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

[**LỜI CẢM ƠN** 1](#_Toc67772351)

[**CHƯƠNG 1. TỔNG QUAN** 2](#_Toc67772352)

[**1.1. Giới thiệu bài toán** 2](#_Toc67772353)

[**1.2. Xử lý hình ảnh** 3](#_Toc67772354)

[**1.3. Nhận dạng đối tượng đeo khẩu trang.** 4](#_Toc67772355)

[**1.4. Phạm vi đề tài** 5](#_Toc67772356)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT** 6](#_Toc67772357)

[**2.1. Nhận diện khuôn mặt** 6](#_Toc67772358)

[**2.2. Nhận diện khẩu trang trên khuôn mặt** 7](#_Toc67772359)

[**2.3. Tình hình nghiên cứu ngoài nước về bài toán Face Detection** 8](#_Toc67772360)

[**CHƯƠNG 3: THỰC NGHIỆM CHƯƠNG TRÌNH** 10](#_Toc67772361)

[**3.1. Môi trường** 10](#_Toc67772362)

[3.1.1 Ngôn ngữ Python 10](#_Toc67772363)

[3.1.2 Thư viện Tensorflow 11](#_Toc67772364)

[3.1.3 Thư viện Numpy 11](#_Toc67772365)

[3.1.4 Thư viện Matplotlib 12](#_Toc67772366)

[3.1.5 Thư viện Keras 12](#_Toc67772367)

[**3.2. Dữ liệu** 13](#_Toc67772368)

[3.2.1 Yêu cầu dữ liệu 13](#_Toc67772369)

[3.2.2 Tiền xử lý dữ liệu 16](#_Toc67772370)

[**3.3. Phân tích dữ liệu** 19](#_Toc67772371)

[**CHƯƠNG 4: KẾT LUẬN** 22](#_Toc67772372)

[**4.1. Kết luận** 22](#_Toc67772373)

[**4.2. Hướng phát triển** 22](#_Toc67772374)

[**TÀI LIỆU THAM KHẢO** 23](#_Toc67772375)

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

Hình 2: Ví dụ về cây cú pháp 9

Hình 3: Minh họa về khái niệm trích xuất thông tin (Information Extraction) 11

Hình 4: Minh họa về hoạt động của NER (Named Entity Recoginition) 12

Hình 5: IEEE Spectrum The Top Programming languages 2018 14

Hình 6: Trình tải xuống NLTK để chọn gói nào cần được cài đặt 16

Hình 7: Giao diện trang thu thập các văn bản pháp quy 19

Hình 8: Ví dụ về định dạng của một văn bản pháp quy (.htm) 20

Hình 9: : Thêm cấu trúc link download vào Regex trong Settings   
của Link Klipper 21

Hình 10: Giao diện khi mở file step01\_download\_textfile.py bằng Spyder   
(Python 3.7) 22

Hình 11: File "step01\_download\_textfile.py" đang được thực thi 23

Hình 12: Dữ liệu được tải về thành công được lưu trong folder “download” 24

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

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

Hình 15: Biểu đồ tròn phân tích nội dung của văn bản pháp quy   
tháng 1 - năm 2019 26

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

Hình 17: Biểu đồ cột phân tích nội dung trong các văn bản pháp quy   
tháng 4 - 2019 27

# **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. /**

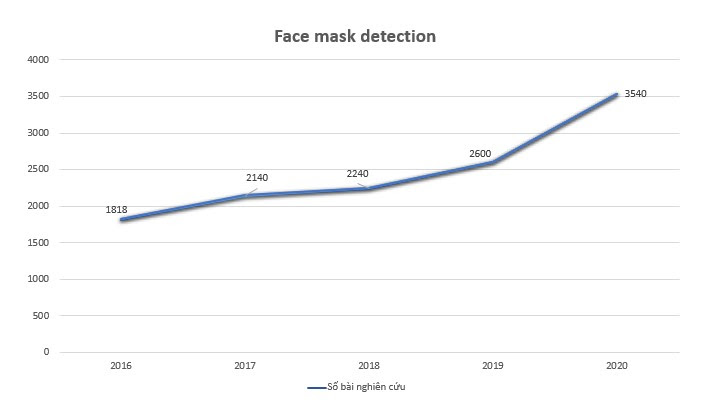
Lê Tấn Lộc

# **CHƯƠNG 1. TỔNG QUAN**

## **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: Springer

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 thần. 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.

## **Xử lý hình ảnh**

Xử lý hình ảnh 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 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.

Nhờ vào sự tiến bộ của công nghệ kỹ thuật số đã góp phần tạo điều kiện cho chúng ta có thể thu thập được nguồn dữ liệu phong phú, song việc xác định, phân tích và nghiên cứu cũng trở nên chuẩn xác hơn.

## **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. Một dữ liệu sau khi được kiểm tra sẽ được xếp vào một trong hai trạng thái là có đeo khẩu trang hoặc 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: C**hưa được phát hiện, thì bài toàn này được sử dụng để hỗ trợ cho các thiết bị ghi hình nhằm phát hiện những hành vi bất thường như trộm cướp, hay có ý định thực hiện những hành vị sai trái và phạm pháp 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 cần thiết và 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.

## **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.

Phạm vi áp dụng có thể áp dụng trong các siêu thị, cửa hàng tiện lợi, tiệm tạp hóa, trường học, trung tâm thương mai…

Đối tượng được xử lý phải đảm bảo có 2/3 khuôn mặt được xác định qua khung hình camera thì mới có thể nhận diện được.

**Hướng phát triển trước, trong và sau dịch:**

**Trước mùa dịch CoV**: Sản phẩm sẽ tập trung vào những người đeo khẩu trang khi họ đi vào những nơi công cộng vì những người này không muốn người khác nhìn thấy mặt mình nên rất có khả năng họ sẽ làm điều gì đó mờ ám.

**Trong dịch mùa dịch CoV:** Để đảm bảo sức khỏe của mọi người và tránh tình trạng lây lan dịch bệnh thì việc đeo khẩu trang đến các nơi công cộng là điều bắt buộc. Vì vậy những ai không đeo khẩu trang khi đến những nơi công cộng là người có những hành vi bất thường.

**Sau mùa dịch CoV:** Có thể đổi lại cài đặt giống như lúc trước dịch. Vì việc đeo khẩu trang lúc này không còn bắt buộc nữa.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## **2.1. Nhận diện khuôn mặt**

Nhận diện khuôn mặt (Face Detection) là một nhánh của Trí tuệ nhân tạo, tập trung vào việc nghiên cứu các bài toán để máy tính có thể tìm và nhận diện khuôn mặt trên những hình ảnh kỹ thuật số, được phát triển dựa trên những nghiên cứu của Thị giác máy tính (Computer Vision) cùng với sự tiến bộ các lĩnh vực như: máy học (Machine Learning), Mạng nơ-ron nhân tạo (Neural Networks), và nhiều công nghệ khác. Mục tiêu của lĩnh vực này là giúp máy tính tìm và nhận diện có hiệu quả với những đối tượng là khuôn mặt, ngoài việc xử lý để nhận diện khuôn mặt thì lĩnh vực này còn tập trung việc xác định các thông tin thông qua khuôn mặt như: tuổi, giới tính và cả cảm xúc của con người.

//lịch sử

**Một số bài toán nhận diện khuôn mặt kinh điển:**

**//**

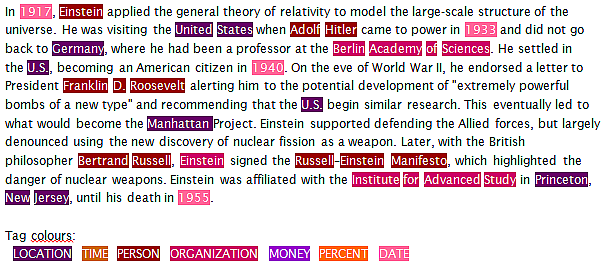
**Các thuật ngữ cơ bản trong nhận diện khuôn mặt:**

## **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”*

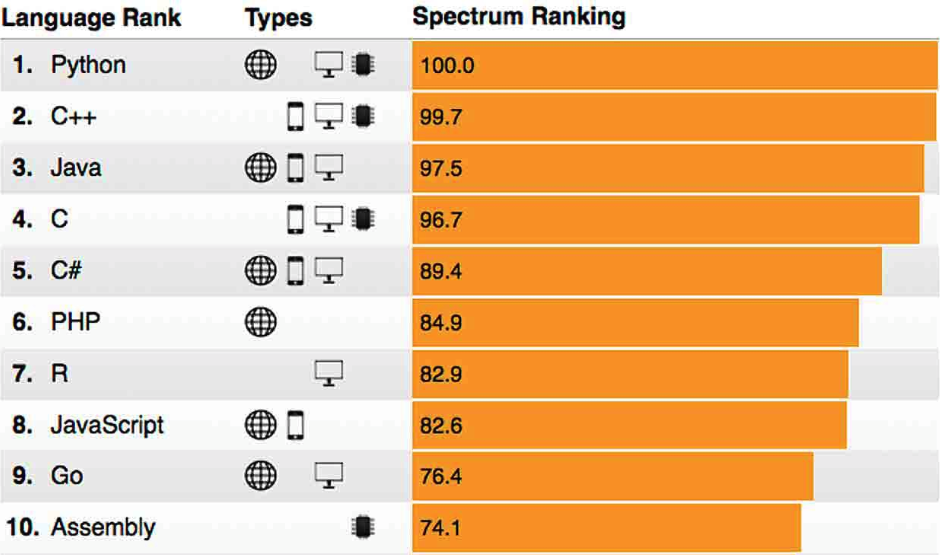
- Giả định ngữ cảnh: Các thực thể như con người, tổ chức, địa điểm thường có các ngữ cảnh xung quanh giúp chúng ta nhận biết được chúng một cách dễ dàng. Chẳng hạn, tên một tổ chức thường được đi cùng bởi các hậu tố như: *Inc., Co.,* …; tên một người thường được 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 hình ảnh. Chính vì vậy, Python là một lựa chọn hợp lý khi thực hiện xử lý hình ảnh và nhận diện khuôn mặt.

Hình 5: IEEE Spectrum The Top Programming languages 2018

### 3.1.2 Thư viện Tensorflow

Tensorflow là một thư viện mã nguồn mở để tính toán và học máy quy mô lớn được sử dụng chủ yếu trong việc đào tạo máy học và suy luận trong các mạng nơ-ron như: thu thập dữ liệu, xây dựng mô hình đào tạo, phục vụ đề án và dự đoán kết quả tương lai.

Tesorflow kết hợp các mô hình và thuật toán Machine Learning và Deep Learning lại với nhau và chạy trên Python.

Cho phép tạo ra một đồ thị tính toán để thực hiện. Mỗi một đỉnh trong biểu đồ đại diện cho một phép toán và mỗi kết nối đại diện cho dữ liệu, giúp có cái nhìn trực quan hơn về lối logic của bài toán.

**-Cài đặt:**

Để cài đặt Tensorflow trên môi trường Python, có thể cài đặt trực tiếp thông qua pip.

pip install tensorflow

### 3.1.3 Thư viện Numpy

Numpy là một package tuyệt vời dành cho tính toán dữ liệu trong Python. Nó là 1 thư viện Python có thể cung cấp 1 mảng đa chiều như mảng đánh dấu hoặc ma trận và cung cấp những phép toán xoay quanh ma trận bao gồm toán logic, định dạng size, sắp xếp, chọn lọc, biến đổi, biểu diển những phép tính cơ bản của tuyến tính, mô phỏng random.

Numpy cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh

-**Cài đặt**

Để cài đặt Numpy trên môi trường Python, có thể cài đặt trực tiếp thông qua pip.

pip install numpy

Nếu sử dụng conda:

Sử dụng environment thay vì cài đặt trong base env

conda create -n my-env

conda activate my-env

Muốn cài đặt từ conda-forge

conda config --env --add channels conda-forge

Sau đó nhập lệnh cài đặt trên terminal

conda install numpy

### 3.1.4 Thư viện Matplotlib

Matplotlib là một thư viện viết bằng Python dùng để hổ trợ cho việc vẽ đồ thị trong ngôn ngữ lập trình Python. Nó biểu diển được cả đồ thị 3d và 2d

Hổ trợ rất mạnh mẽ hữu ích trong việc vẽ đồ thị cho những người làm việc với Python và Numpy

**-Cài đặt**

Để cài đặt Numpy trên môi trường Python

python -m pip install -U pip

python -m pip install -U matplotlib

### 3.1.5 Thư viện Keras

Keras là một thư viện mã nguồn mở được viết bằng Python nó cung cấp giao diện cho mạng neural network. Keras hoạt động như một giao diện cho thư viện TensorFlow.

Nó giúp cho chúng ta xây dựng dễ dàng các models như deep learning mà không cần có kiến thức sâu về các thuật toán máy học cũng như về mạng neural network

**-Cài đặt**

Bưới 1: Cài thư viện tensorflow

pip install tensorflow

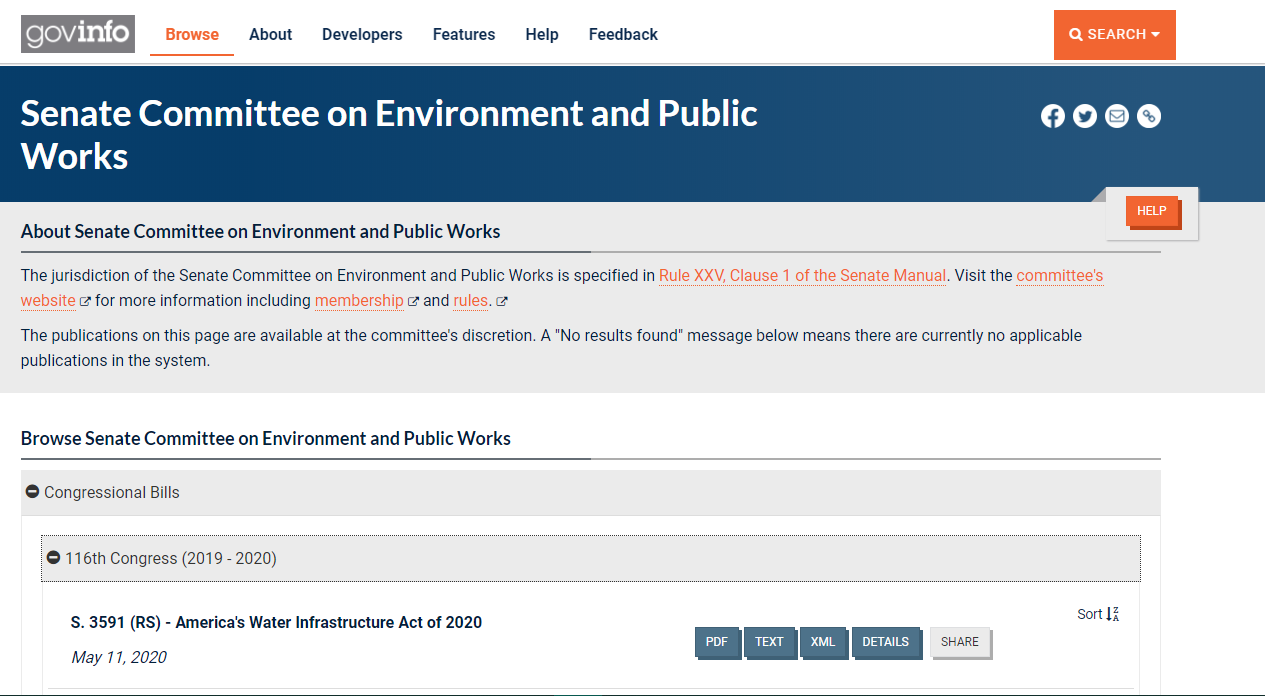
Bưới 2: import thư viện tensorflow trong code

import tensorflow as tf

from tensorflow import keras

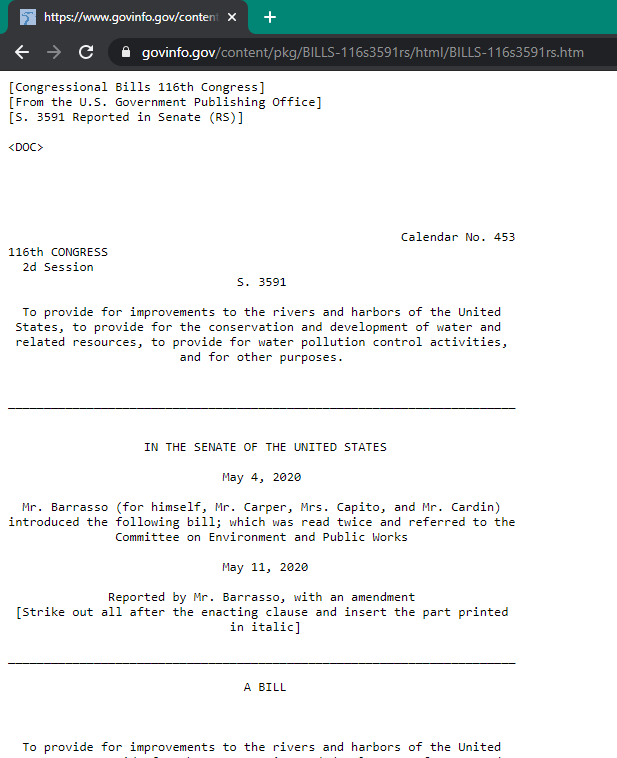
## **3.2. Dữ liệu**

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”* năm 2019 – 2020.

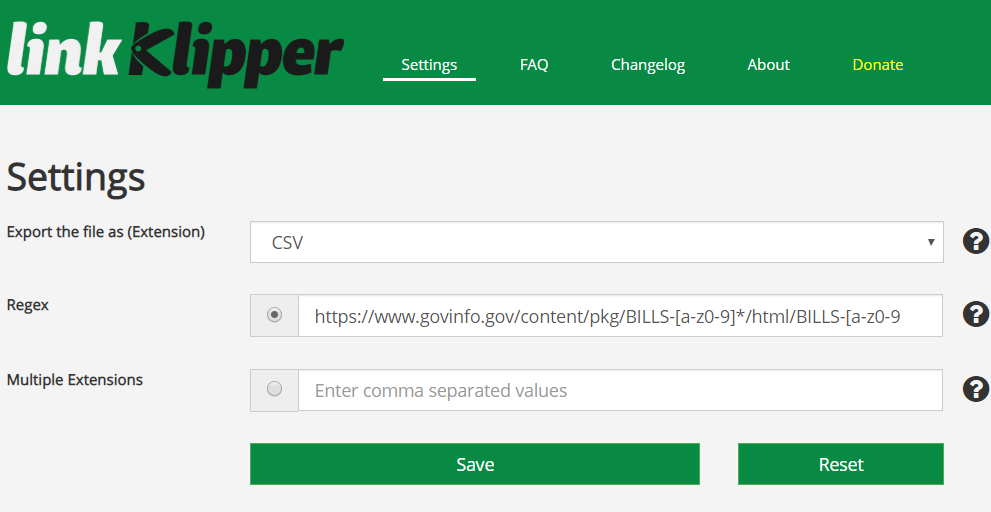
Hình 7: Giao diện trang thu thập các văn bản pháp quy

### 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)

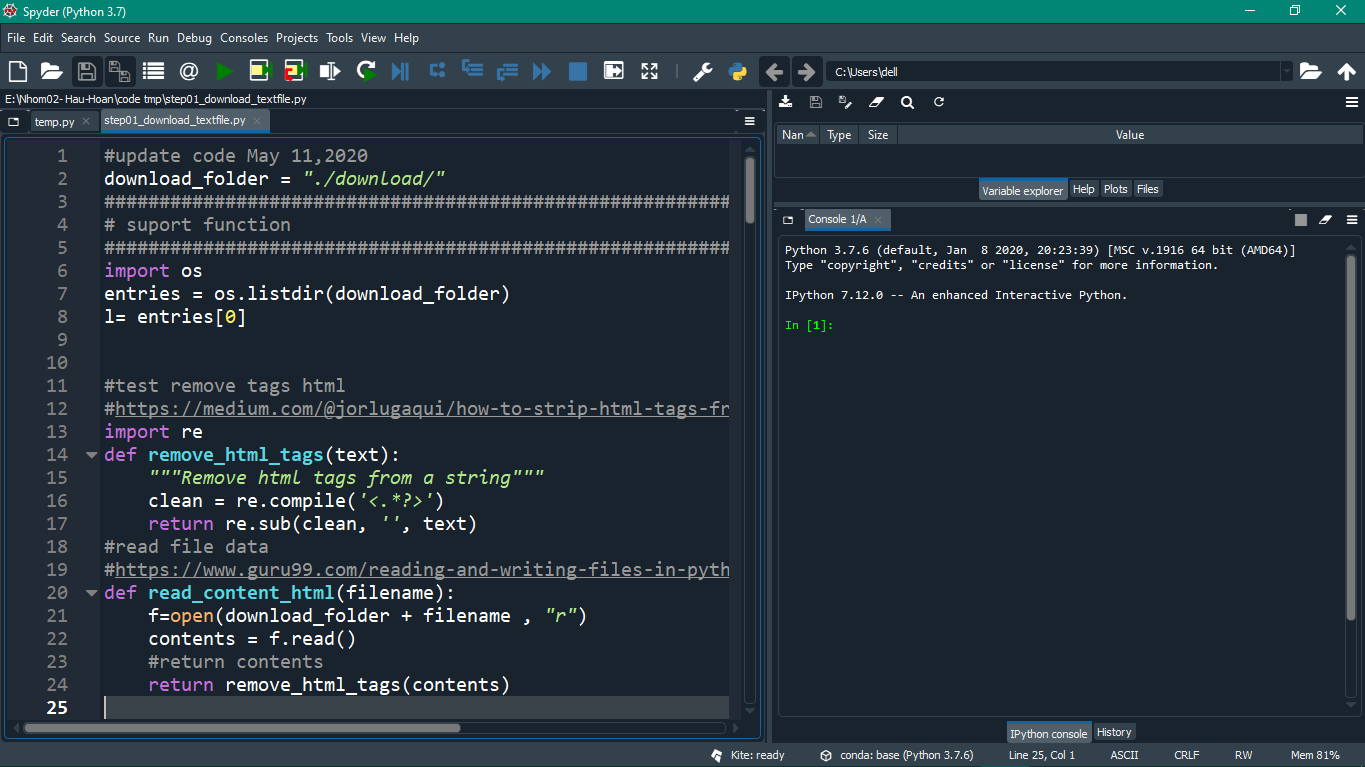
Quá trình tải xuống dữ liệu được thực hiện như sau:

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

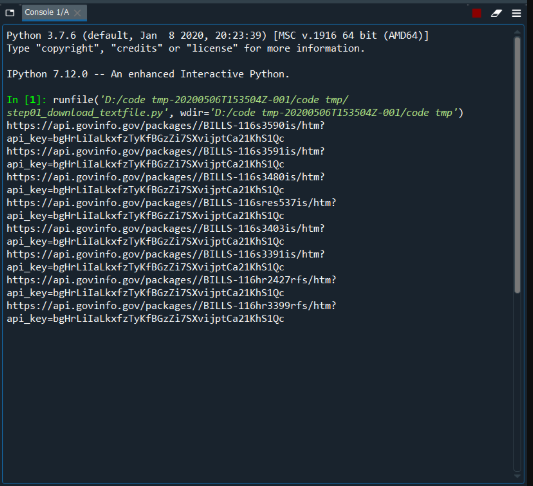
Trong Cài đặt của “Link Klipper”, thêm cấu trúc link *https://www.govinfo.gov/content/pkg/BILLS-[a-z0-9]\*/html/BILLS-[a-z0-9]\** vào mục Regex để lấy được tất cả ID link của các văn bản pháp quy, các ID link này sẽ được lưu vào file “linkdata.csv”.

### 3.2.2 Tiền xử lý dữ liệu

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

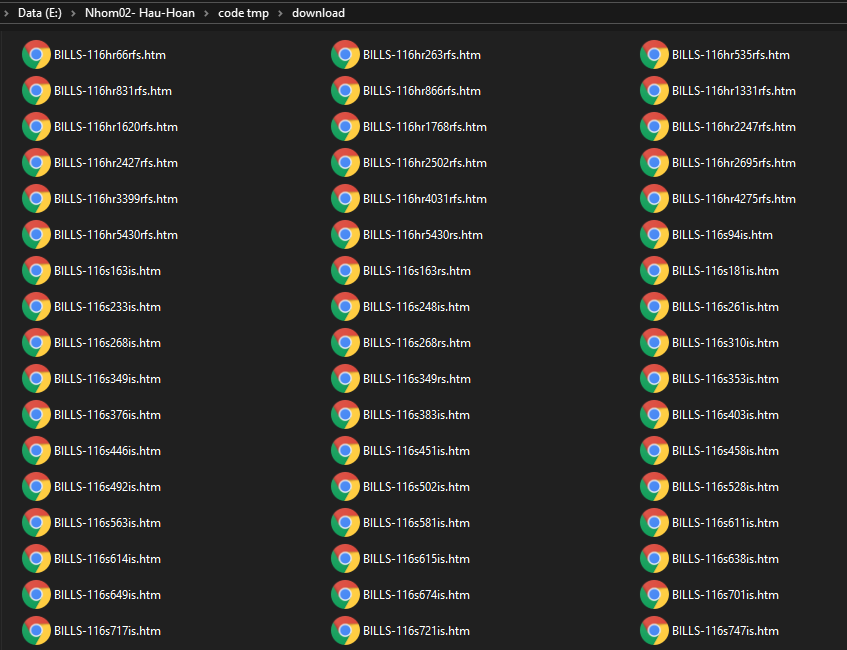
Hình 10: Giao diện khi mở file step01\_download\_textfile.py bằng Spyder (Python 3.7)

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



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

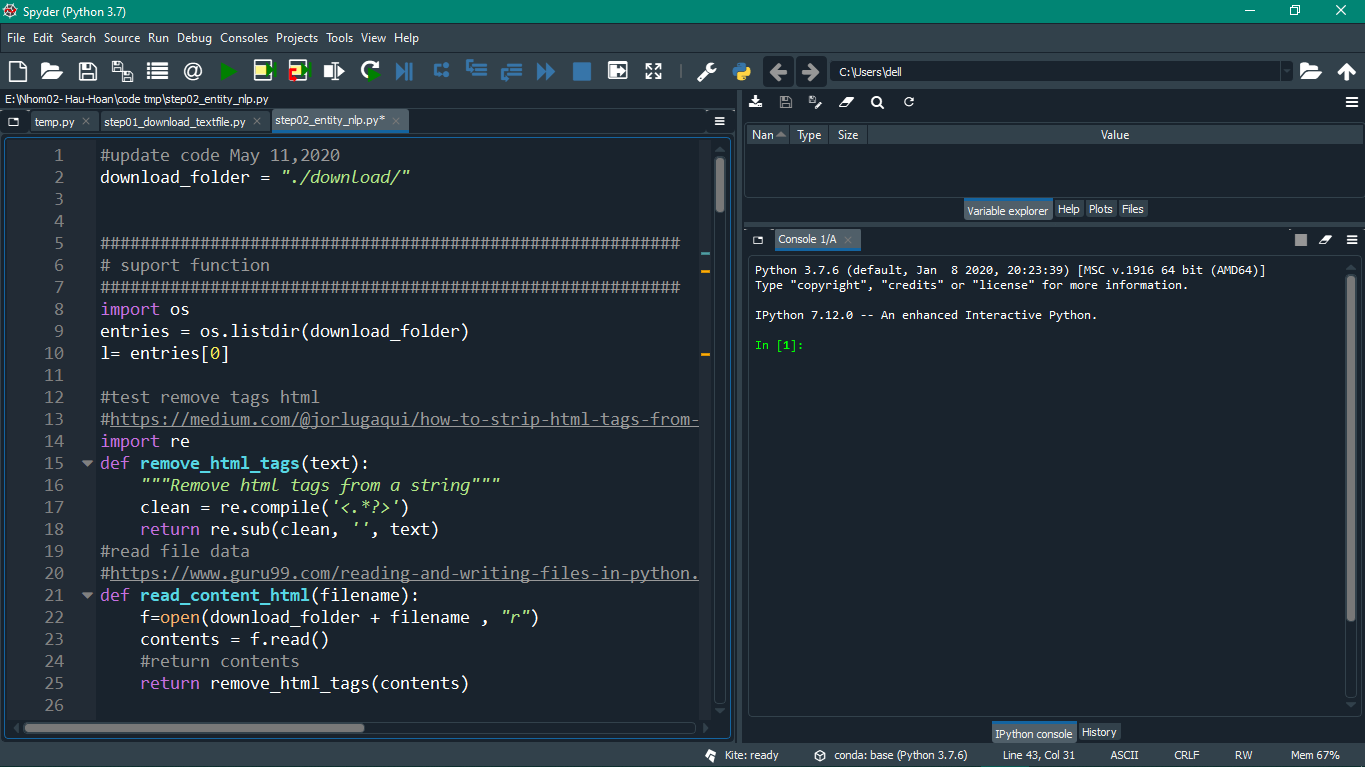
Sau khi chương trình thực thi xong, tất cả các văn bản pháp quy có ID link trong file “linkdata.csv” được tải về dưới dạng file .htm trong folder *“download”*.

**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”

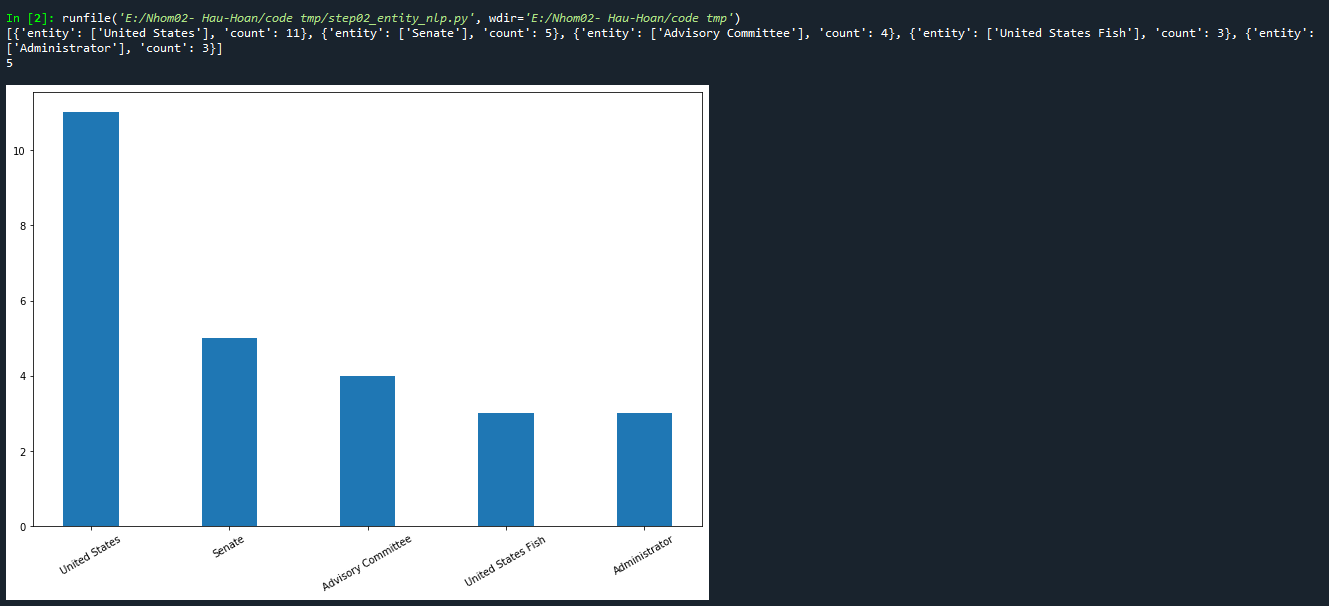
## **3.3. Phân tích dữ liệu**

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

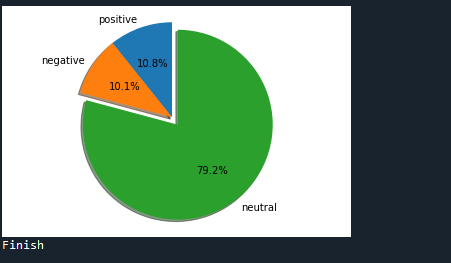


Hình 13: Giao diện khi mở file "step02\_entity\_nlp.py" bằng Spyder (Python 3.7)

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