

ĐẠI HỌC THỦ DẦU MỘT



**BÀI TẬP LỚN MÔN
THU THẬP VÀ TIỀN XỬ LÝ DỮ LIỆU**

**THU THẬP VÀ TIỀN XỬ LÝ DỮ LIỆU TUYỂN DỤNG LAO ĐỘNG
TẠI VIỆT NAM**

Sinh viên thực hiện: **HOÀNG KIM TUYẾN**

Mã số sinh viên: **1824801040043**

Lớp: **D18HT01**

Bình Dương, tháng 11 năm 2020

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của ThS. Hồ Ngọc Trung Kiên. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây.

Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong báo cáo còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung báo cáo của mình. Trường Đại học Thủ Dầu Một không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

Bình Dương, ngày 11 tháng 7 năm 2020

Người thực hiện

(ký tên và ghi rõ họ tên)

MỤC LỤC

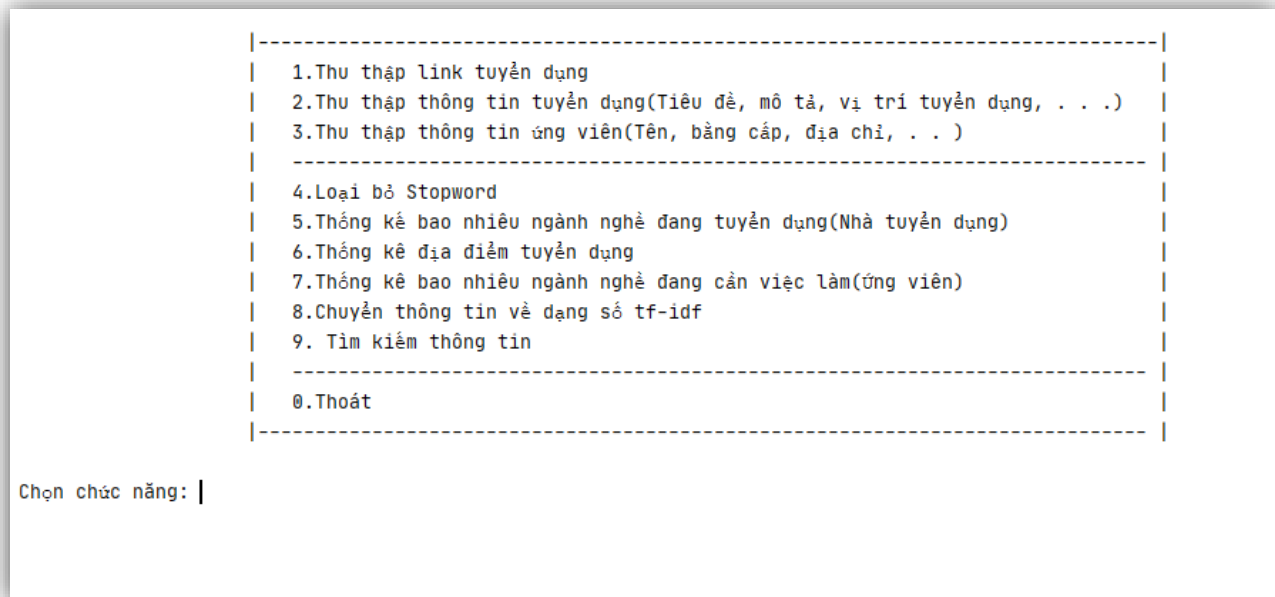
PHẦN 1	4
1.1. Mục đích đề tài.....	4
1.2. Câu hỏi nghiên cứu.....	5
1.4. Phân tích dữ liệu.....	6
1.5. Dự đoán.....	11
❖ Xử lí <i>StopWord</i>	11
❖ TF-IDF	13
❖ Tìm kiếm.....	15
PHẦN 2: TỰ CHẤM.....	17
DANH MỤC TÀI LIỆU THAM KHẢO	18

PHẦN 1

1.1. Mục đích đề tài

Thu thập thông tin tuyển dụng trên website tìm việc nhằm thống kê nhu cầu tuyển dụng lao động tại Việt Nam

Từ các thông tin, dữ liệu thu thập được, tiến hành xử lí, phân tích. Từ đó đưa ra các dự đoán về nhu cầu tuyển dụng và xu hướng tìm việc trong tuyển dụng lao động tại Việt Nam.



Giao diện các chứng năng của đề tài(Pycharm)

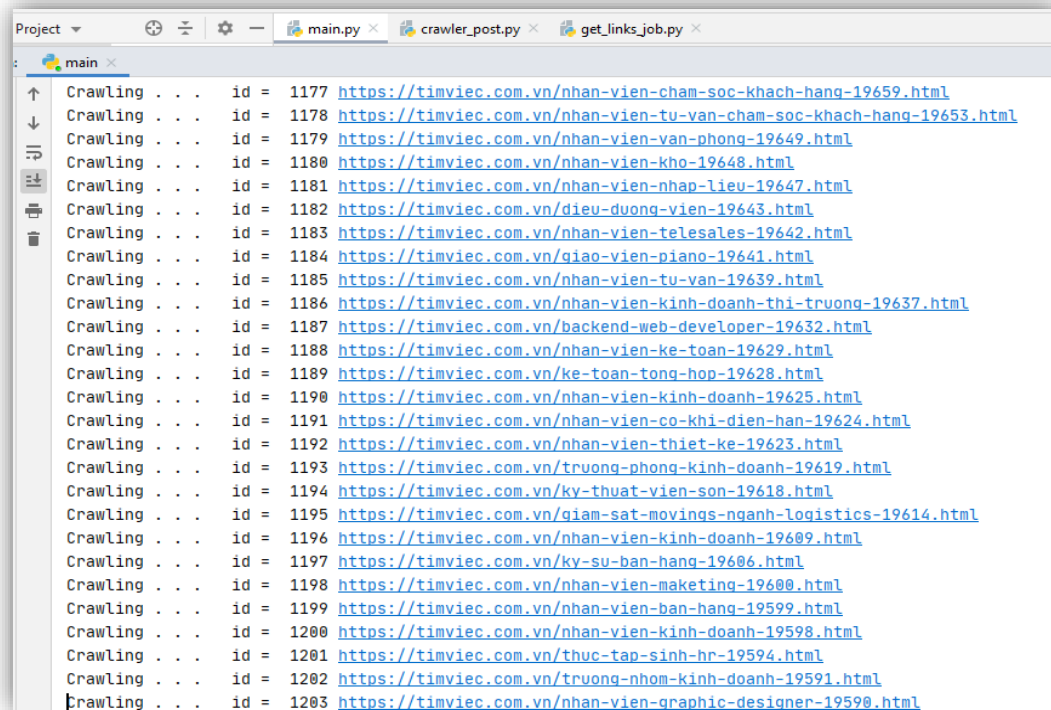
1.2. Câu hỏi nghiên cứu

- Sử dụng thư viện trong Python để thu thập và lưu trữ 3000 tin tuyển dụng và 3000 ứng viên.
- Thu thập dữ liệu tại trang website tuyển dụng: **timviec.com.vn**
- Các trường dữ liệu thu thập *thông tin tuyển dụng* gồm:
 - Tiêu đề, mô tả
 - Link bài viết
 - Tên công ty, địa điểm tuyển dụng
 - Ngành nghề tuyển dụng
 - Mức lương
- Các trường dữ liệu thu thập *thông tin ứng viên* gồm:
 - Họ và tên ứng viên
 - Tên công việc ứng tuyển
 - Địa điểm ứng tuyển

- Số năm kinh nghiệm

1.4. Phân tích dữ liệu

Sử dụng các thư viện **BeautifulSoup**, **requests**, **newspaper**, **sqlite3** để thu thập thông tin tuyển dụng, các thư viện **nltk** để xử lý ngôn ngữ tự nhiên.



The screenshot shows a Python IDE with three open files: `main.py`, `crawler_post.py`, and `get_links_job.py`. The `main.py` file is active and displays a list of job postings. Each entry consists of the word "Crawling" followed by an ID and a URL from `timviec.com.vn`. The URLs represent various job categories such as "cham-soc-khach-hang", "tu-van", "van-phong", "kho", "nhap-lieu", "dieu-duong-vien", "telesales", "giao-vien-piano", "kinh-doanh-thi-truong", "backend-web-developer", "ke-toan", "ke-toan-tong-hop", "kinh-doanh", "co-khi-dien-han", "thiet-ke", "truong-phong-kinh-doanh", "ky-thuat-vien-son", "giam-sat-movings-nganh-logistics", "kinh-doanh", "marketing", "ban-hang", "thuc-tap-sinh-hr", and "graphic-designer".

ID	URL
1177	https://timviec.com.vn/nhan-vien-cham-soc-khach-hang-19659.html
1178	https://timviec.com.vn/nhan-vien-tu-van-cham-soc-khach-hang-19653.html
1179	https://timviec.com.vn/nhan-vien-van-phong-19649.html
1180	https://timviec.com.vn/nhan-vien-kho-19648.html
1181	https://timviec.com.vn/nhan-vien-nhap-lieu-19647.html
1182	https://timviec.com.vn/dieu-duong-vien-19643.html
1183	https://timviec.com.vn/nhan-vien-telesales-19642.html
1184	https://timviec.com.vn/giao-vien-piano-19641.html
1185	https://timviec.com.vn/nhan-vien-tu-van-19639.html
1186	https://timviec.com.vn/nhan-vien-kinh-doanh-thi-truong-19637.html
1187	https://timviec.com.vn/backend-web-developer-19632.html
1188	https://timviec.com.vn/nhan-vien-ke-toan-19629.html
1189	https://timviec.com.vn/ke-toan-tong-hop-19628.html
1190	https://timviec.com.vn/nhan-vien-kinh-doanh-19625.html
1191	https://timviec.com.vn/nhan-vien-co-khi-dien-han-19624.html
1192	https://timviec.com.vn/nhan-vien-thiet-ke-19623.html
1193	https://timviec.com.vn/truong-phong-kinh-doanh-19619.html
1194	https://timviec.com.vn/ky-thuat-vien-son-19618.html
1195	https://timviec.com.vn/giam-sat-movings-nganh-logistics-19614.html
1196	https://timviec.com.vn/nhan-vien-kinh-doanh-19609.html
1197	https://timviec.com.vn/ky-su-ban-hang-19606.html
1198	https://timviec.com.vn/nhan-vien-marketing-19600.html
1199	https://timviec.com.vn/nhan-vien-ban-hang-19599.html
1200	https://timviec.com.vn/nhan-vien-kinh-doanh-19598.html
1201	https://timviec.com.vn/thuc-tap-sinh-hr-19594.html
1202	https://timviec.com.vn/truong-nhom-kinh-doanh-19591.html
1203	https://timviec.com.vn/nhan-vien-graphic-designer-19590.html

Tiến hành thu thập dữ liệu từ website tuyển dụng(timviec.com.vn)

- Sử dụng các hàm trong thư viện **BeautifulSoup**, **newspaper** để tìm ra các khối dữ liệu sau đó tiến hành xử lý, loại bỏ các khối *html*, *dấu cách*, . . . và lưu vào **database(sqlite3)**

```
try:
    response = requests.get(url)
    soup = BeautifulSoup(response.content, "html.parser")

    # get job about
    job_about = soup.find('ul', class_="list-unstyled fs-14 mb-0")

    # get job name
    job_name = job_about.find('h1', class_='fs-20 m-0').text

    # get company name
    company_name = job_about.find('a', class_="color-main fs-16").text.strip()

    #get location -> find all element li -> get li[2]
    all_li_element = job_about.findAll('li')
    location = all_li_element[2].text.replace("Địa điểm tuyển dụng: ", "").strip()

    # get salary
    salary = job_about.find('span', class_="color-ed145b").text
```

Sử dụng thư viện để thu thập thông tin từ website tuyển dụng

```
article = Article(url)
article.download()
article.parse()

query = """
INSERT INTO JOBS_DATA (
    TITLE,
    DESCRIPTION,
    LINK,
    JOB_NAME,
    LOCATION,
    COMPANY_NAME,
    SALARY,
    CONTENT,
    OCCUPATIONS,
)
VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
"""

conn.execute(query, (article.title, article.meta_description, article.url, job_name, location,
                    company_name, salary, article.text, occupation)) # thêm dữ liệu crawl vào db
conn.commit()
```

Lưu trữ thông tin thu thập được vào cơ sở dữ liệu

- Sau khi thu thập dữ liệu, lưu trữ dữ liệu, tiến hành load dữ liệu lên và xử lý các yêu cầu như phân tích, thống kê,

```
21 # -----Load dữ liệu tuyển dụng----- #
22 for row in data:
23     tieude.append(row[1])
24     mota.append(row[2])
25     link.append([row[3]])
26     ten_cv.append(row[4])
27     ngành_nghe.append(row[5])
28     dia_diem.append(row[6])
29 # -----load dữ liệu ứng viên----- #
30 data_ungvien = conn.execute("SELECT * FROM EMPLOYER")
31 ten_ung_vien = []
32 ngành_nghe_ung_tuyen = []
33 dia_diem_ung_tuyen = []
34 kinh_nghiem = []
35 for row in data_ungvien:
36     ten_ung_vien.append(row[1])
37     ngành_nghe_ung_tuyen.append(row[2])
38     dia_diem_ung_tuyen.append(row[3])
39     kinh_nghiem.append(row[4])
```

Tiến hành load dữ liệu

- Sau khi có dữ liệu, tiến hành phân tích và vẽ biểu đồ

```
15 def thong_ke_nganh_nghe(nganh_nghe):
16     # xử lý, tách ngành nghề bỏ vào list, set
17     list_nganh_nghe = []
18     for nghe in nganh_nghe:
19         # bỏ dấu '/'
20         nghe = str(nghe).replace('/', "") # -> nghe = "Hành chính Thu ký Hành chính văn phòng Bất động sản"
21         a = re.split(r"(?=[A-Z])", nghe) # a = ['Trợ lý', 'Hành chính văn phòng', 'Bất động sản'] dùng Regex
22         for i in a:
23             list_nganh_nghe.append(i.replace('nhân viên kinh doanh', "Kinh doanh").replace("nhân viên tư vấn", "Tư
24     |
25     set_list_nganh_nghe = set(list_nganh_nghe) # loại bỏ các nghề trùng nhau
26     for nghe in set_list_nganh_nghe:
27         print(nghe)
28         print("-----")
29         print('Thống kê số lượng bài tuyển dụng theo ngành nghề: ( hơn 100 bài)')
30         for i in set_list_nganh_nghe:
31             count = list_nganh_nghe.count(i)
32             if count > 100:
33                 print(str(i).strip(), ":", count)
34         print("-----Thống kê ngành nghề-----")
35         print("Có tổng số ngành nghề: ", len(set_list_nganh_nghe))
36
37     fdist_nganh_nghe = FreqDist(list_nganh_nghe) # tuần suất xuất hiện
38     fdist_nganh_nghe.plot(10) # vẽ biểu đồ
```

Xử lý, thống kê các ngành nghề và vẽ biểu đồ

Thống kê số lượng bài tuyển dụng theo ngành nghề: (hơn 100 bài)

Trợ lý : 135

Hành chính : 163

Kế toán : 172

Chăm sóc khách hàng : 153

Kinh doanh : 428

Tư vấn : 233

Quảng cáo : 129

Tài chính : 172

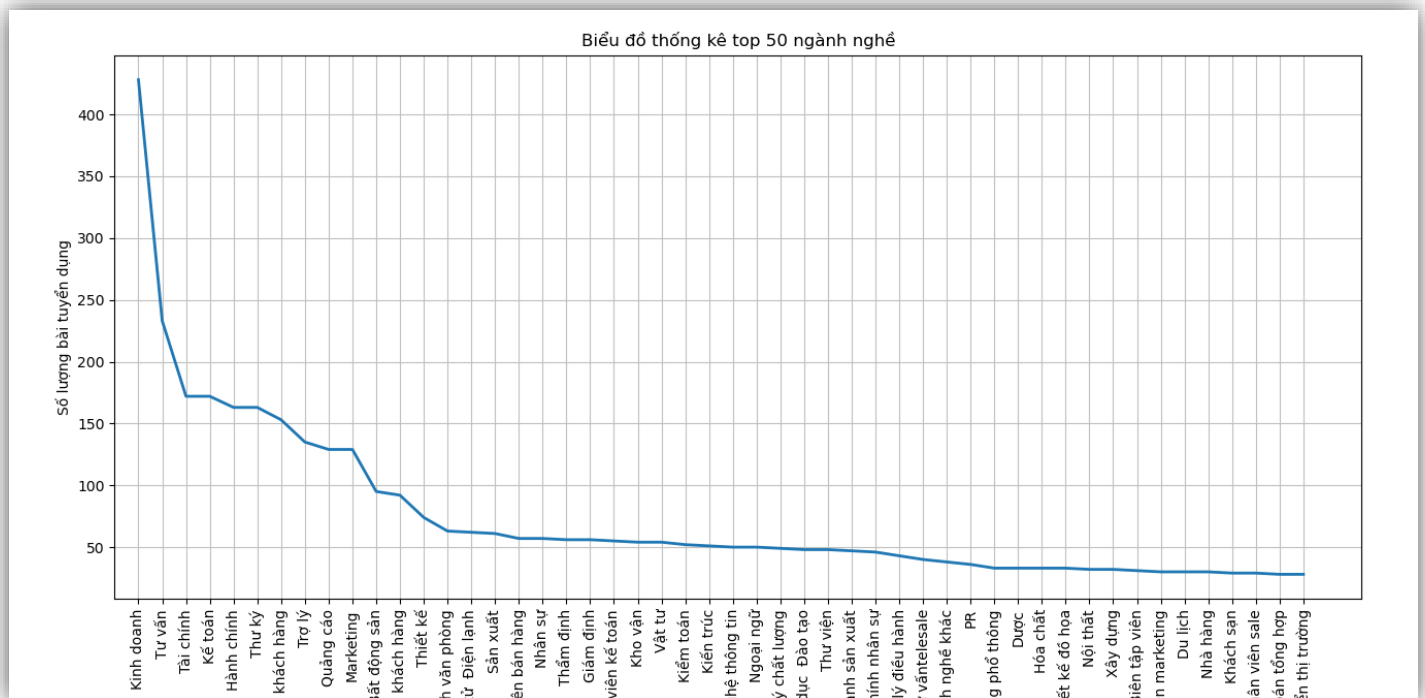
Marketing : 129

Thư ký : 163

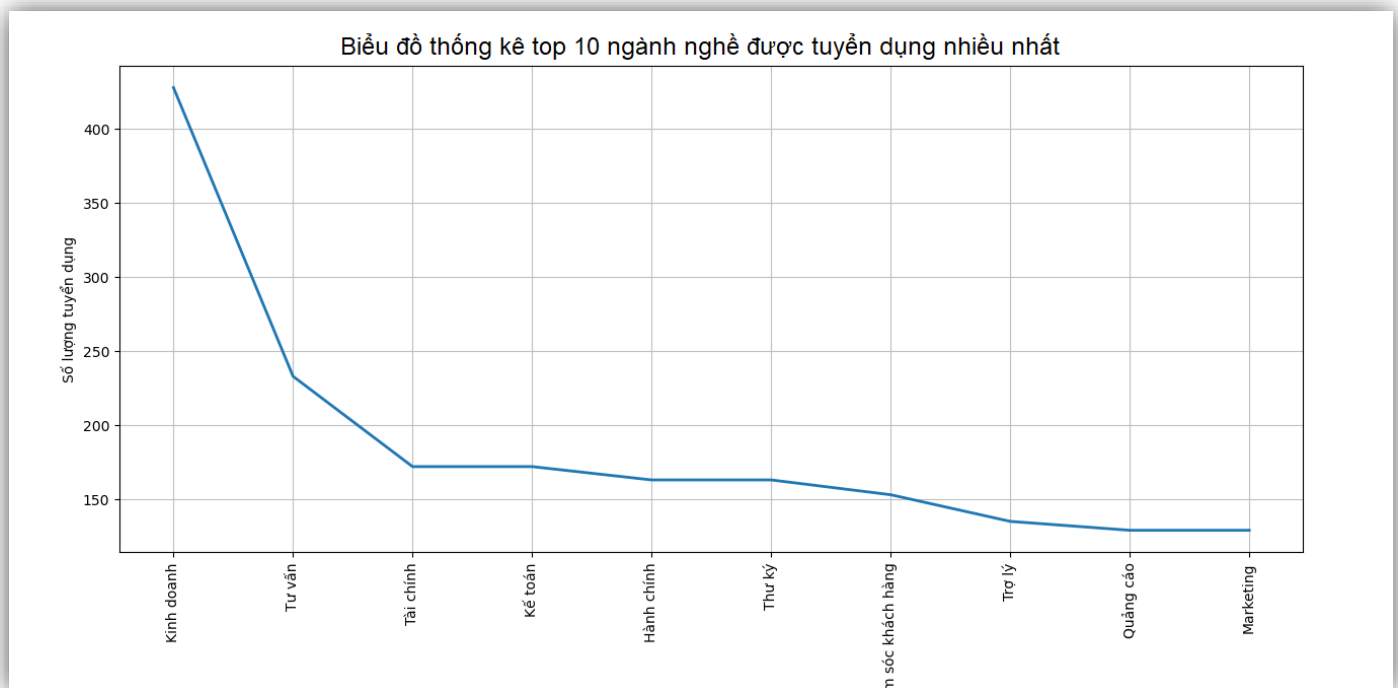
-----Thống kê ngành nghề-----

Có tổng số ngành nghề: 374

Kết quả thống kê ngành nghề tuyển dụng



Biểu đồ thống kê top 50 ngành nghề được tuyển dụng nhiều nhất



Top 10 ngành nghề được tuyển dụng nhiều nhất

► *Kinh doanh* là ngành nghề có nhu cầu tuyển dụng nhiều nhất

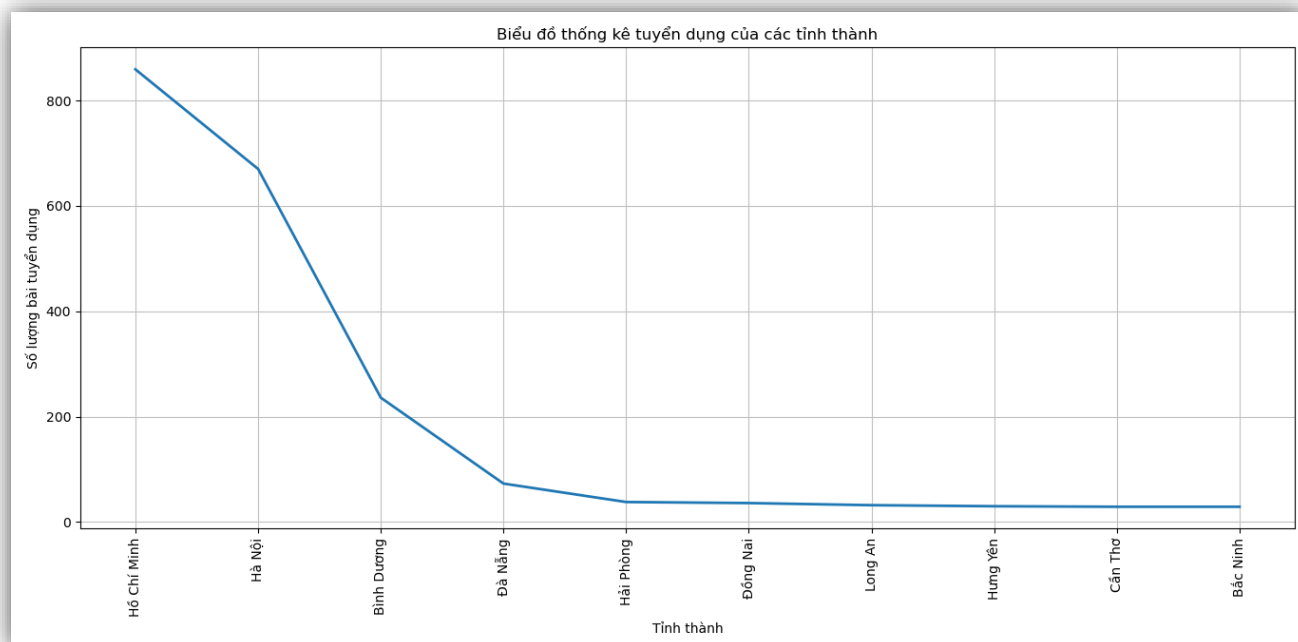
❖ Thống kê nhu cầu tuyển dụng của các tỉnh thành trên cả nước

```

4 def thong_ke_dia_diem_tuyen_dung(dia_diem):
5     list_dia_diem = []
6     for i in dia_diem:
7         locations = str(i).strip().split(',')### "Hồ Chí Minh, Bình Dương" -> "Hồ Chí Minh", "Bình Dương"
8         for location in locations:
9             list_dia_diem.append(location)
10
11     print("Tổng số địa điểm tuyển dụng là: ", len(set(list_dia_diem)))
12     fdist_dia_diem = FreqDist(list_dia_diem) # tuần suất xuất hiện
13     fdist_dia_diem.plot(10) # vẽ biểu đồ top 10 địa điểm
14

```

Thống kê các địa điểm tuyển dụng



Biểu đồ thống kê nhu cầu việc làm của các tỉnh thành

► Tỉnh thành có nhu cầu tuyển dụng cao nhất là *Thành Phố Hồ Chí Minh*, sau đó là *Hà Nội*, *Bình Dương*, *Đà Nẵng*, . . .

1.5. Dự đoán

❖ Xử lí *StopWord*

- StopWords là những từ xuất hiện nhiều trong ngôn ngữ tự nhiên, tuy nhiên lại không mang nhiều ý nghĩa. Ở tiếng việt StopWords là những từ như: *đề*, *này*, *kia*... Tiếng anh là những từ như: *is*, *that*, *this*...

1	bị	11	chứ	20	đã
2	bởi	12	chưa	21	đang
3	cả	13	chuyện	22	đây
4	các	14	có	23	để
5	cái	15	có_thể	24	đến_nổi
6	cần	16	cứ	25	đều
7	càng	17	của	26	điều
8	chỉ	18	cùng	27	do
9	chiếc	19	cũng	28	đó
10	cho	20	đã	29	được
				30	dưới

Danh sách *StopWord* sưu tầm trên wikipedia

- Sau khi thu thập stopwords, tiến hành loại bỏ chúng trong tiêu đề, mô tả của dữ liệu

```
13 stopwords_vietnam = []
14 with open('stopword_vietnam.txt', 'r', encoding="utf8") as f:
15     for line in f:
16         stopwords_vietnam.append(line.strip())
17
18 def loai_bo_stopword(text):
19     text = ' '.join([word for word in text.split() if word not in stopwords_vietnam])
20     return text
21
22 def loai_bo_stopword_trong_danh_sach(danh_sach):
23     danh_sach_ko_co_stopword = [loai_bo_stopword(str(i)) for i in danh_sach]
24     return danh_sach_ko_co_stopword
25
26 if __name__ == "__main__":
27     text = "thì ý là tôi thích học python á hñhj :))"
28     print(loai_bo_stopword(text))
29     ### >>> tôi thích học python
```

Xử lý, loại bỏ StopWord

- Kết quả sau khi loại bỏ stopwords:

```
Chuyên Viên Marketing CÔNG TY TNHH INTELLIPURE VIETNAM
Tổng Đài Viên Dịch Vụ CÔNG TY TNHH AN GIA
Nhân Viên Sale Admin Công ty Cổ phần Tracodi Trading Counselling
Nhân Viên May Da Công Ty Cổ Phần Vcam Việt Nam
Nhân Viên Kế Toán Công Ty CP Xuất Nhập Khẩu Hòa An
Chuyên Viên Môi Giới Đầu Tư Phái Sinh Hàng Hóa Công Ty CP Giao Dịch Hàng Hóa Gia Cát Lợi
Trưởng Phòng Sản Xuất Công ty TNHH Risuntek Việt Nam
Nhân Viên Lễ Tân Công ty CP Grace world
Biên Tập Viên Content - Thu âm Công Ty TNHH Truyền Thông Tầm Nhìn Cộng
Chuyên Viên Chăm Sóc Khách Hàng CÔNG TY TNHH INTELLIPURE VIETNAM
Chuyên Viên Digital Marketing Công Ty TNHH Đầu Tư Thương Mại Và Dịch Vụ Ô Tô Hà Thành
Nhân Viên Kinh Doanh Công Ty TNHH Đầu Tư Thương Mại Và Dịch Vụ Ô Tô Hà Thành
Chuyên Viên Tư Vấn Giáo Dục Englishnow Global
Kỹ Sư XÂY DỰNG HIỆN TRƯỜNG CÔNG TY CP XÂY DỰNG VÀ SẢN XUẤT KANSAI VINA
Quản Lý QA Công ty TNHH Kintex Elastic
Kế Toán Tổng Hợp CÔNG TY CỔ PHẦN ĐẦU TƯ THÁI BÌNH
THỢ HÀN (WELDERS) First Alliances
```

Tiêu đề sau khi loại bỏ stopwords

- ❖ Xử lý ngôn ngữ tự nhiên (natural language processing - NLP) là một nhánh của trí tuệ nhân tạo tập trung vào các ứng dụng trên ngôn ngữ của con người. Trong trí tuệ nhân tạo thì xử lý 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.

- ❖ Bag of Words (Bow)

- Bag of Words là một thuật toán hỗ trợ xử lý ngôn ngữ tự nhiên và mục đích của BoW là phân loại text hay văn bản. Ý tưởng của BoW là phân tích và phân nhóm dựa theo “Bag of Words”(corpus). Với test data mới, tiến hành tìm ra số lần từng từ của test data xuất hiện trong "bag".

- ❖ TF-IDF

- tf-idf, viết tắt của thuật ngữ tiếng Anh term frequency – inverse document frequency, của một từ là một con số thu được qua thống kê thể hiện mức độ quan trọng của từ này trong một văn bản, mà bản thân văn bản đang xét nằm trong một tập hợp các văn bản.
- TF- term frequency – tần số xuất hiện của 1 từ trong 1 văn bản. Cách tính:

$$tf(t, d) = \frac{f(t, d)}{\max\{f(w, d) : w \in d\}}$$

Trong đó:

- $f(t, d)$ - số lần xuất hiện từ t trong văn bản d
- $\max\{f(w, d) : w \in d\}$ - số lần xuất hiện nhiều nhất của một từ bất kỳ trong văn bản.

- ❖ IDF – inverse document frequency. Tần số nghịch của 1 từ trong tập văn bản (corpus).

- Tính IDF để giảm giá trị của những từ phổ biến. Mỗi từ chỉ có 1 giá trị IDF duy nhất trong tập văn bản. Cách tính:

$$idf(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

Trong đó:

- $\text{idf}(t, D)$: giá trị idf của từ t trong tập văn bản
- $|D|$: Tổng số văn bản trong tập D
- $|\{d \in D : t \in d\}|$: thể hiện số văn bản trong tập D có chứa từ t .

❖ Cụ thể, chúng ta có công thức tính tf-idf hoàn chỉnh như sau: $\text{tfidf}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D)$

- Những từ có giá trị TF-IDF cao là những từ xuất hiện nhiều trong văn bản này, và xuất hiện ít trong các văn bản khác. Việc này giúp lọc ra những từ phổ biến và giữ lại những từ có giá trị cao (từ khoá của văn bản đó).
- Tiến hành cài đặt tf, idf, tf-idf :

```
26 def compute_TF(word_count_dict, bow):
27     tf_dict = {}
28     bow_count = len(bow)
29     for word, count in word_count_dict.items():
30         tf_dict[word] = count/float(bow_count)
31
32     return tf_dict
33
34
35 def compute_IDF(doc_list):
36     import math
37     idf_dict = {}
38     N = len(doc_list)
39
40     # đếm số lần xuất hiện của từ. Khởi tạo ban đầu bằng 0
41     idf_dict = dict.fromkeys(doc_list.keys(), 0)
42
43     for word, count in doc_list.items():
44         if count > 0:
45             idf_dict[word] += 1
46
```

Tính tf, idf

- Tính tf-idf bằng cách nhân tf với idf lại với nhau:

```
52 def compute_TFIDF(tf_bow, idfs):
53     tfidf = {}
54     for word, val in tf_bow.items():
55         tfidf[word] = val*idfs[word]
56     return tfidf
57
58
```

Tính tf-idf

```
{'CMND/': 1, 'trường': 63, 'hội': 27, 'Hung': 9, 'Chủ': 6, 'tháng': 35, 'tra': 12, 'trọng': 2, '(vi': 1, 'nước': 2, 'lục': 31, ')': 3, 'dẫn': 1, 'doanh.': 1}
Kết quả tf: {'CMND/': 0.0027397260273972603, 'trường': 0.1726027397260274, 'hội': 0.07397260273972603, 'Hung': 0.024657534246575342, 'Chủ': 0.016438356164383561, 'tháng': 0.02100078005060644, 'tra': 0.012100078005060644, 'trọng': 0.002100078005060644, '(vi': 0.002100078005060644, 'nước': 0.002100078005060644, 'lục': 0.002100078005060644, ')': 0.002100078005060644, 'dẫn': 0.002100078005060644, 'doanh.': 0.002100078005060644}
Kết quả idf: {'CMND/': 7.665284718471351, 'trường': 7.665284718471351, 'hội': 7.665284718471351, 'Hung': 7.665284718471351, 'Chủ': 7.665284718471351, 'tháng': 7.665284718471351, 'tra': 7.665284718471351, 'trọng': 7.665284718471351, '(vi': 7.665284718471351, 'nước': 7.665284718471351, 'lục': 7.665284718471351, ')': 7.665284718471351, 'dẫn': 7.665284718471351, 'doanh.': 7.665284718471351}
Kết quả tf-idf: {'CMND/': 0.02100078005060644, 'trường': 1.3230491431882059, 'hội': 0.5670210613663739, 'Hung': 0.18900702045545795, 'Chủ': 0.126004680303638, 'tháng': 0.126004680303638, 'tra': 0.126004680303638, 'trọng': 0.126004680303638, '(vi': 0.126004680303638, 'nước': 0.126004680303638, 'lục': 0.126004680303638, ')': 0.126004680303638, 'dẫn': 0.126004680303638, 'doanh.': 0.126004680303638}
```

Kết quả tính tf-idf trong một bài tuyển dụng ngẫu nhiên

❖ Tìm kiếm

```
1 import sqlite3
2 def Search():
3     Search_words = str(input("Nhập từ khoá tìm kiếm: "))
4     Search_words = "%{}%".format(Search_words)
5
6     conn = sqlite3.connect("data/DBTimviec.db")
7     query = """SELECT TITLE, LINK, DESCRIPTION from JOBS_DATA
8             where CONTENT like ? OR TITLE LIKE ? OR JOB_NAME LIKE ? OR DESCRIPTION LIKE ?
9             """
10    a = conn.execute(query, (Search_words, Search_words, Search_words, Search_words)).fetchall()
11    conn.commit()
12    for i in a:
13        print("-----")
14        for item in i:
15            print(item)
16 if __name__ == '__main__':
17     Search()
```

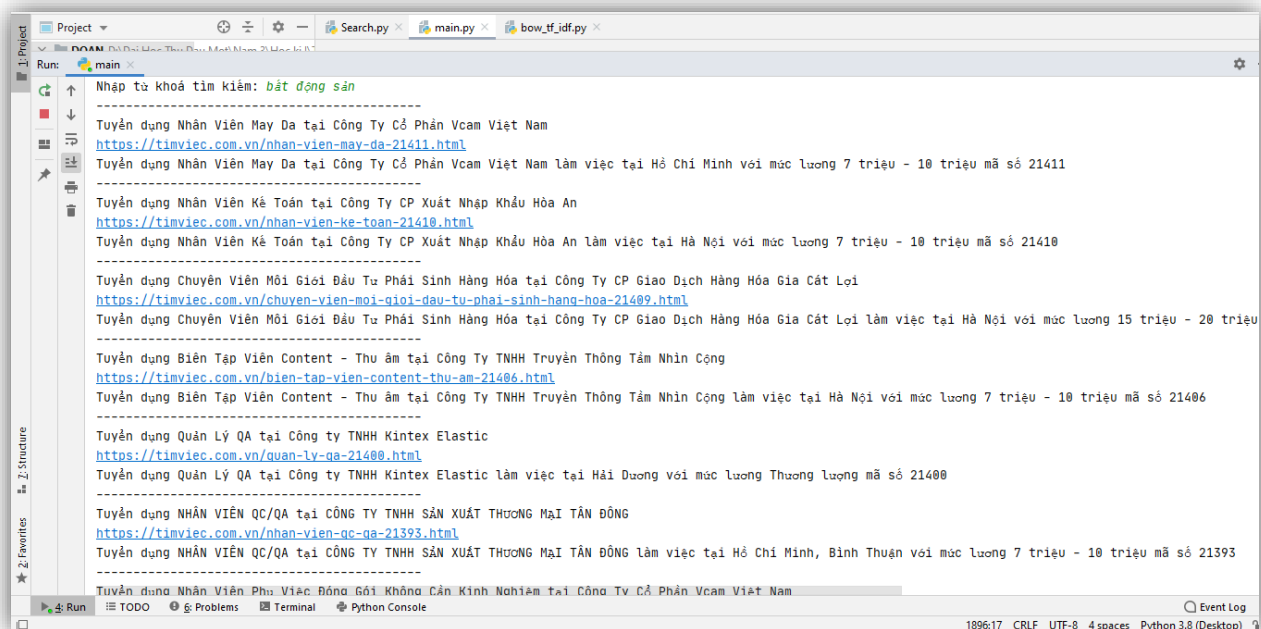
Tìm kiếm các thông tin liên quan đến từ khoá tìm kiếm

```

Nhập từ khoá tìm kiếm: giáo viên
-----
Tuyển dụng Chuyên Viên Tư Vấn Giáo Dục tại Englishnow Global
https://timviec.com.vn/chuyen-vien-tu-van-giao-duc-21402.html
Tuyển dụng Chuyên Viên Tư Vấn Giáo Dục tại Englishnow Global làm việc tại Hà Nội với mức lương 15 triệu - 20 triệu (Có phần trăm hoa hồng) mã số 21402
-----
Tuyển dụng Nhân Viên Tư Vấn Tuyển Sinh Du Học (Làm việc tại Hà Nội) tại Công Ty CP Đầu Tư ISTAR
https://timviec.com.vn/nhan-vien-tu-van-tuyen-sinh-du-hoc-lam-viec-tai-ha-noi-21336.html
Tuyển dụng Nhân Viên Tư Vấn Tuyển Sinh Du Học (Làm việc tại Hà Nội) tại Công Ty CP Đầu Tư ISTAR làm việc tại Hà Nội với mức lương 12 triệu - 15 triệu (Có phần trăm hoa hồng) mã số 21336
-----
Tuyển dụng Giáo Viên Tiếng Anh tại Công Ty TNHH Language Academy
https://timviec.com.vn/giao-vien-tieng-anh-21206.html
Tuyển dụng Giáo Viên Tiếng Anh tại Công Ty TNHH Language Academy làm việc tại Hà Nội với mức lương 10 triệu - 12 triệu mã số 21206
-----
Tuyển dụng Cộng Tác Viên Dự Án tại CÔNG TY CỔ PHẦN TRUYỀN THÔNG VINAROMA VIỆT NAM
https://timviec.com.vn/cong-tac-vien-du-an-21005.html
Tuyển dụng Cộng Tác Viên Dự Án tại CÔNG TY CỔ PHẦN TRUYỀN THÔNG VINAROMA VIỆT NAM làm việc tại Hà Nội với mức lương Thương lượng (Có phần trăm hoa hồng) mã số 21005
-----
Tuyển dụng Giáo Viên STEM tại Công ty Cổ phần Giáo dục KDI (KDI Education)
https://timviec.com.vn/giao-vien-stem-19461.html
Tuyển dụng Giáo Viên STEM tại Công ty Cổ phần Giáo dục KDI (KDI Education) làm việc tại Hồ Chí Minh với mức lương Thương lượng mã số 19461
-----

```

Kết quả tìm kiếm với từ khoá “giáo viên”



```

Project
Run: main
Nhập từ khoá tìm kiếm: bất động sản
-----
Tuyển dụng Nhân Viên May Da tại Công Ty Cổ Phần Vcam Việt Nam
https://timviec.com.vn/nhan-vien-may-da-21411.html
Tuyển dụng Nhân Viên May Da tại Công Ty Cổ Phần Vcam Việt Nam làm việc tại Hồ Chí Minh với mức lương 7 triệu - 10 triệu mã số 21411
-----
Tuyển dụng Nhân Viên Kế Toán tại Công Ty CP Xuất Nhập Khẩu Hòa An
https://timviec.com.vn/nhan-vien-ke-toan-21410.html
Tuyển dụng Nhân Viên Kế Toán tại Công Ty CP Xuất Nhập Khẩu Hòa An làm việc tại Hà Nội với mức lương 7 triệu - 10 triệu mã số 21410
-----
Tuyển dụng Chuyên Viên Môi Giới Đầu Tư Phái Sinh Hàng Hóa tại Công Ty CP Giao Dịch Hàng Hóa Gia Cát Lợi
https://timviec.com.vn/chuyen-vien-moi-gioi-dau-tu-phai-sinh-hang-hoa-21409.html
Tuyển dụng Chuyên Viên Môi Giới Đầu Tư Phái Sinh Hàng Hóa tại Công Ty CP Giao Dịch Hàng Hóa Gia Cát Lợi làm việc tại Hà Nội với mức lương 15 triệu - 20 triệu (Có phần trăm hoa hồng) mã số 21409
-----
Tuyển dụng Biên Tập Viên Content - Thu âm tại Công Ty TNHH Truyền Thông Tầm Nhìn Cộng
https://timviec.com.vn/bien-tap-vien-content-thu-am-21406.html
Tuyển dụng Biên Tập Viên Content - Thu âm tại Công Ty TNHH Truyền Thông Tầm Nhìn Cộng làm việc tại Hà Nội với mức lương 7 triệu - 10 triệu mã số 21406
-----
Tuyển dụng Quản Lý QA tại Công ty TNHH Kintex Elastic
https://timviec.com.vn/quan-ly-qa-21400.html
Tuyển dụng Quản Lý QA tại Công ty TNHH Kintex Elastic làm việc tại Hải Dương với mức lương Thương lượng mã số 21400
-----
Tuyển dụng NHÂN VIÊN QC/QA tại CÔNG TY TNHH SẢN XUẤT THƯƠNG MẠI TÂN ĐÔNG
https://timviec.com.vn/nhan-vien-qc-qa-21393.html
Tuyển dụng NHÂN VIÊN QC/QA tại CÔNG TY TNHH SẢN XUẤT THƯƠNG MẠI TÂN ĐÔNG làm việc tại Hồ Chí Minh, Bình Thuận với mức lương 7 triệu - 10 triệu mã số 21393
-----
Tuyển dụng Nhân Viên Phụ Việc Bón Gội Khônda Cần Kinh Nghiệm tại Công Ty Cổ Phần Vcam Việt Nam
-----

```

Kết quả tìm kiếm với từ khoá “bất động sản”

PHẦN 2: TỰ CHẤM

Nội dung	Yêu cầu	Thang điểm	Điểm Thành viên 1
Phần 1	Dữ liệu	3 điểm	3
	Phân tích	3 điểm	3
	Dự báo	4 điểm	3
Tổng		10	9

DANH MỤC TÀI LIỆU THAM KHẢO

<https://www.nltk.org/book/ch01.html>

<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

<https://newspaper.readthedocs.io/en/latest/>

<https://viblo.asia/p/xu-ly-ngon-ngu-tu-nhien-voi-python-p4-WAyK8RymIxx>

<https://codetudau.com/machine-learning-nlp-scikit-learn/index.html>

<https://codetudau.com/bag-of-words-tf-idf-xu-ly-ngon-ngu-tu-nhien/index.html>

<https://www.sqlitetutorial.net/>

https://maelfabien.github.io/machinelearning/NLP_2/#2-term-frequency-inverse-document-frequency-tf-idf