đun: Mô đun xử lý truy vấn (Query Processing Module) – tiến hành phân loại câu hỏi và mở rộng truy vấn; Mô đun xử lý tài liệu (Document Processing Module) – tiến hành truy xuất thông tin để tìm ra tài liệu thích hợp; và Mô hình xử lý câu trả lời (Answer Processing Module) – trích chọn câu trả lời từ tài liệu đã được truy xuất.

***Tóm tắt văn bản tự động*** là bài toán thu gọn văn bản đầu vào để cho ra một bản tóm tắt ngắn gọn với những nội dung quan trọng nhất của văn bản gốc. Có hai phương pháp chính trong tóm tắt, là phương pháp trích xuất (extractive) và phương pháp tóm lược ý (abstractive). Những bản tóm tắt trích xuất được hình thành bằng cách ghép một số câu được lấy y nguyên từ văn bản cần thu gọn. Những bản tóm lược ý thường truyền đạt những thông tin chính của đầu vào và có thể sử dụng lại những cụm từ hay mệnh đề trong đó, nhưng nhìn chung được thể hiện ở ngôn ngữ của người tóm tắt.

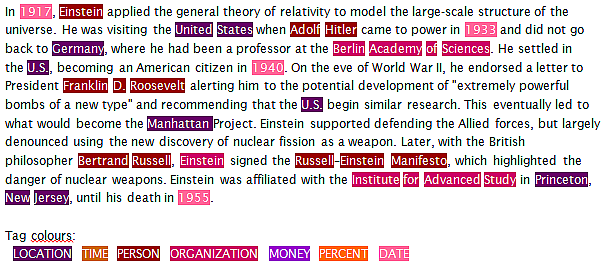
***Dịch máy*** (Machine translation – MT) là việc sử dụng máy tính để tự động hóa một phần hoặc toàn bộ quá trình dịch từ ngôn ngữ này sang ngôn ngữ khác. Các phương pháp dịch máy phổ biến bao gồm dịch máy dựa trên ví dụ (example-based machine translation – EBMT), dịch máy dựa trên luật (rule-based machine translation – RBMT), và dịch máy thống kê (statistical machine translation – SMT). Những nghiên cứu gần đây tập trung vào dịch máy thống kê bởi nhiều ưu điểm của nó so với các phương pháp khác. Dịch dựa trên từ (word-based translation), dịch dựa trên cú pháp (syntax-based translation), dịch dựa trên cụm từ (phrase-based translation), và dịch dựa trên cụm từ phân cấp (hierarchical phrase-based translation) là những mô hình dịch máy thống kê thành công nhất.

## **2.2. Nhận diện khẩu trang trên khuôn mặt**

**Information Extraction (IE: trích xuất thông tin) là quá trình phân tích, xử lý dữ liệu để trích xuất các thông tin hữu ích, có cấu trúc từ nguồn thông tin phi cấu trúc hoặc bán cấu trúc. Thông thường quá trình này bao gồm ba bước chính là: xác định thực thể (NER: Named Entity Recognition), xác định mối liên hệ (Relation Extraction) và trích xuất sự kiện (Event Extraction).

Hình 3: Minh họa về khái niệm trích xuất thông tin (Information Extraction)

Bước đầu tiên trong quá trình Trích xuất thông tin là việc xác định các thực thể xuất hiện trong đoạn dữ liệu rồi tiến hành phân loại thực thể này (được gọi là Named Entity Recoginition hay NER). Việc xác định thực thể có thể bắt đầu bằng việc xác định các từ trong đoạn văn bản có POS tag là Proper Name (như person, location, organization). Sau đó, việc phân loại thực thể (NER) có thể được thực hiện bằng việc áp dụng các sequence model như MEMM hoặc CRF. Tuy nhiên công việc này thường gặp khó khăn do tính không rõ ràng của các từ. Ví dụ như từ ‘Washington’ có thể được phân loại là Tên người (PERSON) như tổng thống Washington hoặc Địa danh (LOCATION) như thủ đô Washington,…

**Ngoài ra, một khái niệm khác có liên quan đến NER là Coreference Resolution. Coreference Resolution được sử dụng để tìm tất cả các từ đại diện cho cùng một thực thể nhất định trong đoạn văn bản.

Hình 4: Minh họa về hoạt động của NER (Named Entity Recoginition)

Bên cạnh vai trò trong việc xác định mối liên hệ (Relation Extraction) và phát hiện sự kiện (Event Extraction) thì NER còn được áp dụng trong nhiều bài toán liên quan đến Xử lý ngôn ngữ tự nhiên .

## **2.3. Tình hình nghiên cứu ngoài nước về bài toán Face Detection**

**Học có giám sát:** Hiện nay phần lớn các công trình giải quyết bài toán trích rút thực thể NER thường sử dụng kỹ thuật học máy có giám sát như mô hình Markov ẩn, cây quyết định, mô hình Maximum entropy, máy vector hỗ trợ (SVM), mô hình trường ngẫu nhiên có điều kiện CRF. Nhược điểm của học có giám sát là đòi hỏi tập dữ liệu huấn luyện gán nhãn bằng tay cực lớn.

**Học không giám sát:**

Để giải quyết vấn đề gán nhãn bằng tay một số nhà nghiên cứu đã sử dụng phương pháp học không giám sát. Các hệ hống học máy không giám sát không đòi hỏi dữ liệu huấn luyện, nhưng khả năng phát hiện thực thể không cao và độ chính xác thấp.

**Học bán giám sát:** Phương pháp học bán giám sát có thể khắc phục được nhược điểm của hai phương pháp trên. Phương pháp học bán giám sát huấn luyện hệ thống sử dụng cả dữ liệu gán nhãn và không gán nhãn. Có nhiều nghiên cứu về trích rút thực thể sử dụng phương pháp học bán giám, trong đó có Liao:

Liao sử dụng phương pháp học bán giám sát để huấn luyện hệ thống. Hệ thống ban đầu có một tập dữ liệu được gán nhãn thực thể với số lượng nhỏ, sử dụng mô hình CRF để huấn luyện hệ thống. Kết quả của quá trình huấn luyện được sử dụng để phân loại các thực thể trên tập dữ liệu chưa được gán nhãn. Sau đó tập dữ liệu và tập đặc trưng được cải tiến bằng cách tìm các dữ liệu được phân loại có độ tin cậy thấp bởi mô hình CRF bằng tập dữ liệu được gán nhãn bằng tay kích cỡ nhỏ. Tiếp theo, sử dụng mô hình này để phân loại dữ liệu chưa có nhãn và sau đó tập dữ liệu đã được cải thiện và tập đặc trưng được cải thiện bằng cách áp dụng cho các dữ liệu chưa được gán nhãn được phân loại có độ tin cậy thấp nhờ mô hình học phân lớp được huấn luyện theo tập dữ liệu đã được gán nhãn thủ công có kích thước nhỏ. Liao đề xuất các tri thức để tìm các dữ liệu huấn luyện mới, các tri thức bên ngoài được sử dụng là hai giả định ngôn ngữ sau đây:

- Giả định xuất hiện nhiều lần: Những cụm từ (viết hoa) giống hệt nhau cùng xuất hiện trong một văn bản thường có kiểu thực thể giống nhau. Chẳng hạn, trong một văn bản tài chính, *John* có thể xuất hiện nhiều lần với cùng một tên là *“John”*

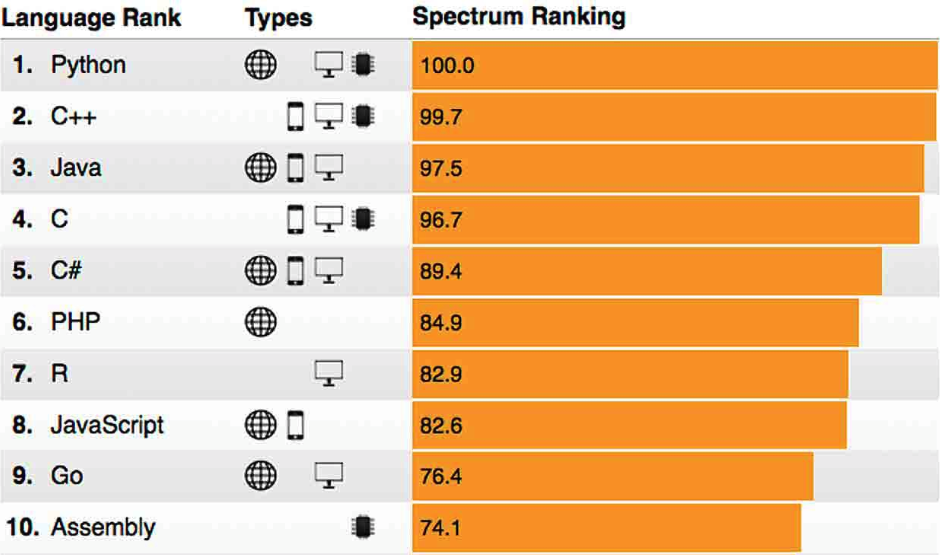
- Giả định ngữ cảnh: Các thực thể như con người, tổ chức, địa điểm thường có các ngữ cảnh xung quanh giúp chúng ta nhận biết được chúng một cách dễ dàng. Chẳng hạn, tên một tổ chức thường được đi cùng bởi các hậu tố như: *Inc., Co.,* …; tên một người thường được đi cùng bởi các tiền tố như: *Mr.*, *CEO*,…

# **CHƯƠNG 3: THỰC NGHIỆM CHƯƠNG TRÌNH**

## **3.1. Môi trường**

### 3.1.1 Ngôn ngữ Python

Python là một ngôn ngữ lập trình thông dịch, hướng đối tượng, ngôn ngữ lập trình cấp cao được giải thích với ngữ nghĩa động. Python với triết lý thiết kế của nó rất thuận tiện cho việc đọc hiểu code, đơn giản và rõ ràng được thiết kế bởi **Guido van Rossum**. Thiết kế bắt đầu vào cuối những năm 1980 và được phát hành lần đầu tiên vào tháng**2 năm 1991**. Đến nay thì cộng đồng người sử dụng ngôn ngữ này rất đông, nếu so sánh từ bảng xếp hạng các ngôn ngữ năm 2018 thì Python đã leo lên vị trí số 1 trên bảng xếp hạng những ngôn ngữ lập trình phổ biến.

Hiện nay, Python cũng là một ngôn ngữ rất phát triển trong lĩnh vực Data Science và Machine Learning. Python cũng cung cấp những hàm và thư viện xử lý ngôn ngữ tuyệt vời. Chính vì vậy, Python là một lựa chọn hợp lý khi thực hiện xử lý ngôn ngữ tự nhiên và phân tích dữ liệu văn bản.

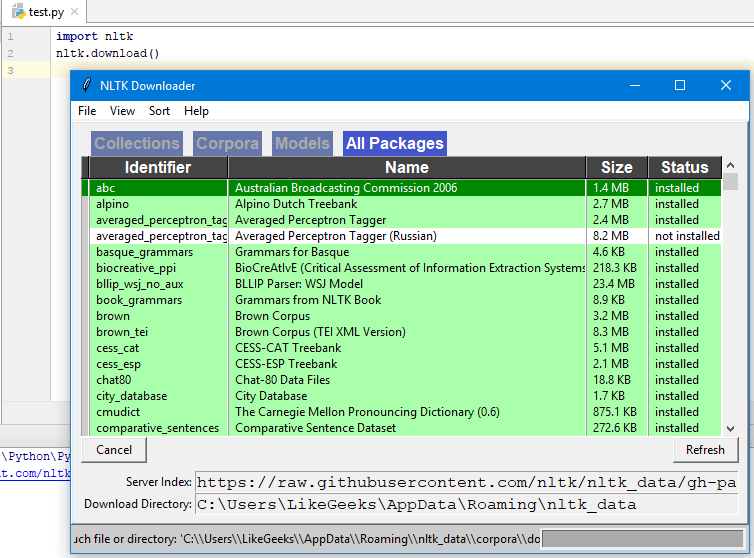
Hình 5: IEEE Spectrum The Top Programming languages 2018

### 3.1.2 Bộ công cụ hỗ trợ NLTK (Natural Language Toolkit)

Bộ công cụ ngôn ngữ tự nhiên (NLTK) là một nền tảng hàng đầu để xây dựng các chương trình Python để làm việc với dữ liệu ngôn ngữ của con người (Xử lý ngôn ngữ tự nhiên). Nó cung cấp các giao diện dễ sử dụng cho [hơn 50 tài nguyên ngôn ngữ và từ vựng](http://nltk.org/nltk_data/) như WordNet, cùng với bộ thư viện xử lý văn bản để phân loại, mã thông báo, xuất phát, gắn thẻ, phân tích cú pháp và lý luận ngữ nghĩa, trình bao bọc cho các thư viện NLP công nghiệp, và một [diễn đàn thảo luận](http://groups.google.com/group/nltk-users) tích cực . NLTK nhằm hỗ trợ nghiên cứu và giảng dạy trong NLP hoặc các lĩnh vực liên quan chặt chẽ, bao gồm ngôn ngữ học thực nghiệm, khoa học nhận thức, trí tuệ nhân tạo, truy xuất thông tin và học máy.

Cài đặt NLTK:

Các yêu cầu cụ thể cài đặt bộ thư viện NLTK tham khảo tại địa chỉ sau: <http://www.nltk.org/install.html>

NLTK là một bộ thư viện, chương trình hỗ trợ xử lý ngôn ngữ tự nhiên (NLP) bằng tất cả các kỹ thuật có thể có một cách trực tiếp hoặc gián tiếp trong ngôn ngữ python. Nó bao gồm tất cả các ví dụ, dữ liệu mẫu và các ứng dụng trình diễn bằng đồ họa. Một số modules quan trọng của NLTK:

Hình 6: Trình tải xuống NLTK để chọn gói nào cần được cài đặt

|  |  |  |
| --- | --- | --- |
| **Chức năng xử lý** | **Diễn giải** | **Module** |
| Accessing corpora | Chuẩn hóa giao diện đoạn văn (corpora) và từ vựng (lexicons). | nltk.corpus |
| String processing | Tách câu, tách từ (token) và họ từ (stemmers) | nltk.tokenize  nltk.stem |
| Part-of-speech tagging | Gán nhãn từ: n-gram, backoff, Brill. HMM, TnT | nltk.tag |
| Classification | Phân loại:  cây quyết định  maximum entropy  naive Bayes  EM  k-means | nltk.classify  nltk.cluster |
| Chunking | regular expression  n-gram  named-entity | nltk.chunk |
| Parsing | Chart  feature-based  unification  probabilistic, dependency | nltk.parse |
| Semantic interpretation | Lambda calculus, first-order logic, model checking | nltk.sem  nltk.inference |
| Evaluation | precision, recall, agreement coeficients | nltk.metrics |
| Probability and estimation | Frequency distributions, smoothed probability distributions | nltk.probability |
| Applications | Graphical concordancer, parsers, WordNet browser, chatbots | nltk.app  nltk.chat |
| Linguistic fieldwork | Manipulate data in SIL Toolbox format | nltk.toolbox |

Ngoài ra NLTK còn cung cấp giao diện giao tiếp với nhiều thư viện xử lý khác.

### 3.1.2 Thư viện asyncio

Cung cấp cơ sở hạ tầng để viết mã đồng thời một luồng bằng cách sử dụng coroutines, truy cập I / O ghép kênh qua ổ cắm và các tài nguyên khác, chạy máy khách và máy chủ mạng và các nguyên hàm khác có liên quan.

### 3.1.3 Thư viện pandas

Pandas là một thư viện mã nguồn mở được xây dựng dựa trên NumPy, sử dụng thao tác và phân tích dữ liệu, được thiết kế để cho phép bạn làm việc với dữ liệu được gắn nhãn hoặc quan hệ theo cách trực quan hơn.

Có thể xử lý tập dữ liệu khác nhau về định dạng: chuỗi thời gian, bảng không đồng nhất, ma trận dữ liệu.

Khả năng import dữ liệu từ nhiều nguồn khác nhau như CSV, DB/SQL…

Có thể xử lý vô số phép toán cho tập dữ liệu: subsetting, slicing, filtering, merging, groupBy, re-ordering, and re-shaping,..

Xử lý dữ liệu mất mát theo ý người dùng mong muốn.

Xử lý, phân tích dữ liệu tốt như mô hình hoá và thống kê.

Tích hợp tốt với các thư viện khác của python.

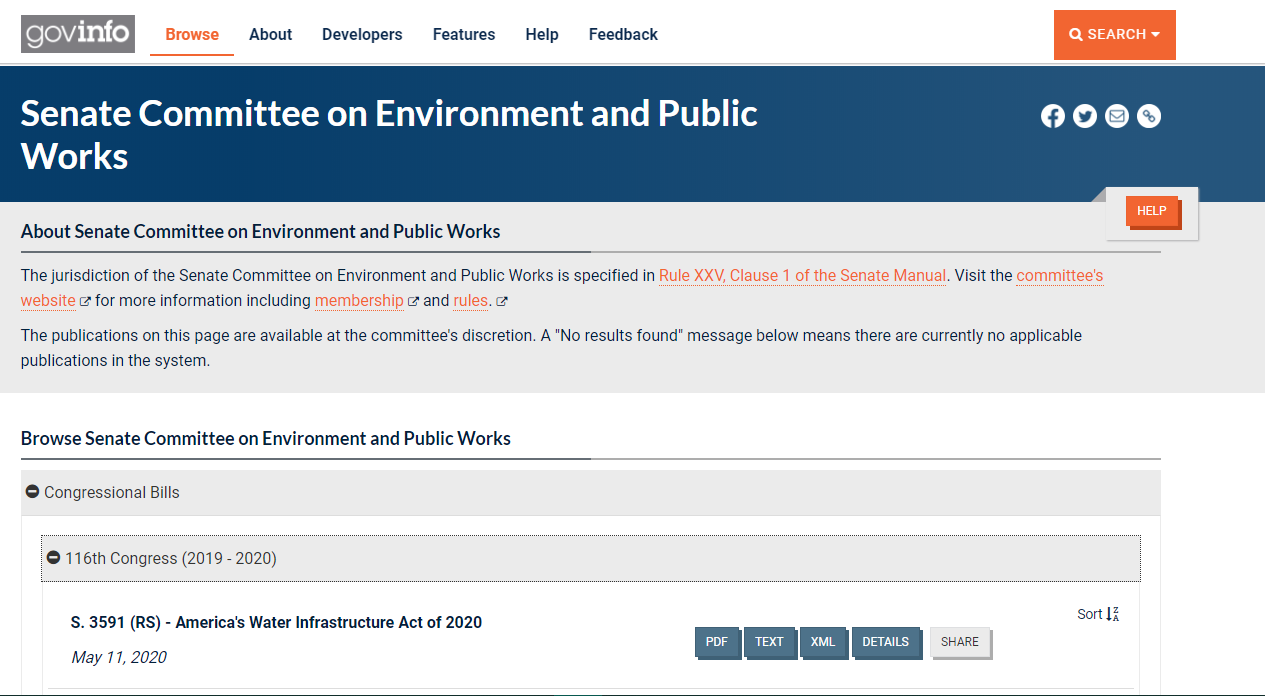
Cung cấp hiệu suất tốt.

### 3.1.4 Thư viện nest\_asyncio

Theo thiết kế, asyncio không cho phép vòng lặp sự kiện của nó được lồng vào nhau. Vì vậy, nest\_asyncio bổ sung cho module asyncio, cho phép sử dụng 2 lệnh asyncio.run và loop.run\_until\_complete lồng nhau.

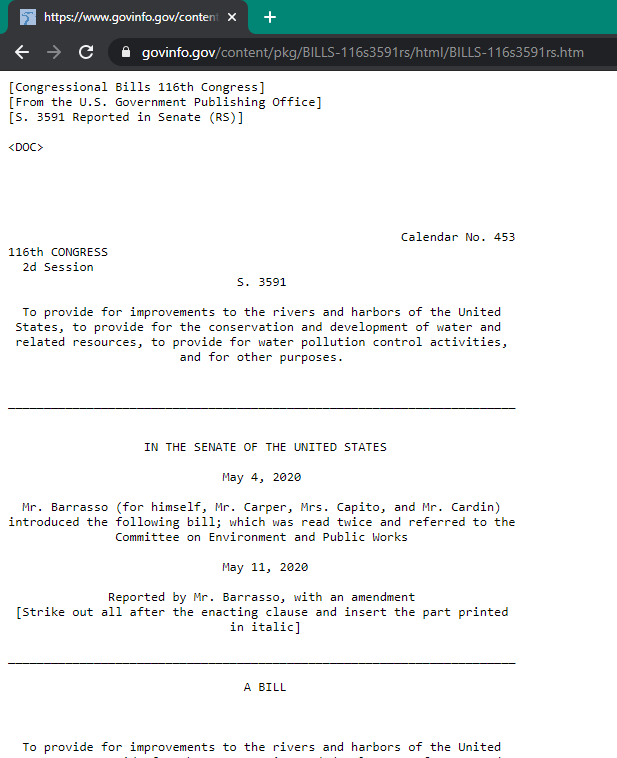
## **3.2. Dữ liệu**

Các dữ liệu về văn bản pháp quy của chính phủ Mỹ được thu thập tại địa chỉ <https://www.govinfo.gov/>

**Số lượng văn bản pháp quy của chính phủ Mỹ tăng lên nhanh chóng qua từng năm vì vậy lượng dữ liệu trở nên vô cùng lớn. Chính vì thế, ở phần thực nghiệm này chỉ sử dụng các văn bản pháp quy thuộc chủ để *“*[*Environment and Public Works*](https://www.govinfo.gov/committee/senate-environment)*”* năm 2019 – 2020.

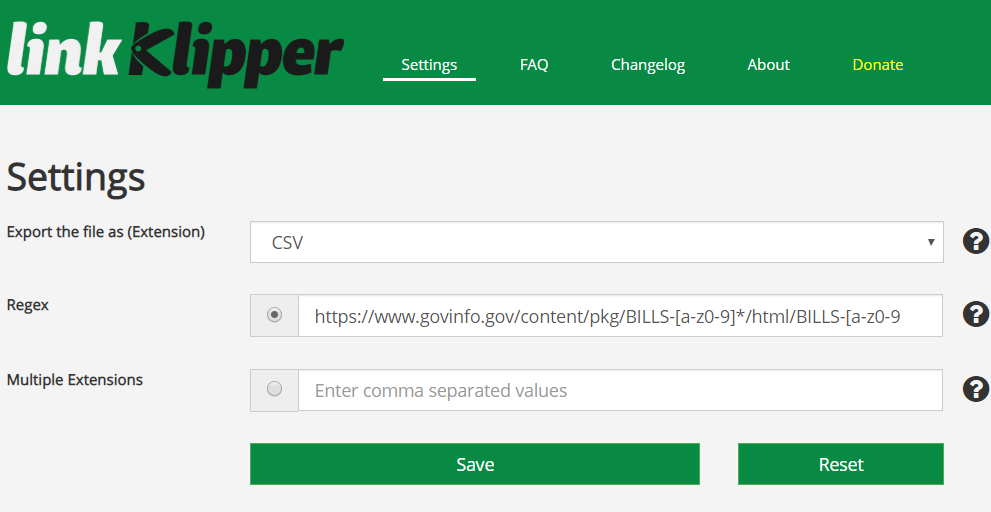
Hình 7: Giao diện trang thu thập các văn bản pháp quy

### 3.2.1 Yêu cầu dữ liệu

Thực hiện thu thập dữ liệu thuộc chủ đề *“Senate Committee on Environment and Public Works”* năm 2019 – 2020. Các văn bản được tải xuống dưới dạng file text (.htm) như định dạng dưới đây:

Hình 8: Ví dụ về định dạng của một văn bản pháp quy (.htm)

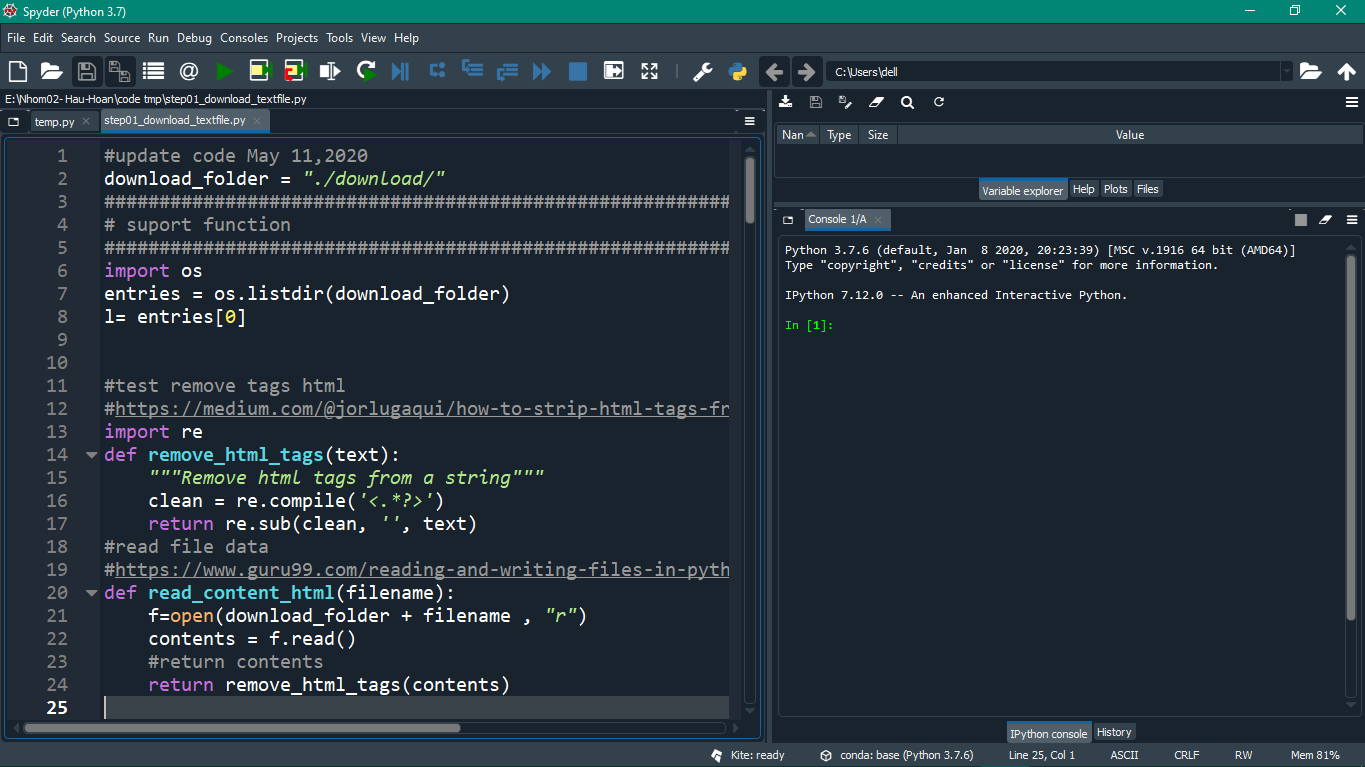
Quá trình tải xuống dữ liệu được thực hiện như sau:

Sử dụng tiện ích mở rộng “Link Klipper – Extract all links” của Google để tiến hành lấy các ID link của các văn bản pháp quy thuộc năm 2019 – 2020. **

Hình 9: : Thêm cấu trúc link download vào Regex trong Settings của Link Klipper

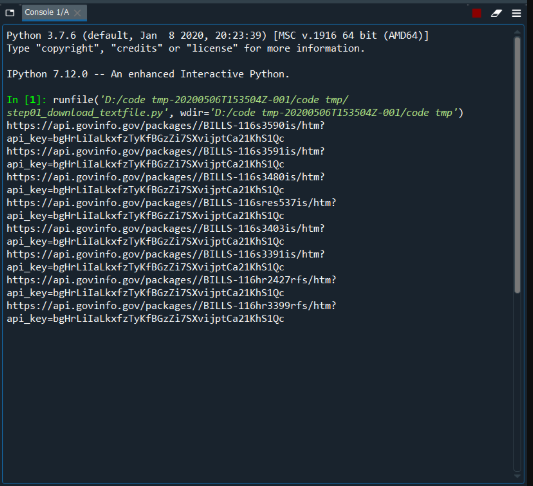
Trong Cài đặt của “Link Klipper”, thêm cấu trúc link *[https://www.govinfo.gov/content/pkg/BILLS-[a-z0-9]\*/html/BILLS-[a-z0-9](https://www.govinfo.gov/content/pkg/BILLS-%5ba-z0-9%5d*/html/BILLS-%5ba-z0-9?fbclid=IwAR0MafueRKHSpJB3o_O0z9lZogVcmNtat8VrSbOzMlCzexrwhOPMxJnwgGE" \t "_blank)]\** vào mục Regex để lấy được tất cả ID link của các văn bản pháp quy, các ID link này sẽ được lưu vào file “linkdata.csv”.

### 3.2.2 Tiền xử lý dữ liệu

**Mở file *“step01\_download\_textfile.py”*:

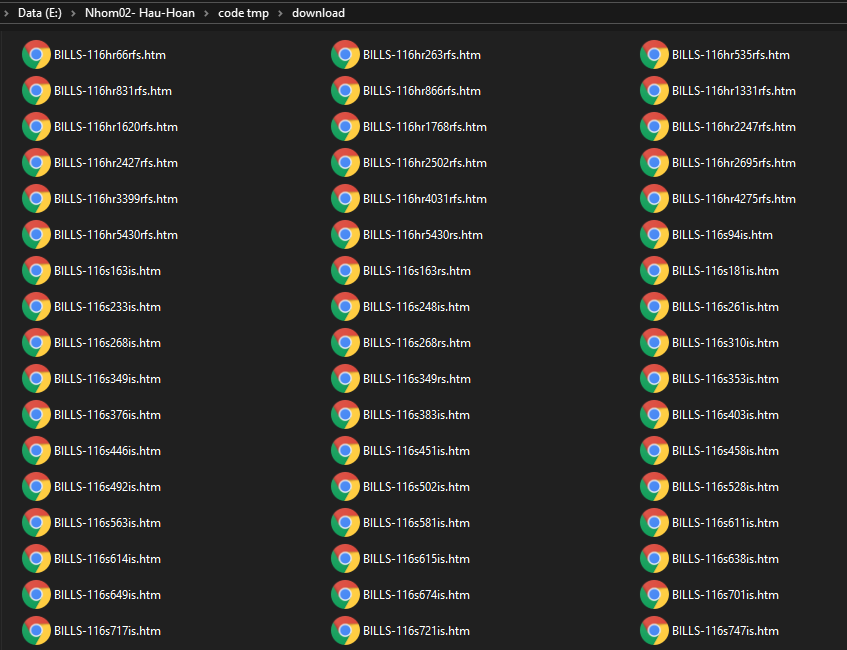
Hình 10: Giao diện khi mở file step01\_download\_textfile.py bằng Spyder (Python 3.7)

Tiến hành chạy file *“step01\_download\_textfile.py”*:



Hình 11: File "step01\_download\_textfile.py" đang được thực thi

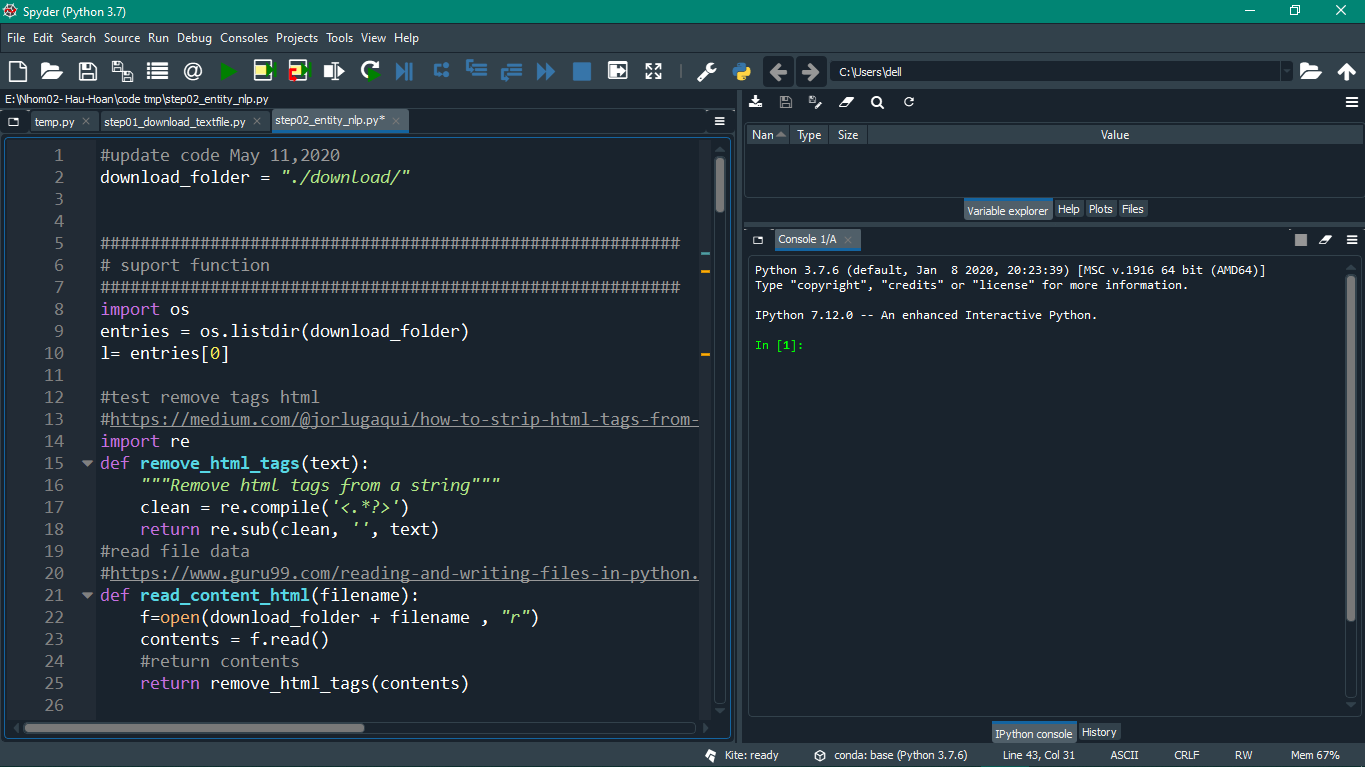
Sau khi chương trình thực thi xong, tất cả các văn bản pháp quy có ID link trong file “linkdata.csv” được tải về dưới dạng file .htm trong folder *“download”*.

**Sau khi đã tải xuống thành công tất cả các dữ liệu văn bản pháp quy, chúng ta tiếp tục thực hiện bước phân tích dữ liệu.

Hình 12: Dữ liệu được tải về thành công được lưu trong folder “download”

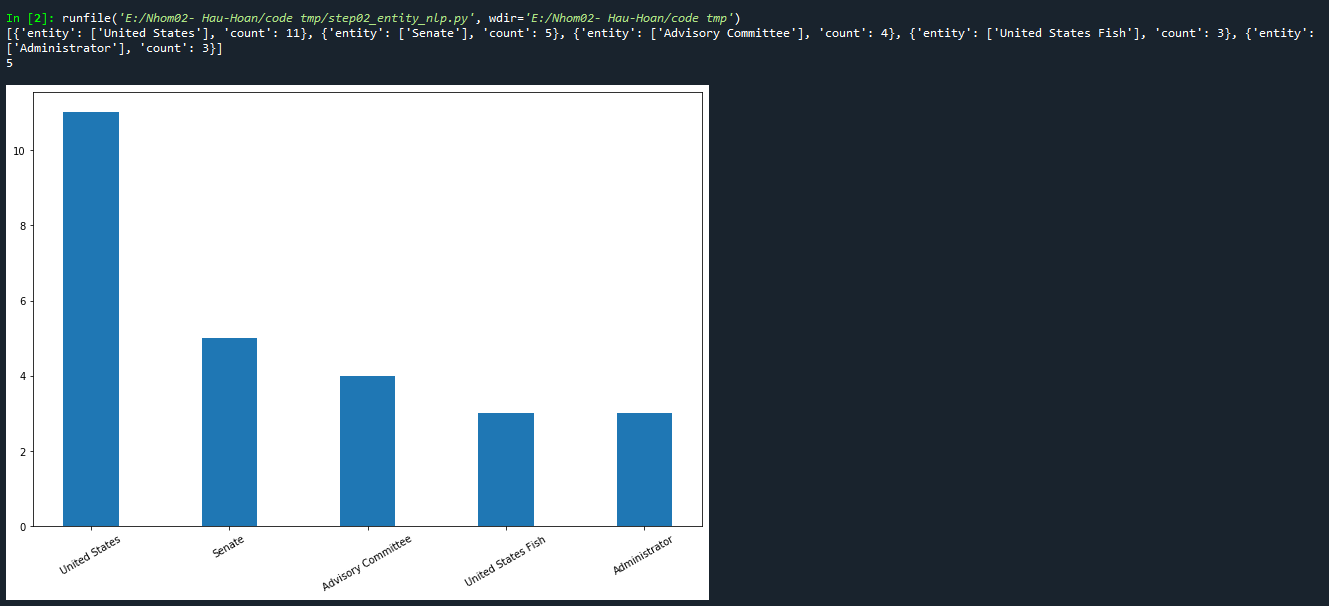
## **3.3. Phân tích dữ liệu**

Mở file **“***step02\_entity\_nlp.py”***:**

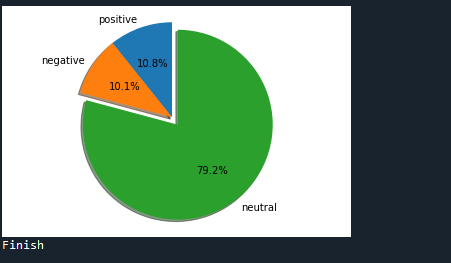


Hình 13: Giao diện khi mở file "step02\_entity\_nlp.py" bằng Spyder (Python 3.7)

Tiến hành thực thi chương trình trong file *“step02\_entity\_nlp.py”*:

Tại đây, dữ liệu trong các văn bản được thu thập và chương trình bắt đầu thống kê tần suất xuất hiện của các từ, ở đây vì lượng dữ liệu khá lớn nên dữ liệu sẽ được xử lý và thống kê theo từng tháng trong năm:

Hình 14: Biểu đồ cột thống kê các từ có số lần xuất hiện nhiều nhất trong văn bản pháp quy tháng 1 – năm 2019



Hình 15: Biểu đồ tròn phân tích nội dung của văn bản pháp quy tháng 1 - năm 2019

**A picture containing drawing

Description automatically generated**A screenshot of a cell phone

Description automatically generated

Hình 16: Biểu đồ cột thống kê các từ có số lần xuất hiện nhiều nhất trong văn bản pháp quy tháng 2 – năm 2019

Hình 17: Biểu đồ cột phân tích nội dung trong các văn bản pháp quy tháng 4 - 2019

# **CHƯƠNG 4: KẾT LUẬN**

## **4.1. Kết luận**

Nghiên cứu đã trình bày về vấn đề nhận dạng thực thể có tên trong văn bản pháp quy, một bài toán quan trọng trong lĩnh vực xử lý ngôn ngữ tự nhiên. Luận án tập trung nghiên cứu, phát triển về lý thuyết và ứng dụng đối với bài toán nhận dạng thực thể, đề xuất một số mô hình và giải pháp nhằm nâng cao hiệu quả nhận dạng thực thể trong văn bản pháp quy và đưa ra một số khung làm việc phục vụ cho quá trình nhận dạng thực thể.

Nâng cao chất lượng nhận dạng thực thể liên quan tới các văn bản pháp quy của chính phủ bằng cách thống kê số lượng từ, xây dựng thành các biểu đồ trực quan, phục vụ cho việc phân tích và tìm kiếm các nội dung liên quan thuận tiện hơn.

## **4.2. Hướng phát triển**

Hiện tại, do hạn chế về thời gian, nghiên cứu dừng lại thử nghiệm với tập dữ liệu có kích cỡ vừa. Sau này chúng em sẽ mở rộng thử nghiệm trên tập dữ liệu lớn hơn.

Mở rộng và thử nghiệm trên các kiểu thực thể và mối quan hệ thực thể khác.

Cải tiến áp dụng các dạng biểu đồ trực quan khác nhau trong việc phân tích các thực thể.

Tuy nghiên cứu còn những hạn chế và thiếu sót nhất định, nhưng chúng em sẽ tiếp tục cố gắng hoàn thiện trong thời gian tới.

# **TÀI LIỆU THAM KHẢO**

[1] Indurkhya, N., & Damerau, F. J. (2010). *Handbook of natural language processing*. Chapman and Hall/CRC.

[2] Keselj, V. (2009). Speech and Language Processing Daniel Jurafsky and James H. Martin (Stanford University and University of Colorado at Boulder) Pearson Prentice Hall, 2009, xxxi+ 988 pp; hardbound, ISBN 978-0-13-187321-6, $115.00.

[3] Mitkov, R. (Ed.). (2004). *The Oxford handbook of computational linguistics*. Oxford University Press.

[4] Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

[5] Manning, C. D., Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT press.

[6] Daniel Waegel. The Development of Text-Mining Tools and Algorithms.Ursinus College, 2006.

[7] Trần, M. V. (2018). *Nghiên cứu nhận dạng thực thể có tên và thực thể biểu hiện trong văn bản và ứng dụng* (Doctoral dissertation).

[8] Chanrathany, S. (2012). *Trích rút thực thể có tên và quan hệ thực thể trong văn bản tiếng Việt* (Doctoral dissertation, Trường Đại học Bách Khoa Hà Nội).

[9] www.nltk.org

[10] pandas.pydata.org

[11] www.govinfo.gov