ĐẠI HỌC BÁCH KHOA HÀ NỘI

BÁO CÁO
KHOA HỌC DỮ LIỆU
Phân tích thị trường tuyển dụng trong lĩnh vực Công nghệ thông tin

Vũ Việt Bách 20200061
Phan Văn Đạt 20200130
Lê Đình Thái Sơn 20200529
Nguyễn Văn Thọ 20204694
Nguyễn Minh Tuấn 20204700

Giảng viên hướng dẫn: TS. Trần Việt Trung

Chữ ký của GVHD

Khoa: Khoa học máy tính
Trường: Công nghệ Thông tin và Truyền thông

Hà Nội, 12-2023

LỜI CẢM ƠN

Chúng em xin được gửi lời cảm ơn đến TS. Trần Việt Trung cùng các thầy, cô: PGS. TS. Thân Quang Khoát, TS. Nguyễn Thị Oanh và TS. Bùi Thị Mai Anh đã giảng dạy và hỗ trợ chúng em trong quá trình học và thực hiện bài tập lớn học phần Nhập môn Khoa học Dữ liệu, giúp chúng em hoàn thành tốt và đầy đủ nhiệm vụ của học phần này.

MỤC LỤC

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG BIỂU

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

1.1. Đặt vấn đề

Thị trường lao động Việt Nam đang sôi nổi hơn bao giờ hết. Với việc chúng ta đang sống trong thời đại công nghệ 4.0, có rất nhiều vị trí công việc liên quan đến ngành công nghệ thông tin cần tuyển dụng để đáp ứng đủ nhu cầu của thời đại. Cùng với đó, rất nhiều sinh viên mới ra trường có nhu cầu, mong muốn tìm kiếm được một công việc phù hợp với ngành mình theo học. Tuy nhiên, thực trạng đáng buồn hiện nay là đang có quá nhiều trang web đóng vai trò "trung gian", thay các công ty đăng tin tuyển dụng. Việc có quá nhiều nguồn đăng, qúa nhiều tin tuyển dụng đã khiến những người đang tìm việc đã khó tìm việc phù hợp nay còn khó hơn vì có quá nhiều thông tin khiến họ không nắm được hết, chưa hiểu rõ việc mình cần chuẩn bị những kỹ năng gì để có thể đáp ứng cho công việc. Ngoài ra, vì không nắm được thông tin thị trường tuyển dụng tổng quan, không có sự so sánh giữa các công ty có nhu cầu cho cùng một loại công việc nên một số người luôn có thắc mắc, liệu rằng công việc họ sắp nộp hồ sơ xin việc có mức lương phù hợp với sức mình bỏ ra, hay chế độ đãi ngộ của công ty liệu rằng đủ tốt, mình có thể phát triển kỹ năng khi làm việc ở công ty hay không,…

Nhận thấy nhu cầu như trên, chúng em đề xuất thực hiện đề tài "Phân tích thị trường tuyển dụng trong lĩnh vực Công nghệ thông tin" với mong muốn giúp cho những người đang tìm việc, nhất là những bạn sinh viên có cái nhìn tổng quan về thị trường tuyển dụng nhờ các số liệu thống kê, phân tích về thị trường, phân tích những yêu cầu và kỹ năng cần có cho ngành nghề cụ thể, từ đó giúp định hướng cho họ có thể chuẩn bị tốt các kiến thức, kỹ năng và giúp họ tìm được công việc như ý thích.

1.2. Mục tiêu và đề xuất giải pháp

Với thực trạng đã nêu ở phần 1.1, chúng em đề ra mục tiêu cụ thể của dự án như sau:

1. Phân tích tổng quan thị trường tuyển dụng trong lĩnh vực Công nghệ thông tin, bằng việc trả lời câu hỏi liên quan đến:
   - Danh sách lĩnh vực con trong lĩnh vực Công nghệ thông tin.
   - Tương quan giữa lương và số năm kinh nghiệm.
   - Tương quan giữa lương và lĩnh vực con.
   - Phân bố số lượng việc làm theo cấp bậc.
   - Phân bố số lượng việc làm theo hình thức làm việc.
   - Danh sách những điều kiện làm việc phổ biến.
2. Phân tích lĩnh vực con trong ngành Công nghệ thông tin, trả lời câu hỏi liên quan đến:
   - Yêu cầu chính của công việc.
   - Mô tả chính của công việc.
   - Những từ khoá chính của công việc.

Để có thể đáp ứng được mục tiêu và giải quyết được các vấn đề nêu trên, trước hết chúng em đề xuất sử dụng các công cụ, thư viện có sẵn như Scrapy, Selenium,… để cào dữ liệu từ các trang web uy tín có nội dung bài đăng tin tuyển dụng. Tiếp theo sẽ sử dụng ngôn ngữ lập trình Python cùng các thư viện có sẵn để tiền xử lý và tích hợp dữ liệu. Với mục tiêu phân tích tổng quan thị trường tuyển dụng, chúng em đề xuất sử dụng thư viện Matplotlib và Seaborn để trực quan hóa dữ liệu, từ các biểu đồ được xây dựng tiến hành phân tích thị trường. Với mục tiêu còn lại, chúng em đề xuất phương pháp mô hình hóa chủ đề sử dụng BERT để truy xuất các yêu cầu và kỹ năng cần thiết cho công việc.

1.3. Bố cục báo cáo

Phần còn lại của báo cáo được tổ chức như sau:

Chương 2 thực hiện các bước đầu tiên về thu thập và tiền xử lý dữ liệu. Trong chương này, tiến hành khảo sát các trang web có dữ liệu tuyển dụng, sử dụng các công cụ và thư viện để tiến hành thu thập dữ liệu, tiền xử lý và tích hợp dữ liệu đã thu thập được.

Chương 3 trình bày về các công nghệ được sử dụng trong dự án, bao gồm các công cụ được sử dụng để trực quan hóa dữ liệu. Cùng với đó, chương này cũng sẽ trình bày về kỹ thuật mà chúng em sử dụng để có thể đưa ra các phân tích về ngành nghề.

Chương 4 trình bày về các công nghệ được sử dụng để tiến hành triển khai hệ thống.

Chương 5 đưa ra kết quả trực quan hóa dữ liệu cùng những phân tích để hiểu dữ liệu. Cùng với đó, chương này cũng trình bày về kết quả phân tích dữ liệu và kết quả triển khai hệ thống.

Chương 6 là chương cuối cùng, tổng kết lại kết quả đã phát triển và phân tích các hướng đi mới cho phép cải thiện hơn những tác vụ được thực hiện trong dự án.

CHƯƠNG 2. THU THẬP VÀ TIỀN XỬ LÝ DỮ LIỆU

2.1. Thu thập dữ liệu

Dựa trên mục tiêu và yêu cầu bài toán mà chúng em đã nêu ở chương 1, nhóm quyết định thu thập dữ liệu tin tuyển dụng, gồm các trường thuộc tính được liệt kê dưới đây:

Bảng 2.1: Mô tả chi tiết yêu cầu về dữ liệu.

STT	Tên trường thuộc tính	Mô tả ý nghĩa thuộc tính	Ví dụ minh họa
1	Liên kết (URL) tới bài đăng của nhà tuyển dụng	Cho phép truy cập trực tiếp vào bài đăng của nhà tuyển dụng, từ đó có nhiều thông tin hơn, ví dụ: thông tin về cách nộp đơn xin việc, thời hạn,…	https://itviec.com/it-jobs/mid-sr-java-developer-english-required-up-to-3200-rakuten-fintech-vietnam-co-ltd-4058
2	Tên công việc	Mô tả chung về công việc được đăng tuyển	Fullstack Developer
3	Mức lương	Mô tả mức lương có thể khi ứng viên nhận công việc này	Từ 5 – 10 triệu đồng
4	Mô tả công việc	Mô tả chi tiết công việc, giúp ứng viên có được góc nhìn chi tiết nhất về những gì mà mình phải làm khi nhận được công việc	Định kỳ thực hiện việc kiểm tu bảo dưỡng thiết bị phần cứng, cài đặt và hỗ trợ phần mềm tại các văn phòng làm việc của Công ty kịp thời, chính xác, an toàn và hiệu quả;…
5	Loại công việc	Cho biết công việc thuộc lĩnh vực con nào của ngành công nghệ thông tin	IT phần mềm
6	Đãi ngộ công việc	Cho biết thông tin về mức đãi ngộ, quyền lợi khi làm công việc này	Thưởng lương tháng thứ 13
7	Hình thức làm việc	Cho biết loại hình làm việc của công việc này là toàn thời gian (full-time), bán thời gian (part-time) hay linh hoạt	Toàn thời gian
8	Yêu cầu số năm kinh nghiệm	Cho biết số năm kinh nghiệm đã tích lũy trong quá trình làm việc để có thể ứng tuyển vào công việc này	Từ 3-5 năm kinh nghiệm
9	Cấp bậc của công việc	Cho biết chức vụ khi nhận công việc: nhân viên, quản lý, trưởng phòng,…	Nhân viên, quản lý
10	Yêu cầu công việc	Cho biết các yêu cầu mà ứng viên cần có để có thể nhận công việc	Tốt nghiệp đại học chuyên ngành công nghệ thông tin, chấp nhận ứng viên mới ra trường, ưu tiên có từ 1 năm kinh nghiệm tại vị trí tương đương.
11	Kỹ năng yêu cầu	Là một phần của yêu cầu công việc, nhưng cụ thể hơn, nói về những kỹ năng mà ứng viên phải thông thạo để có thể nhận công việc	Python, Java
12	Khu vực làm việc	Cho biết địa điểm làm việc của công việc đăng tuyển	Hà Nội, Hồ Chí Minh

Qua tìm hiểu cũng như phân tích một số trang web có tin tuyển dụng về công nghệ thông tin, chúng em đề xuất cào dữ liệu tin tuyển dụng từ các trang web sau:

Bảng 2.2: Thông tin về các trang web cùng với công cụ, thư viện được sử dụng để cào dữ liệu.

STT	Tên trang web	Đường dẫn (URL) trang web	Công cụ, thư viện sử dụng để cào dữ liệu
1	VietnamWorks	https://www.vietnamworks.com	Selenium
2	itviec	https://itviec.com/	Scrapy
3	TopCV	https://www.topcv.vn/	Scrapy
4	Vieclam24h	https://vieclam24h.vn/	Scrapy
5	Timviec365	https://timviec365.vn/	Scrapy

Bảng trên cũng đã nêu chi tiết các công cụ và thư viện mà chúng em sử dụng để thu thập dữ liệu từ các trang web trên, điển hình là Scrapy và Selenium.

2.1.1. Scrapy

Thư viện Scrapy là một framework mã nguồn mở được viết bằng Python để thu thập dữ liệu từ web. Scrapy bao gồm các thành phần chính sau:

- **Spiders:** Đây là các chương trình chịu trách nhiệm thu thập dữ liệu từ các trang web. Spiders được viết bằng Python và có thể được tùy chỉnh để thu thập dữ liệu từ các trang web theo cách cụ thể.
- **Downloader:** Đây là thành phần chịu trách nhiệm tải các trang web xuống máy tính. Downloader sử dụng các kỹ thuật như caching và proxy để tăng tốc độ và hiệu quả của việc tải xuống.
- **Parsers:** Đây là thành phần chịu trách nhiệm phân tích các trang web đã tải xuống và trích xuất dữ liệu cần thiết. Parsers sử dụng các kỹ thuật như XPath, CSS selectors và regular expressions để trích xuất dữ liệu.
- **Pipelines:** Đây là thành phần chịu trách nhiệm xử lý dữ liệu đã được trích xuất. Pipelines có thể được sử dụng để lưu trữ dữ liệu vào cơ sở dữ liệu, xử lý dữ liệu thô hoặc thực hiện các tác vụ khác.

```python
class TimViec365Spider(scrapy.spiders.SitemapSpider):
    name = "tim_viec_365"
    allowed_domains = ['timviec365.vn']
    sitemap_urls = ['https://timviec365.vn/sitemap.xml']
    sitemap_follow = [r'sitemap-job\d+\.xml']


    def sitemap_filter(self, entries):
        for entry in entries:
            date_time = datetime.strptime(entry["lastmod"], "%Y-%m-%dT%H:%M:%S%z")
            if date_time >= datetime(2023, 9, 23, 0, 0, 29, 0, custom_tz):
                yield entry

    def parse(self, response):
        job = Job()
        job['url'] = response.url
        job['job'] = response.xpath('*//div[@class="com_info"]//h1[@class="com_post"]/text()').get()
        job['company'] = response.xpath(
            '*//div[@class="com_info"]//a[@class="com_name"]//p[@class="com_name_text"]/text()').get()
        job['location'] = response.xpath('*//p[contains(text(), "Địa điểm làm việc")]/following-sibling::di
        job['benefits'] = response.xpath('*//div[h2[contains(text(), "QUYỀN LỢI")]]/following-sibling::div[
        job['requirements'] = response.xpath('//div[contains(@class, "text_content") and contains(@class, "
        job['description'] = response.xpath('//h2[contains(text(), "MÔ TẢ CÔNG VIỆC")]/../following-sibling
        job['industry'] = response.xpath('//p[contains(text(), "Lĩnh vực")]/a/text()').get()
        job['category'] = response.xpath('*//p[contains(text(),"Ngành nghề:")]/a/text()').getall()
        job['level'] = None
```

Hình 2.1: Ví dụ một đoạn mã nguồn dùng Scrapy để cào dữ liệu web từ trang https://timviec365.vn.

2.1.2. Selenium

Thư viện Selenium là một framework mã nguồn mở được viết bằng Python để tự động hóa các ứng dụng web. Selenium có thể được sử dụng cho nhiều mục đích khác nhau, bao gồm:

- **Kiểm tra tự động:** Selenium có thể được sử dụng để viết các kịch bản kiểm tra tự động để kiểm tra các ứng dụng web.
- **Tự động hóa dữ liệu:** Selenium có thể được sử dụng để tự động hóa các tác vụ như điền biểu mẫu, tải lên tệp và tạo nội dung.

Selenium bao gồm các thành phần chính sau:

- **WebDriver:** Đây là thành phần chịu trách nhiệm điều khiển các trình duyệt web. WebDriver hỗ trợ nhiều trình duyệt web khác nhau, bao gồm Chrome, Firefox, Edge và Safari.
- **API:** Selenium cung cấp một bộ API để kiểm soát các trình duyệt web, cho phép thực hiện các tác vụ như điều hướng trang web, tương tác với các yếu tố giao diện người dùng và kiểm tra trạng thái của ứng dụng web.
- **Extensions:** Selenium có sẵn các tiện ích mở rộng cho nhiều trình IDE và môi trường phát triển tích hợp (IDE). Extensions này cung cấp các tính năng bổ sung để giúp viết và chạy các kịch bản Selenium.

Ưu điểm của Selenium so với Scrapy là có thể truy cập và lấy dữ liệu từ những pop-up hay những đoạn mã Javascript mà chỉ khi người dùng nhấp chuột hay cuộn chuột vào thì mới xuất hiện, nhờ các hàm giả lập tương tác người dùng. Thông thường, người ta sử dụng kết hợp Selenium và Scrapy (hoặc với một số thư viện giúp thu thập dữ liệu, chẳng hạn BeautifulSoup) trong việc cào dữ liệu để được hiệu suất tốt nhất.

```python
def crawl_url(driver, url, output_file):
    driver.get(url)

    time.sleep(1)
    driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
    time.sleep(2)

    html_content = driver.page_source
    soup = BeautifulSoup(html_content, 'html.parser')

    elements = soup.select('div.sc-ikHNZD.loEdRV a')
    elements = [ROOT_URL+element['href'] for element in elements]

    if len(elements) != 50:
        with open("error.txt", "a") as f:
            f.write(url + '\n')
    else:
        write_row(output_file, '\n'.join(elements))
```

Hình 2.2: Ví dụ một đoạn mã nguồn dùng Selenium kết hợp với BeautifulSoup để cào dữ liệu web.

Bộ dữ liệu thu thập được sau đó được lọc ra công việc theo ngành công nghệ thông tin (viết tắt: **IT**, tiếng Anh: *Information Technology*) dựa trên trường "**loại công việc**" (trường *category*).

Kết quả thu được gồm 3 298 dòng dữ liệu ứng với 3298 công việc IT và các lĩnh vực con trong ngành IT. Bảng dưới đây cung cấp chi tiết thông tin về dữ liệu thu thập được từ các trang web kể trên.

Bảng 2.3: Thông tin về bộ dữ liệu thu thập được từ các nguồn.

STT	Bộ dữ liệu	Số lượng dữ liệu	Số lượng công việc IT	Thành viên phụ trách thu thập	Toàn bộ đều là công việc IT?
1	vietnamworks	8490	751	Vũ Việt Bách	
2	ITviec	764	764	Nguyễn Văn Thọ	x
3	timviec365	889	24	Nguyễn Minh Tuấn	
4	vieclam24h	1333	240	Lê Đinh Thái Sơn	
5	TopCV	1721	1721	Phan Văn Đạt	x

Các lĩnh vực con của các công việc IT trong dữ liệu thu thập được gồm:

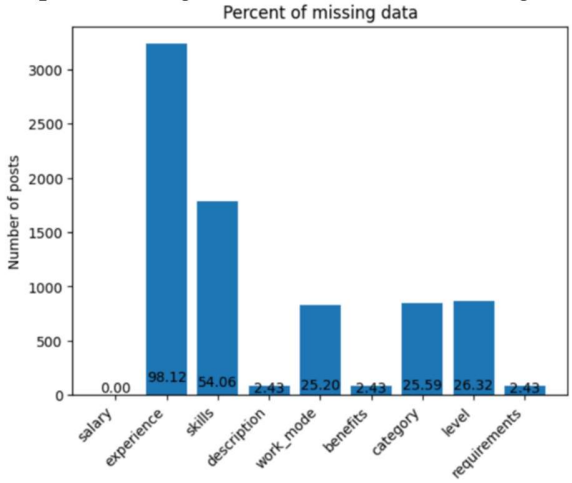
- System/Cloud/DevOps Engineer
- UX/UI Design
- IT phần cứng
- IT phần mềm
- Data Engineer/Data Analyst/AI
- Quản lý dự án công nghệ thông tin
- Viễn thông
- Chuyển đổi số
- Bảo mật công nghệ thông tin
- IT Support/Help Desk
- Quản trị cơ sở dữ liệu
- QA/QC/Software Testing
- Phân tích kinh doanh/Phân tích hệ thống

2.2. Tiền xử lý dữ liệu

Dữ liệu thu thập được còn nhiều giá trị thiếu, chưa được tích hợp do thu thập từ nhiều nguồn. Vì thế, tiến hành các bước để làm sạch và tích hợp dữ liệu.

Đầu tiên, tiến hành lọc và thống kê những giá trị thiếu (missing value). Những giá trị bị thiếu có thể do một số nguyên nhân: i) do trang web không chia sẻ một số thông tin hoặc không cung cấp thông tin; ii) do nhà tuyển dụng đăng bài không theo định dạng đã được quy định của trang web mà các trường thuộc tính nằm lồng vào trong thuộc tính khác và iii) do kỹ năng thu thập dữ liệu còn hạn chế, không bao quát hết các khả năng có thể xảy ra.

Biểu đồ cột dưới đây mô tả tỉ lệ phần trăm giá trị/dữ liệu bị thiếu trong dữ liệu thu thập được.



Hình 2.3: Biểu đồ mô tả tỉ lệ phần trăm giá trị thiếu của các trường thuộc tính trong dữ liệu thu thập được.

Đánh giá chung, trường thuộc tính "kinh nghiệm" (*experience*) bị thiếu nhiều do thông tin về số năm kinh nghiệm hay được nhà tuyển dụng viết trong phần "yêu cầu công việc" (*requirements*). Một số trường thuộc tính khác, ví dụ "kỹ năng làm việc" (*skills*) hay "cấp độ" (*level*) cũng bị thiếu khá nhiều, mà nguyên nhân chủ yếu có thể cho rằng là do những nội dung này thường được lồng vào nội dung yêu cầu công việc.

Để khắc phục vấn đề dữ liệu thiếu, chúng em thực hiện một số thao tác phân tích, từ đó đưa ra giải pháp khôi phục và điền dữ liệu thiếu phù hợp.

Đối với những trường thuộc tính có khả năng "có thông tin" trong phần yêu cầu công việc do những nguyên nhân đề cập ở trên, chúng em sẽ sử dụng trường "*requirements*" và các trường liên quan để trích xuất ra thông tin này. Ngoài ra, do tính chất thu thập dữ liệu trên nhiều nguồn khác nhau nên dữ liệu có thể không đồng nhất, chẳng hạn như đơn vị tính tiền lương không giống nhau (sử dụng đơn vị tiền tệ của Hoa Kỳ và Việt Nam), các từ khóa giống nhau nhưng in hoa/in đậm khác thường, hay những từ cùng chỉ định danh "Thành phố Hồ Chí Minh" nhưng khác nhau về biểu diễn: "TP.HCM", "Ho Chi Minh City", "Hồ Chí Minh",… gây khó khăn rất nhiều trong việc thống kê và phân tích. Do đó, chúng em tiến hành tích hợp và chuẩn hóa dữ liệu để giúp dữ liệu được "sạch" và giúp dễ dàng hơn trong việc đánh giá, phân tích sau này.

Bảng dưới đây nêu rõ chi tiết các bước mà chúng em đã thực hiện trên từng trường thuộc tính, với mục đích làm sạch, khôi phục/điền dữ liệu thiếu và tích hợp dữ liệu từ các nguồn khác nhau.

Bảng 2.4: Đặc điểm các trường thuộc tính và các thao tác tiền xử lý, tích hợp dữ liệu.

STT	Trường thông tin	Dạng dữ liệu	Vấn đề	Phương pháp thực hiện
1	Mức lương (salary)	Văn bản	Dữ liệu về lương là một chuỗi văn bản, các đơn vị đo lường và định dạng không giống nhau, chẳng hạn "*5-10 triệu*", "*Từ $200 đến $1.000*", "*Lên tới 30.000.000 đồng*",…	• **Đơn vị chuẩn hóa:** triệu đồng • Sử dụng biểu thức chính quy để trích xuất ra các con số theo đúng định dạng. • Đối với những dữ liệu lương là khoảng (*từ X đến Y*), tạo ra hai thuộc tính "**min_salary**" và "**max_salary**" để lưu các giá trị này • Đối với những dữ liệu dạng "*Lên tới Y*" "*Đến Y*",..., lưu **min_salary** bằng 0 và lưu giá trị *Y* vào **max_salary**. Lưu ngược lại đối với những dữ liệu dạng "*Từ X* ",... • **Số liệu đô-la Mỹ** (có đơn vị $, USD) thì nhân thêm với tỉ giá đô-la (24260 đồng cho một đô-la) rồi chuyển về đơn vị chuẩn hóa (triệu đồng). • **Điền dữ liệu thiếu:** Các trường hợp không có dữ liệu lương, thực hiện tìm kiếm trong trường "**requirements**" sử dụng các biểu thức chính quy, nếu tìm thấy thì lưu như trên, còn không thì lưu chung là "*Thỏa thuận*".
2	Số năm kinh nghiệm (experience)	Số nguyên	Dữ liệu bị thiếu khá nhiều, chủ yếu do số năm kinh	• Chỉ lọc lấy phần số trong giá trị (nếu có)

STT	Trường thông tin	Dạng dữ liệu	Vấn đề	Phương pháp thực hiện
			nghiệm không được tách làm một phần riêng mà lồng chung vào mô tả "*yêu cầu công việc*".	• Với công việc nhiều cấp bậc đòi hỏi nhiều mức năm kinh nghiệm, chọn số năm kinh nghiệm cao nhất • Các giá trị: "Không yêu cầu kinh nghiệm" và tương tự được chuyển về 0 năm • **Điền dữ liệu thiếu**: Các trường hợp không có dữ liệu về số năm kinh nghiệm, thực hiện tìm kiếm trong trường "yêu cầu công việc", lấy ra từ "*kinh nghiệm*" và các từ xung quanh, quan sát cách thức của chuỗi từ được lấy ra để lấy ra thông tin về số năm kinh nghiệm.
3	Đãi ngộ, quyền lợi việc làm (benefits)	Văn bản		• Tách thành các câu đơn • Viết thường, loại bỏ các dấu câu như: dấu chấm, phẩy,…
4	Mô tả công việc (description)	Văn bản		• Tách thành các câu đơn • Viết thường, loại bỏ các dấu câu như: dấu chấm, phẩy,…
5	Yêu cầu công việc (requirements)	Văn bản		• Tách thành các câu đơn • Viết thường, loại bỏ các dấu câu như: dấu chấm, phẩy,…
6	Hình thức làm việc (work_mode)	Văn bản	Phần lớn đều có dữ liệu là "*full-time*", "*part-time*" hay "*linh hoạt*", tuy nhiên vẫn có một số nhãn không thể xác định được là hình thức nào như *Theo hợp đồng,…*	• Đối với những giá trị "*Toàn thời gian*" và tương đương, ví dụ "*Full-time*", "*Toàn thời gian cố định*",… thì đồng nhất dạng là "*Full-time*". • Đối với những giá trị "*Bán thời gian*" và tương đương, ví dụ "*Part-time*", "*Bán thời gian cố định*",… thì đồng nhất dạng là "*Part-time*". • Đối với những giá trị thiếu hoặc không thể xác định thuộc hình thức nào, ta thực hiện tìm kiếm trong trường "*description*" để tìm ra các từ khóa phù hợp cho việc xác định hình thức làm việc.
7	Loại công việc (category)	Văn bản	Có lượng lớn tin tuyển dụng thiếu, các kỹ năng trùng nhau về mặt ngữ nghĩa	• Lược giản đi thành những lĩnh vực con, đối với những dữ liệu không có trường này thì sử dụng tên công việc để phân loại công việc về các lĩnh vực con
8	Kỹ năng (skills)	Văn bản	Có nhiều tin tuyển dụng thiếu, các kỹ năng trùng nhau về ngữ nghĩa	• Lược giản thành những kỹ năng chính • Chấp nhận dữ liệu thiếu

STT	Trường thông tin	Dạng dữ liệu	Vấn đề	Phương pháp thực hiện
9	Cấp bậc công việc (level)	Văn bản	Một số tin tuyển dụng bị thiếu trường này, một số chức danh giống nhau về mặt ngữ nghĩa nhưng cách hiển thị là khác nhau.	• Một số chức danh giống nhau được gộp chung ("Quản lý / Giám sát" và "Quản lý nhóm- giám sát",) • Khái quát hoá chức danh ("Trưởng/Phó phòng" gộp vào "Trưởng phòng")
10	Nơi làm việc (place)	Văn bản	Định dạng địa chỉ không đồng nhất, một số địa chỉ được viết dưới dạng tiếng Việt không dấu do bản tin thu được ở ngôn ngữ Anh. Một số địa danh tỉnh, thành phố viết tắt như TP.HCM, HN,…	• Chuẩn hóa dữ liệu về tên tỉnh, thành phố viết ở Tiếng Việt (trong tập 63 tên tỉnh, thành phố), Quốc tế (nếu địa chỉ ở nước ngoài) và Khác (nếu địa chỉ không rõ thông tin/không thể xác định địa phương). • Các địa chỉ dài thực hiện truy xuất tên tỉnh, thành phố trong địa chỉ đó. • Nếu có từ hai địa điểm trở lên, chấp nhận việc mất mát dữ liệu bằng việc chỉ lấy tỉnh, thành phố của địa điểm đầu tiên được duyệt. • Các địa điểm viết tắt hoặc được viết bằng ngôn ngữ Anh thì dùng ánh xạ để chuyển về Tiếng Việt.

Sau khi thực hiện tiền xử lý và tích hợp dữ liệu, chúng em thu được bộ dữ liệu khá đầy đủ về các trường thông tin như mong muốn, để tiến hành các bước hiểu, trực quan hóa dữ liệu và sử dụng dữ liệu bằng một số kỹ thuật phân tích dữ liệu.

CHƯƠNG 3. PHƯƠNG PHÁP ĐỀ XUẤT

Sau khi thu thập, tích hợp và tiền xử lý dữ liệu, chúng em tiến hành sử dụng một số kỹ thuật phân tích để hiểu thêm về dữ liệu, đồng thời sử dụng công cụ có sẵn để trực quan hóa dữ liệu. Trong chương này, chúng em sẽ giới thiệu và phân tích về các kỹ thuật được sử dụng.

3.1. Trực quan hóa dữ liệu sử dụng thư viện Matplotlib và Seaborn

Trong thế giới ngày nay, việc hiểu được khối lượng dữ liệu khổng lồ mà doanh nghiệp tạo ra mỗi ngày ngày càng quan trọng. Để có thể hiểu dữ liệu, ta cần trực quan hóa dữ liệu trên các yếu tố trực quan như biểu đồ, đồ thị. Trực quan hóa dữ liệu giúp chúng ta phát hiện xu hướng, mẫu hình và các giá trị ngoại lệ, từ đó đúc kết nhanh thông tin chuyên sâu và đưa ra quyết định đúng đắn hơn.

Python hiện là một trong những ngôn ngữ lập trình phổ biến nhất được sử dụng để trực quan hóa dữ liệ, với nhiều thư viện hỗ trợ. Trong dự án này, chúng em sử dụng ngôn ngữ lập trình Python với các thư viện trực quan hóa phổ biến là Matplotlib và Seaborn.

3.1.1. Thư viện Matplotlib

Matplotlib là một thư viện trực quan hóa trong Python cho các mảng 2D, được giới thiệu bởi John Hunter vào năm 2002. Matplotlib là một thư viện đa nền tảng được xây dựng trên các mảng NumPy và được thiết kế để hoạt động với ngăn xếp SciPy rộng hơn. Nó bao gồm các biểu đồ khác nhau như biểu đồ thanh, biểu đồ đường, biểu đồ tần suất và biểu đồ phân tán, mỗi biểu đồ cung cấp một cách khác nhau để hiển thị dữ liệu.
Một số ưu điểm của thư viện Matplotlib có thể kể đến như sau:

- **Linh hoạt và hỗ trợ các hình thức biểu diễn dữ liệu khác nhau:** Matplotlib hỗ trợ biểu diễn dữ liệu trong biểu đồ thanh, đồ thị và các hình thức trực quan hóa khác.
- **Một công cụ mạnh mẽ với nhiều ứng dụng:** Chất lượng trực quan hóa dữ liệu của Matplotlib có thể được sử dụng dưới nhiều hình thức khác nhau, chẳng hạn như tập lệnh Python, powershell, trình duyệt web và tệp Jupyter Notebook.
- **Làm cho việc phân tích dữ liệu dễ dàng hơn:** Do có nhiều tính năng và kết quả chất lượng cao, Matplotlib giúp phân tích dữ liệu dễ dàng và hiệu quả hơn. Nó cũng giúp tiết kiệm thời gian và tài nguyên sẽ dành để phân tích các tập dữ liệu lớn. Không giống như các nền tảng trực quan hóa dữ liệu khác, Matplotlib trong Python chỉ yêu cầu một vài dòng mã để tạo biểu đồ cho các tập dữ liệu.

3.1.2. Thư viện Seaborn

Seaborn là một thư viện để trực quan hóa dữ liệu trong Python, được giới thiệu bởi Michael L. Waskom vào năm 2021. Nó cung cấp một giao diện cấp cao cho Matplotlib và tích hợp chặt chẽ với cấu trúc dữ liệu của Pandas. Các hàm trong thư viện Seaborn cung cấp một API khai báo, hướng tới tập dữ liệu, giúp dễ dàng chuyển đổi câu hỏi về dữ liệu thành đồ họa có thể trả lời chúng. Khi có một bộ dữ liệu và một đặc tả của đồ thị cần tạo, Seaborn tự động ánh xạ các giá trị dữ liệu vào các thuộc tính hình ảnh như màu sắc, kích thước hoặc kiểu dáng, tính toán các biến đổi thống kê và hiển thị đồ thị với nhãn trục thông tin và một chú thích. Bằng cách cung cấp nhiều tùy chọn tùy chỉnh, cùng với việc tiếp cận các đối tượng Matplotlib cơ bản, nó có thể được sử dụng để tạo ra những hình ảnh thống kê chất lượng cao. Tất cả sự phức tạp của Matplotlib được trừu tượng hóa bởi Seaborn, điều này giúp cho Seaborn dễ sử dụng hơn Matplotlib.

Tuy nhiên, Matplotlib cung cấp tính linh hoạt cao hơn về khả năng tùy biến và đôi khi hiệu suất vượt trội hơn so với Seaborn. Do vậy, nhìn chung, Seaborn là sự lựa chọn tốt nhất để trực quan hóa dữ liệu thống kê. Mặt khác, Matplotlib tốt hơn cho nhu cầu tùy chỉnh.

Trong phần trực quan hóa dữ liệu, chúng em sử dụng hai thư viện là Matplotlib và Seaborn, với đa số biểu đồ được vẽ bằng Matplotlib vì thư viện này được sử dụng phổ biến hơn.

3.2. Kỹ thuật phân tích: Mô hình hóa chủ đề

Kỹ thuật phân tích chính của bài toán là mô hình hoá chủ đề (tiếng Anh: *topic modeling*). Bài toán mô hình hoá chủ đề là bài toán nhận đầu vào là tập hợp các tài liệu với mục tiêu là: i) khám phá ra các chủ đề chính từ tập tài liệu đó, ii) với mỗi chủ đề tìm ra từ khoá, nội dung chính và iii) với mỗi tài liệu tìm ra phân bố chủ đề của tài liệu đó.

Bài toán mô hình hóa chủ đề hiện tại có rất nhiều phương pháp giải. Chúng em thống nhất đề xuất sử dụng phương pháp BERTopic. Ưu điểm của phương pháp BERTopic là sử dụng mô hình ngôn ngữ BERT có khả năng học biểu diễn văn bản tốt để mô hình hoá dữ liệu, khắc phục được hạn chế là chỉ dựa vào thống kê từ trong văn bản mà các phương pháp thống kê khác gặp phải.

Các giai đoạn chính của phương pháp BERTopic như sau:

1. Phân cụm tài liệu, tìm ra chủ đề
    - Tạo ra vec-tơ đặc trưng của tài liệu
    - Giảm chiều vectơ đặc trưng
    - Phân cụm tài liệu sử dụng vectơ đặc trưng
2. Tính toán biểu diễn của chủ đề

    Đầu vào cho các chủ đề, mỗi chủ đề có tập hợp tài liệu thuộc chủ đề. Đầu ra là biểu diễn của chủ đề có thể là một tiêu đề, một dãy từ khoá của chủ đề. Phương pháp tìm ra từ khoá của một chủ đề được mô tả dưới đây:
    - Hợp nhất các tài liệu của một chủ đề về một tài liệu đại diện
    - Với một tài liệu đại diện, tách thành các từ (sử dụng thư viện Underthesea NLP)
    - Với mỗi từ tính giá trị c-TF-IDF.

3.2.1. Tạo ra vec-tơ đặc trưng của tài liệu

Sử dụng mô hình BERT để tạo ra vec-tơ đặc trưng của tài liệu.

Bidirectional Encoder Representations from Transformers (BERT) là mô hình ngôn ngữ được Jacob Devlin và cộng sự từ Google phát triển và công bố năm 2018. Khi mới công bố, BERT đạt kết quả tốt nhất trên nhiều bài toán của lĩnh vực xử lý ngôn ngữ tự nhiên như phân loại cảm xúc văn bản, trả lời câu hỏi, nhận dạng thực thể.

BERT có khả năng mô hình ngữ cảnh của câu văn theo hai chiều, tức là cho một đoạn văn gồm các từ $x_1, x_2, \dots, x_n$:

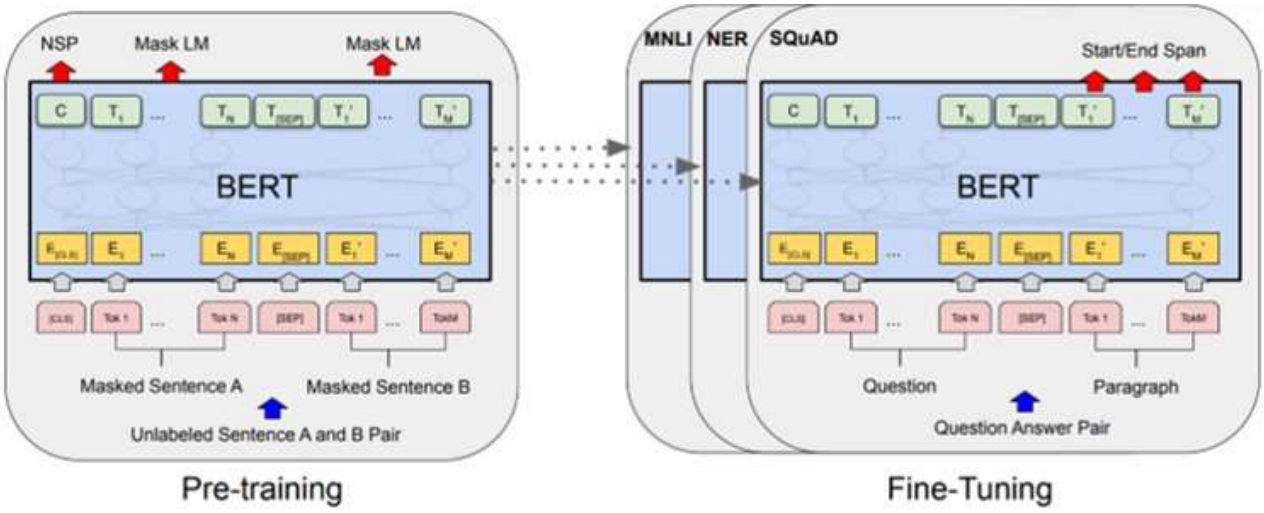
- Mô hình xuôi (Forward autoregressive): dự đoán từ x_i dựa trên các từ trước đó $x_1, \dots x_{i-1}$
- Mô hình ngược (Backward autoregressive): dự đoán từ x_i dựa trên các từ $x_{i+1}, \dots, x_n$

Quá trình huấn luyện BERT gồm hai giai đoạn là tiền huấn luyện (pre-training) và tinh chỉnh (fine-turning).

Trong giai đoạn tiền huấn luyện, ta sử dụng kỹ thuật huấn luyện mô hình giải quyết cho tác vụ dự đoán từ bị che giấu (Masked Language Model), và tác vụ dự đoán câu tiếp theo dựa trên một câu văn có sẵn (Next Sentence Prediction).

Trong giai đoạn tinh chỉnh, hầu hết các siêu tham số giữ nguyên như trong pre-training BERT, chỉ có một số tham số là thay đổi để phù hợp với tác vụ cần giải quyết. Ví dụ, trong bài toán Trả lời câu hỏi (Question Answering, ví dụ: mô hình SQuAD v1.1), mô hình nhận được một câu hỏi liên quan đến chuỗi văn bản và được

yêu cầu đánh dấu câu trả lời trong chuỗi. Sử dụng BERT, một mô hình Hỏi và Đáp có thể được huấn luyện bằng cách học thêm hai vec-tơ đánh dấu điểm bắt đầu và kết thúc của câu trả lời.



Hình 3.1: Tổng quan mô hình BERT.

Ưu điểm của BERT:

- Hiệu suất cao: BERT đã đạt được hiệu suất vượt trội trên nhiều tác vụ NLP, vượt qua các mô hình trước đó.
- Linh hoạt: BERT có thể được sử dụng cho nhiều tác vụ NLP khác nhau.
- Tính khả mở: BERT có thể được mở rộng để xử lý dữ liệu lớn hơn và phức tạp hơn.

Nhược điểm của BERT:

- Yêu cầu nhiều tài nguyên: BERT yêu cầu nhiều tài nguyên để đào tạo và sử dụng.
- Có thể bị thiên vị (bias): BERT có thể bị thiên vị do dữ liệu được sử dụng để đào tạo nó.

3.2.2. Giảm chiều dữ liệu

UMAP (Uniform Manifold Approximation and Projection) là một thuật toán giảm chiều không tuyến tính được phát triển bởi McInnes và Healy (2018). UMAP được thiết kế để bảo tồn cấu trúc cục bộ và toàn cục của dữ liệu gốc trong không gian giảm chiều.

Cách hoạt động của UMAP được mô tả như sau:

UMAP hoạt động bằng cách tìm một ma trận chiếu P từ không gian ban đầu X sang không gian giảm chiều Y. Mục tiêu là làm cho ma trận chiếu P tối ưu hóa hai mục tiêu sau:

- Mục tiêu cục bộ: Giữ cho các điểm dữ liệu gần nhau trong không gian ban đầu X gần nhau trong không gian giảm chiều Y.
- Mục tiêu toàn cục: Giữ cho các điểm dữ liệu có cấu trúc tương tự nhau trong không gian ban đầu X gần nhau trong không gian giảm chiều Y.

UMAP sử dụng một thuật toán tối ưu hóa dựa trên gradient để tìm ma trận chiếu P tối ưu hóa hai mục tiêu trên.

Ưu điểm của UMAP:

- Giữ cấu trúc cục bộ và toàn cục của dữ liệu gốc: UMAP được thiết kế để bảo tồn cấu trúc cục bộ và toàn cục của dữ liệu gốc trong không gian giảm chiều. Điều này làm cho UMAP trở nên hữu ích cho nhiều tác vụ phân tích dữ liệu, chẳng hạn như phân cụm, phân loại và hiển thị dữ liệu.

- Hiệu suất cao: UMAP có thể được thực hiện hiệu quả cho các tập dữ liệu lớn.
- Đơn giản để sử dụng: UMAP có thể được sử dụng dễ dàng với các thư viện khoa học máy tính phổ biến.

Nhược điểm của UMAP:

- Có thể không phù hợp với dữ liệu có cấu trúc phức tạp: UMAP có thể không phù hợp với dữ liệu có cấu trúc phức tạp, chẳng hạn như dữ liệu có nhiều lỗ hoặc đường cong.
- Có thể không bảo toàn khoảng cách: UMAP không đảm bảo bảo toàn khoảng cách giữa các điểm dữ liệu trong không gian ban đầu X trong không gian giảm chiều Y.

3.2.3. Phân cụm tài liệu HDBSCAN

HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) là một thuật toán phân cụm dựa trên mật độ không gian phân cấp. Nó được thiết kế để giải quyết các vấn đề sau:

- Xác định các cụm có hình dạng bất kỳ: HDBSCAN không yêu cầu các cụm phải có hình dạng cầu, điều này làm cho nó trở nên linh hoạt hơn các thuật toán phân cụm dựa trên mật độ khác.
- Xác định các cụm có mật độ khác nhau: HDBSCAN có thể phát hiện các cụm có mật độ khác nhau, bao gồm các cụm thưa thớt, dày đặc và phân tán.
- Xác định các điểm nhiễu: HDBSCAN có thể xác định các điểm nhiễu, là các điểm không thuộc bất kỳ cụm nào.

HDBSCAN hoạt động theo hai giai đoạn. Giai đoạn đầu tiến hành tạo cây phân cấp dựa trên mật độ. Giai đoạn hai thực hiện cắt cây phân cấp để tạo các cụm.

Trong giai đoạn đầu, HDBSCAN sử dụng một thuật toán gọi là DBSCAN (Density-Based Spatial Clustering of Applications with Noise) để tạo một cây phân cấp dựa trên mật độ. DBSCAN bắt đầu bằng cách chọn một điểm dữ liệu ngẫu nhiên làm điểm seed. Sau đó, nó tìm tất cả các điểm dữ liệu khác trong vùng lân cận điểm này. Nếu vùng lân cận này có đủ điểm dữ liệu, thì điểm seed được coi là một điểm hạt. Các điểm hạt được hợp nhất thành một cụm. Quá trình này được lặp lại cho đến khi tất cả các điểm dữ liệu được phân loại.

Trong giai đoạn thứ hai, HDBSCAN sử dụng một tham số gọi là *min_cluster_size* để cắt cây phân cấp thành các cụm. Các cụm có số điểm dữ liệu ít hơn *min_cluster_size* được coi là các điểm nhiễu.

Dưới đây là một số tham số của HDBSCAN.

- **eps**: Là bán kính của vùng lân cận được sử dụng bởi DBSCAN để xác định các điểm hạt.
- **min_samples**: Là số lượng điểm dữ liệu tối thiểu trong vùng lân cận của một điểm dữ liệu để điểm dữ liệu đó được coi là một điểm hạt.
- **min_cluster_size**: Là số điểm dữ liệu tối thiểu trong một cụm.

Ưu điểm của HDBSCAN:

- Linh hoạt, có thể phát hiện các cụm có hình dạng bất kỳ, mật độ khác nhau và có điểm nhiễu.
- Hiệu quả, có thể hoạt động với dữ liệu lớn.
- Mức độ tin cậy cao, có thể phát hiện các cụm chính xác.

Nhược điểm của HDBSCAN:

- Có thể mất nhiều thời gian để thực hiện, đặc biệt là đối với dữ liệu lớn.
- Có thể khó điều chỉnh các tham số, vì chúng có thể ảnh hưởng đáng kể đến kết quả phân cụm.

3.2.4. Tính toán chủ đề sử dụng thuật toán TF-IDF

TF-IDF (Term Frequency - Inverse Document Frequency) là một tác vụ con trong truy xuất thông tin (information retrieval) và trích xuất thông tin (information extraction) nhằm thể hiện tầm quan trọng của một từ đối với một tài liệu (là một phần trong tập tài liệu), đồng thời tính đến mối quan hệ với các tài liệu khác từ

cùng một tập tài liệu. Nó thường được sử dụng bởi một số công cụ tìm kiếm để giúp họ có được kết quả tốt hơn có liên quan hơn đến một truy vấn cụ thể.
TF-IDF gồm hai thành phần:

- **TF (Term Frequency):** Đo lường tần suất xuất hiện của một từ trong một tài liệu (document), phản ánh tầm quan trọng của từ đó đối với nội dung của tài liệu.
- **IDF (Inverse Document Frequency):** Đo lường độ quan trọng của một từ trong toàn bộ tập hợp các tài liệu. Đối với từng tài liệu, việc một từ xuất hiện nhiều lần sẽ có mức độ cao hơn. Tuy nhiên, đối với toàn bộ tập tài liệu, một từ xuất hiện ở nhiều tài liệu thì sẽ trở nên không còn quan trọng nữa.

Trọng số TF-IDF của từ t trong tài liệu d được tính bởi công thức sau:

$$W(t,d) = tf(t,d) \times idf(t,D)$$

trong đó:

- $tf(t,d)$ là tần suất xuất hiện của từ t trong tài liệu d.
- $idf(t,D)$ là độ quan trọng của từ t trong tập tài liệu D.

Ta gọi: n là tổng số tài liệu có trong tập tài liệu D, $df(t,D)$ (document fluency) là số tài liệu trong tập tài liệu D có chứa từ t. Khi đó, ta có công thức tính $idf(t,D)$ như sau (công thức được sử dụng mặc định của thư viện scikit-learn):

$$idf(t,D) = ln\left[\frac{1+n}{1+df(t,D)}\right] + 1$$

Bằng cách kết hợp cả tần suất xuất hiện của từ và tính ngược tần suất của từ, thuật toán TF-IDF giúp đánh giá tầm quan trọng của một từ đối với một tài liệu cụ thể, cung cấp cơ sở vững chắc cho nhiều ứng dụng xử lý ngôn ngữ tự nhiên, bao gồm tìm kiếm thông tin, phân loại văn bản và tóm tắt văn bản.

Trong dự án này, chúng em sử dụng phương pháp mô hình hóa chủ đề để tiến hành phân tích một từ khóa là công việc cho trước, cho ra kết quả là yêu cầu và các kỹ năng cần có tương ứng với từ khóa được truy vấn.

CHƯƠNG 4. CÔNG NGHỆ SỬ DỤNG

Để có thể giúp cho người dùng sử dụng được các tác vụ mà chúng em đã xây dựng, chúng em tiến hành triển khai hệ thống trên nền tảng web. Trong chương 4, chúng em sẽ nêu rõ hơn các công nghệ sử dụng trong dự án này.

4.1. Xây dựng front-end với React

React (còn được gọi là ReactJS hay React.js) là một mã nguồn mở được phát hành vào năm 2013 bởi Jordan Walke – một kỹ sư phần mềm của Facebook. React có công dụng giống một thư viện JavaScript có chức năng tạo giao diện người dùng cho các thành phần trên một trang web. Mục tiêu chính của React là tạo ra sự nhanh gọn, đơn giản và có thể mở rộng. Nó hoạt động trong ứng dụng với giao diện người dùng. React có gồm một số đặc tính nổi bật như: i) có tính khai báo (declarative), ii) cách thức sử dụng rất đơn giản, iii) công nghệ phần mềm được tạo ra dựa trên thành phần (component), iv) có tính khả mở,…

Có thể kể đến một số ưu điểm của React như sau:

1. **Kịch bản đơn giản hóa:** React có một phần mở rộng cú pháp được gọi là JSX, làm cho đánh dấu (mark up) HTML trong thư viện dễ dàng hơn nhiều. Các phím tắt viết của JSX cho phép bạn làm cho mã khóa học của mình đơn giản và gọn gàng hơn, chuyển đổi các mockup HTML thành cây ReactElement. JSX không chỉ giúp ngăn chặn việc chèn mã mà còn giúp ứng dụng chạy nhanh hơn.
2. **Kiến trúc dựa trên thành phần (component):** Một trong những lợi thế chính của React là thiết kế mô-đun của nó. React sử dụng kiến trúc dựa trên thành phần cho phép tạo các thành phần có thể tái sử dụng cho giao diện người dùng. Điều này giúp dễ dàng duy trì và mở rộng quy mô ứng dụng.
3. **Kết xuất (rendering) nhanh hơn:** Virtual DOM của thư viện React hướng lưu lượng truy cập và yêu cầu hiệu quả hơn. Nó cung cấp tốc độ và độ chính xác cho các ứng dụng có khối lượng lớn.
4. **Phát triển ứng dụng di động:** Trong khi React chủ yếu được xem như một thư viện ứng dụng web, nó đã được nâng cấp khung hoạt động để phát triển các ứng dụng gốc di động cho cả iOS và Android.
5. **Dễ học, dễ sử dụng:** Hầu hết các lập trình viên nắm vững JavaScript có thể học React nhanh chóng.

4.2. Xây dựng back-end với Node.js

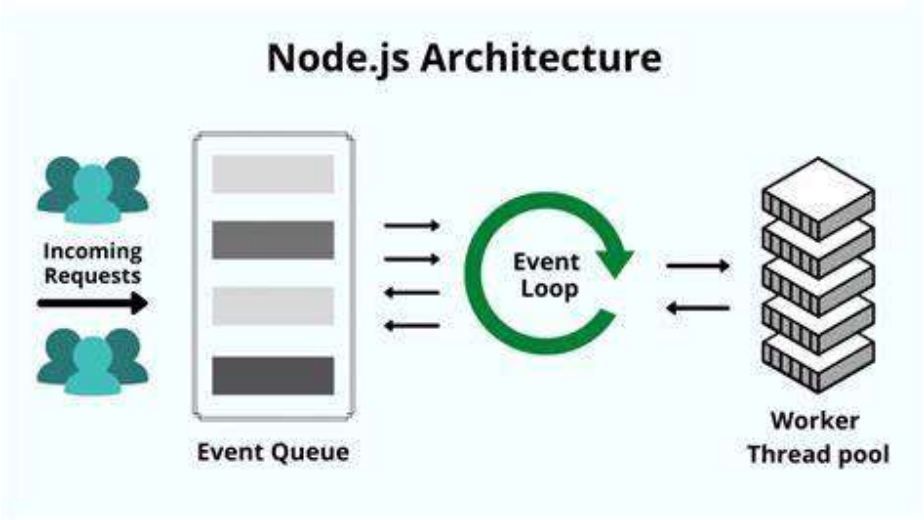
Node.js là một nền tảng được xây dựng trên V8 JavaScript Engine – trình thông dịch thực thi mã JavaScript, giúp xây dựng các ứng dụng web một cách đơn giản và dễ dàng mở rộng. Node.js được phát triển bởi Ryan Dahl vào năm 2009 và có thể chạy trên nhiều hệ điều hành khác nhau: OS X, Microsoft Windows, Linux. Theo khảo sát của Stack Overflow về các framework, nền tảng được sử dụng nhiều nhất năm 2019, NodeJS đã giành vị trí đầu tiên với số lượng người dùng lên đến gần 50%.

Ưu điểm lớn nhất của Node.js là cho phép thực hiện lập trình bất đồng bộ. Ở chế độ đồng bộ, chương trình thực thi từng dòng và tiến hành thực thi dòng tiếp theo khi dòng hiện tại đã thực thi xong. Ở chế độ bất đồng bộ, chương trình thực thi tất cả dòng code cùng một lúc. Node.js sử dụng mô hình I/O lập trình theo sự kiện, non-blocking, do đó nó khá gọn nhẹ và hiệu quả - công cụ hoàn hảo cho các ứng dụng chuyên sâu về dữ liệu theo thời gian thực chạy trên các thiết bị phân tán.

Dưới đây là các tính năng chính của Node.js:

1. **Lập trình hướng sự kiện và không đồng bộ:** Toàn bộ giao diện lập trình ứng dụng (API) trong thư viện Node.js đều không đồng bộ, hay không bị chặn (non-blocking). Về cơ bản, điều này có nghĩa là một máy chủ sử dụng Node.js sẽ không bao giờ chờ một API trả về dữ liệu. Máy chủ sẽ chuyển sang API kế tiếp sau khi gọi API đó và cơ chế thông báo của các sự kiện (events) trong Node.js giúp máy chủ nhận được phản hồi từ lần gọi API trước.

2. **Xử lý nhanh chóng:** Được xây dựng trên công cụ JavaScript V8 của Google Chrome, thư viện Node.js có khả năng xử lý mã vô cùng nhanh.
3. **Lập trình đơn luồng (single thread) nhưng có khả năng mở rộng cao:** Node.js sử dụng một mô hình luồng đơn với vòng lặp sự kiện (event). Cơ chế sự kiện cho phép máy chủ phản hồi non-blocking và cũng cho phép khả năng mở rộng cao hơn so với các máy chủ truyền thống hỗ trợ giới hạn các luồng để xử lý yêu cầu. Hình 4.1 mô tả trực quan về luồng thao tác trong Node.js.
   Node.js sử dụng một chương trình đơn luồng, cùng một chương trình có thể cung cấp dịch vụ cho một số lượng yêu cầu lớn hơn so với các máy chủ truyền thống như máy chủ HTTP Apache.
4. **Không có vùng nhớ tạm thời (buffer):** Các ứng dụng Node.js không có vùng nhớ tạm thời cho bất kỳ dữ liệu nào. Các ứng dụng này chỉ đơn giản xuất dữ liệu theo khối.



Hình 4.1: Kiến trúc tổng quan của Node.js.

4.3. Cơ sở dữ liệu MongoDB

MongoDB là một cơ sở dữ liệu hướng tài liệu (document), một dạng của NoSQL, lần đầu được giới thiệu bởi MongoDB Inc., tại thời điểm đó là thế hệ 10, vào tháng 10 năm 2007. Nó là một phần của sản phẩm PaaS (Platform as a Service) tương tự như Windows Azure và Google App Engine, sau đó đã được chuyển thành nguồn mở từ năm 2009.

Mô hình lưu trữ dữ liệu của MongoDB rất đơn giản để các nhà phát triển tìm hiểu và sử dụng, trong khi vẫn cung cấp tất cả các khả năng cần thiết để đáp ứng các yêu cầu phức tạp nhất ở mọi quy mô. Dưới đây là một số điểm nổi bật:

1. MongoDB lưu trữ dữ liệu trong các tài liệu linh hoạt, giống như JSON, có nghĩa là các trường có thể thay đổi từ tài liệu này sang tài liệu khác và cấu trúc dữ liệu có thể được thay đổi theo thời gian.
2. Mô hình dữ liệu tương đồng với các đối tượng dữ liệu trong quá trình lập trình, giúp dữ liệu dễ dàng làm việc với các dữ liệu.
3. Truy vấn đặc biệt, lập chỉ mục và tổng hợp thời gian thực cung cấp các cách mạnh mẽ để truy cập và phân tích dữ liệu.
4. MongoDB là một cơ sở dữ liệu phân tán ở nhân của nó, vì vậy tính sẵn sàng cao, khả năng mở rộng cao (theo chiều ngang), có thể mổ rộng mô hình lưu trữ bằng cách thêm các máy chủ khác nhau ở nhiều địa điểm mà không phải nâng cấp phần cứng.
5. MongoDB là miễn phí để sử dụng.

Trong dự án này, chúng em tiến hành triển khai hệ thống trên nền tảng web, sử dụng các công nghệ trên để có thể xây dựng được một trang web đơn giản.

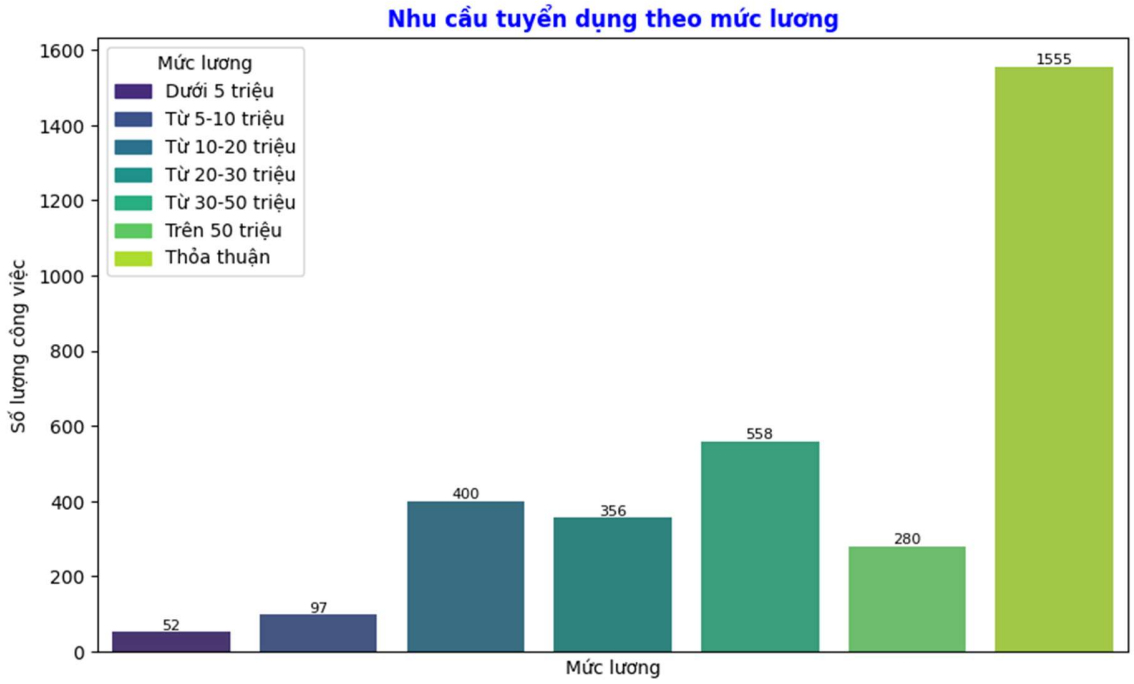
CHƯƠNG 5. KẾT QUẢ THỰC NGHIỆM

Chương này sẽ trình bày chi tiết việc hiểu dữ liệu thông qua các công cụ trực quan hóa. Các kết quả thực nghiệm của phương pháp mô hình hóa chủ đề và kết quả triển khai hệ thống cũng được trình bày và phân tích ở chương này.

5.1. Trực quan hóa dữ liệu

Dưới đây là một số kết quả đạt được trong quá trình tiến hành trực quan hóa dữ liệu nhằm có cái nhìn tổng quan về thị trường tuyển dụng IT.

Biểu đồ biểu diễn nhu cầu tuyển dụng theo mức lương



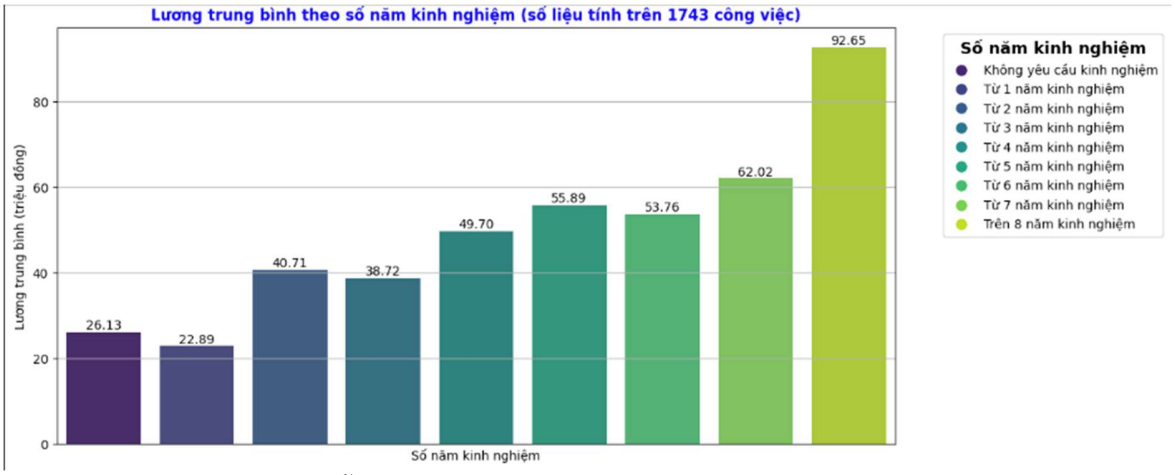
Hình 5.1: Biểu đồ biểu diễn nhu cầu tuyển dụng theo mức lương

Từ dữ liệu trên, ta có một vài nhận xét:

- Phần đông chiếm đa số là các công việc có mức lương từ 10 triệu trở lên, so với mức sống ở Việt Nam hiện nay thì mức lương này là khá ổn. Những người làm về lĩnh vực IT có thể lạc quan về mức lương của ngành ít nhất sẽ đủ để trang trải cuộc sống.
- Chiếm nhiều nhất là các công việc với mức lương từ 30 đến 50 triệu, với 558 công việc (chiếm 16,91%). Các mức lương từ 10-20 triệu, từ 20-30 triệu và trên 50 triệu không có biến động nhiều so với mức lương từ 30-50 triệu.
- Mức lương dưới 5 triệu là ít nhất, chỉ chiếm 1,57% trong tổng số công việc.

Có tới 1555 công việc với lương thỏa thuận, cho thấy có sự linh hoạt của nhà tuyển dụng trong việc đàm phán với ứng viên. Trong việc đàm phán, nhà tuyển dụng sẽ không đưa ra mức lương cố định cụ thể mà tùy thuộc vào kỹ năng, kinh nghiệm và đặc điểm cụ thể của từng ứng viên mà nhà tuyển dụng sẽ đưa ra mức lương phù hợp.

Biểu đồ biểu diễn mối quan hệ giữa lương trung bình theo số năm kinh nghiệm



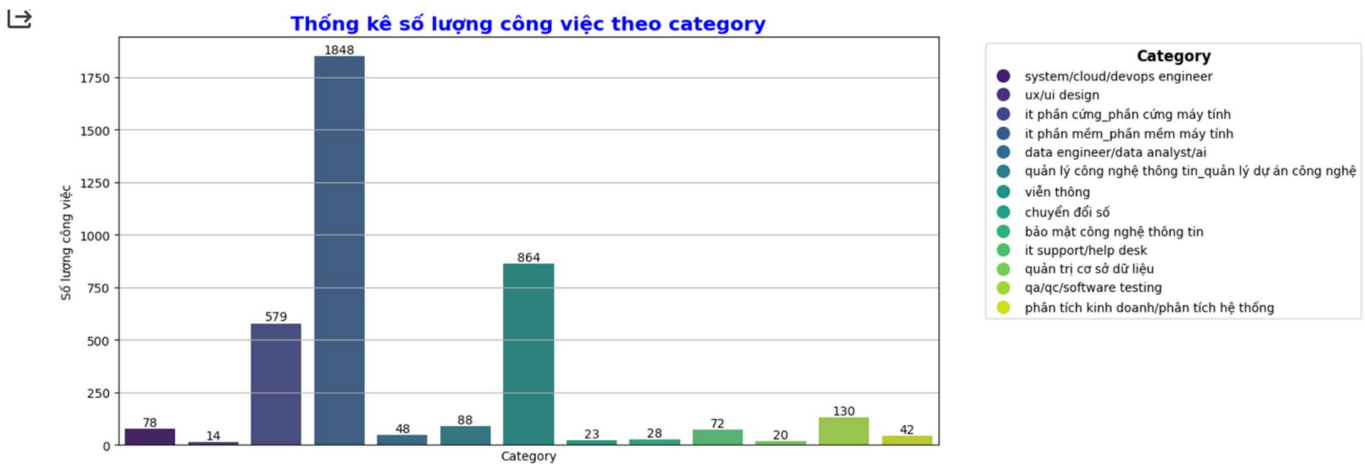
Hình 5.2: Biểu đồ biểu diễn mối quan hệ giữa lương trung bình theo số năm kinh nghiệm

Biểu đồ cột trên thực hiện tính toán trên dữ liệu của 1,743 công việc có dữ liệu về lương. Từ dữ liệu trên, ta có một vài nhận xét:

- Các công việc tuyển dụng về công nghệ thông tin đều có mức lương trung bình trên 22 triệu.
- Mức lương trung bình thấp nhất rơi vào mức từ 1 năm kinh nghiệm (khoảng 22.89 triệu đồng), cao nhất là trên 8 năm kinh nghiệm với mức lương trung bình trên 90 triệu đồng (92.65 triệu).
- Lương trung bình **tỉ lệ thuận** theo số năm kinh nghiệm cần thiết. Điều này là hợp lẽ thường, bởi người có "thâm niên" cao thường được các công ty săn đón nhiều hơn cũng như sẵn sàng trả lương cao hơn.

Tuy nhiên, cũng phải nhìn nhận thực tế rằng, do tập dữ liệu với số mẫu bài đăng tuyển về công việc IT mà chúng em thu thập được còn khá khiêm tốn, nên có một số dữ liệu có vẻ "không đúng" với thực tế, ví dụ so sánh giữa "không yêu cầu kinh nghiệm" và "một năm kinh nghiệm" thì mức lương trung bình của "không yêu cầu kinh nghiệm" lại cao hơn, một phần nguyên nhân do số lượng công việc không yêu cầu kinh nghiệm có số lượng nhiều hơn rất nhiều so với "một năm kinh nghiệm", một phần khác do có một số công việc không ghi yêu cầu về kinh nghiệm trong bài đăng tuyển nhưng thực tế thì lại có yêu cầu khiến số liệu về lương không phù hợp với yêu cầu kinh nghiệm. Những vấn đề này có thể được cải thiện bằng việc tìm thêm nhiều nguồn để thu thập dữ liệu nhằm tăng số lượng mẫu có trong tập dữ liệu.

Biểu đồ cột thống kê số lượng công việc theo nhãn phân loại (category)

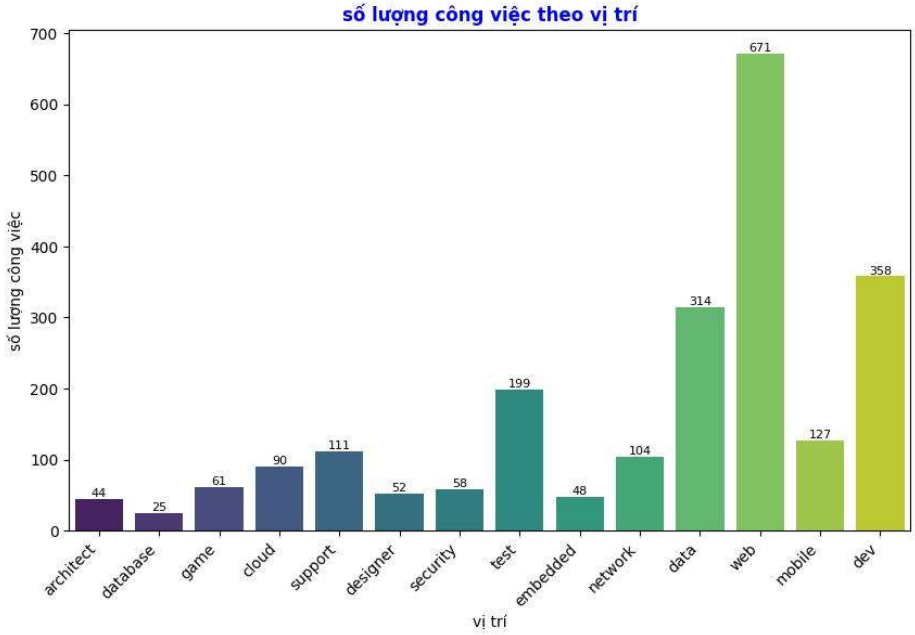


Hình 5.3: Biểu đồ cột thống kê số lượng công việc theo nhãn phân loại (category)

Từ biểu đồ trên, ta có thể thấy công việc IT phân bố không đồng đều theo trường *category*, cụ thể:

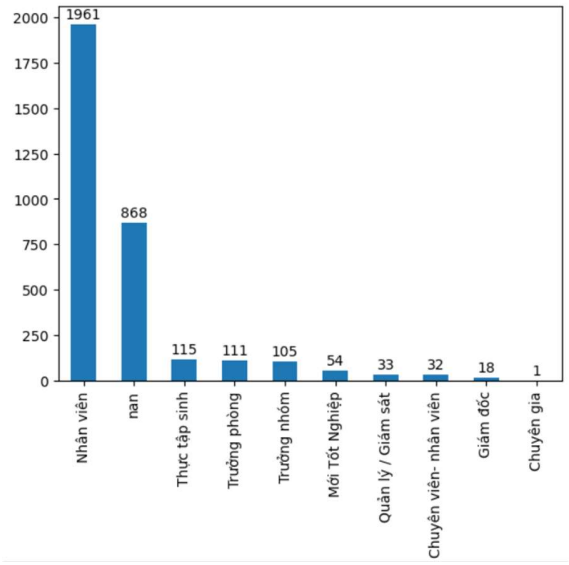
- Các công việc về Phần mềm máy tính chiếm đa số (1848 công việc), gấp đôi các công việc về Viễn thông (864) và gấp hơn ba lần số công việc liên quan đến Phần cứng máy tính (579).
- Sự mất cân bằng cho thấy một thực tế rằng, thị trường tuyển dụng hiện nay đang dành sự quan tâm rất lớn cho ngành công nghệ thông tin làm về phần mềm, đây có thể là một tín hiệu vui cho các bạn sinh viên chuyên ngành Khoa học máy tính của Đại học Bách khoa Hà Nội khi mà công việc liên quan được tuyển khá nhiều, giảm thiểu nỗi lo ra trường không có việc làm.

Cụ thể hơn, chúng ta xem xét cụ thể số lượng công việc theo vị trí/mảng cụ thể như biểu đồ dưới đây. Số lượng công việc dành cho mảng lập trình web chiếm đa số (671 công việc), chiếm tới một phần năm số lượng công việc đăng tuyển, cho thấy các công ty đang rất quan tâm đến web và lượng công việc về lập trình web theo chúng em quan sát các năm gần đây rất nhiều. Ngoài ra, mảng dữ liệu (data) và lập trình phần mềm cũng dành được sự quan tâm từ doanh nghiệp khi lượng công việc đăng tuyển cho hai mảng này khá nhiều (lần lượt là 314 và 358 công việc).



Hình 5.4: Biểu đồ cột thống kê số lượng công việc theo vị trí cụ thể: mobile, test, dev,…

Biểu đồ cột thống kê số lượng công việc theo cấp độ (level)



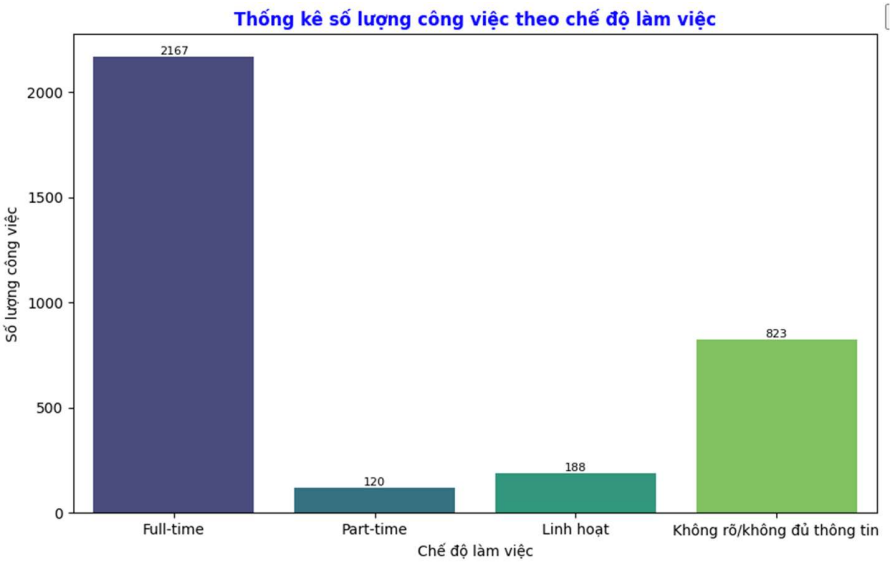
Hình 5.5: Biểu đồ cột thống kê số lượng công việc theo cấp độ (level)

Dù đã tiền xử lý, thực hiện các thao tác nhằm khôi phục giá trị thiếu, nhưng vẫn có 868 dữ liệu thiếu trong trường thuộc tính này (NaN), một vài nguyên nhân có thể kể đến như sau: i) do trong đơn tuyển dụng không ghi rõ vị trí ứng tuyển, ii) với ngữ cảnh được nêu trong phần mô tả công việc và các trường khác thì có thể tự suy ra được vị trí ứng tuyển là gì nên các công ty không ghi thêm vị trí ứng tuyển vào đơn tuyển dụng.

Từ biểu đồ trên, ta có một vài nhận xét:

- Vị trí ứng tuyển được đăng nhiều nhất là "Nhân viên" với 1961 đơn tuyển trên 3298 đơn, chiếm 59,46%, tiếp theo là Thực tập sinh và Trưởng phòng (chiếm khoảng 3,45%).
- Số lượng công việc ở các cấp độ cao hơn, như quản lý/giám sát và giám đốc, chỉ chiếm một tỷ lệ nhỏ.

Biểu đồ cột thống kê số lượng công việc theo chế độ làm việc

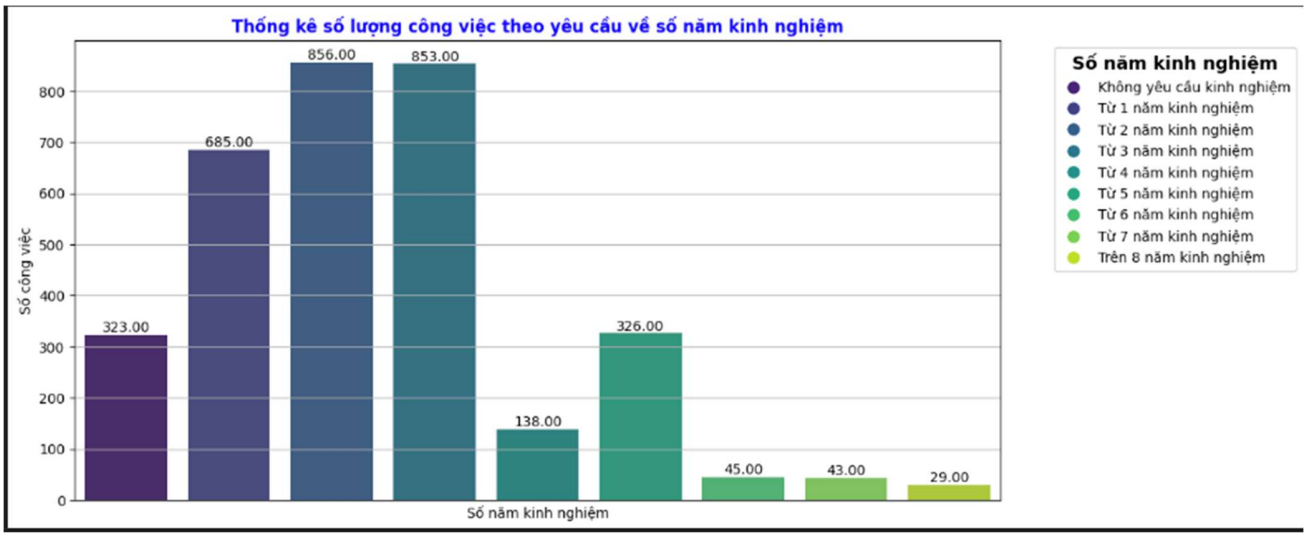


Hình 5.6: Biểu đồ cột thống kê số lượng công việc theo chế độ làm việc

Theo biểu đồ trên, số lượng công việc full-time là cao nhất, với 2167 công việc, chiếm hơn 90% tổng số lượng công việc. Chế độ part-time và linh hoạt có số lượng công việc thấp hơn, với lần lượt 120 (chiếm khoảng 6%) và 188 (chiếm khoảng 8%) công việc.

Dựa vào biểu đồ trên, có thể thấy: Các công ty có xu hướng tuyển dụng nhân viên full-time về làm lâu dài. Điều này là dễ hiểu vì một số nguyên nhân sau: i) nhân viên làm toàn thời gian mang lại sự ổn định và nhất quán trong công việc họ thực hiện, ii) nhân viên toàn thời gian thường có sẵn để làm việc theo lịch trình công ty và đáp ứng các yêu cầu công việc một cách liên tục và iii) do các quy định pháp lý liên quan đến nhân sự, như các quy định về bảo hiểm xã hội,…

Biểu đồ cột thống kê số lượng công việc theo yêu cầu về số năm kinh nghiệm

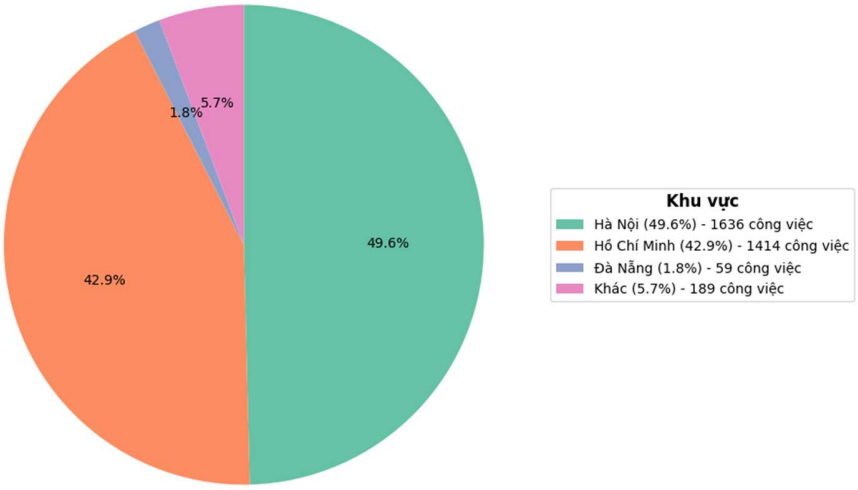


Hình 5.7: Biểu đồ cột thống kê số lượng công việc theo yêu cầu về số năm kinh nghiệm

Theo biểu đồ trên, số lượng công việc yêu cầu 2 năm kinh nghiệm và 3 năm kinh nghiệm chiếm tỉ lệ cao nhất (mỗi yêu cầu công việc chiếm gần 26%). Số lượng công việc yêu cầu 1 năm kinh nghiệm cũng ở mức khá cao (chiếm 20.77%). Tăng dần về số năm kinh nghiệm, số công việc yêu cầu giảm xuống ở mức thấp.

Dựa vào biểu đồ trên, có thể thấy: Các công ty không khắt khe trong việc cần ứng viên có kinh nghiệm trước khi đi làm, và có xu hướng tuyển dụng nhân viên trẻ (số năm kinh nghiệm thấp). Điều này là hợp thực tế, vì nhân lực trẻ, năng động thì hiệu suất làm việc sẽ cao hơn dù chưa có nhiều kinh nghiệm, việc có ít kinh nghiệm giúp nhà tuyển dụng có thể dễ dàng đào tạo hơn so với những người già dặn kinh nghiệm. Mặt khác, những người giàu thâm niên thì thường các công ty sẽ tuyển dụng với vị trí cao hơn như quản lý, trưởng phòng,… nên số lượng việc làm ít cũng là điều dễ hiểu.

Biểu đồ phần trăm mô tả phân bố số lượng công việc theo khu vực

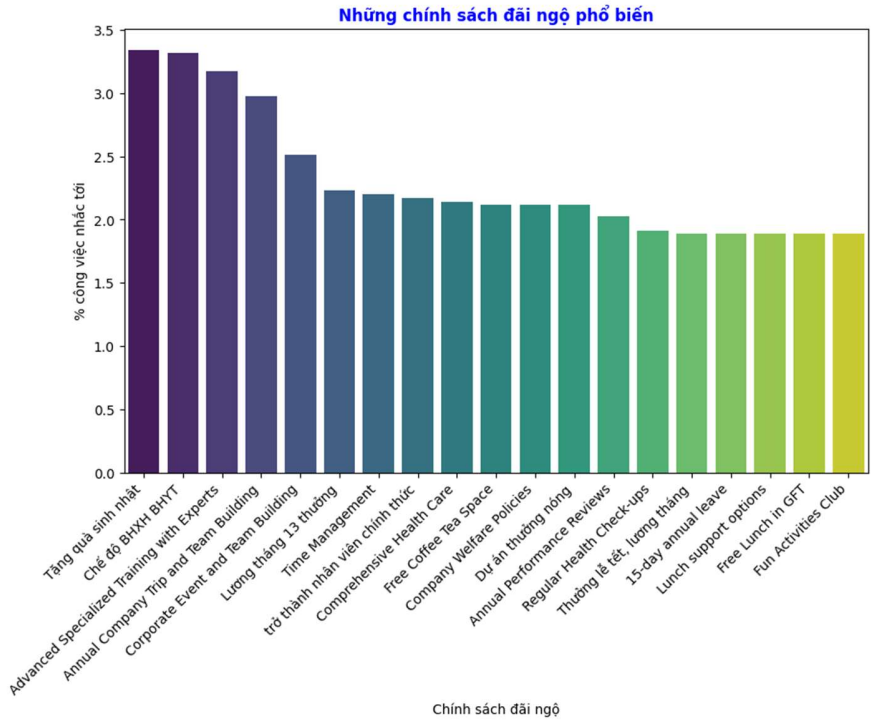


Hình 5.8: Biểu đồ phần trăm mô tả phân bố số lượng công việc theo khu vực

Theo biểu đồ trên, công việc chủ yếu phân bố ở hai thành phố lớn là Hà Nội (chiếm 49,6%) và Thành phố Hồ Chí Minh (chiếm 42,9%). Đứng thứ ba là Đà Nẵng, tuy nhiên tỉ lệ khá ít, chỉ chiếm 1,8% so với tổng số công việc đăng tuyển. Phần "Khác" bao gồm các tỉnh, thành phố còn lại, nước ngoài và không rõ địa điểm, chiếm 5,7%.

Dựa vào biểu đồ trên, có thể thấy: Các công ty công nghệ thông tin có xu hướng tập trung ở hai thành phố lớn là Hà Nội và TP. HCM. Một số nguyên nhân trong thực tế đã chứng minh cho điều này: i) Các thành phố lớn thường có hạ tầng và tiện ích tốt, bao gồm cả mạng lưới giao thông, truyền thông và các dịch vụ công cộng, điều này giúp đảm bảo việc kết nối dễ dàng với các đối tác kinh doanh, khách hàng và nguồn nhân lực, ii) Các thành phố lớn thu hút nguồn nhân lực chất lượng cao từ các trường đại học và các tổ chức đào tạo chuyên ngành, iii) Các thành phố lớn thường có sẵn một mạng lưới doanh nghiệp và cộng đồng kỹ thuật đa dạng và phong phú.

Biểu đồ cột thống kê về những chính sách đãi ngộ phổ biến



Hình 5.9: Biểu đồ cột thống kê về những chính sách đãi ngộ phổ biến

Có một sự khó khăn nhất định trong việc phân tích và hiểu dữ liệu khi trong bản tin đăng tuyển có cả Tiếng Việt và ngôn ngữ Anh. Ví dụ như trong trường hợp về chính sách đãi ngộ trên đây, một số đãi ngộ viết bằng ngôn ngữ Anh nên có thể bị trùng với Tiếng Việt. Ta có một số phân tích về biểu đồ trên như sau:

- Các chế độ đãi ngộ được nhắc đến nhiều nhất là: Tặng quà sinh nhật chiếm khoảng 3.3%, Chế độ BHXH BHYT (bảo hiểm xã hội, bảo hiểm y tế) chiếm khoảng 3.25%.
- Tất cả những chế độ đãi ngộ phổ biến được thống kê trên đây là hợp lý trên thực tế và phù hợp với người lao động, vừa mang lợi ích vật chất, vừa mang yếu tố tinh thần như chính sách BHXH, BHYT, lương thưởng tháng 13, thưởng lễ Tết, các hoạt động giải trí, đi du lịch nhằm kết nối nhân viên,…

5.2. Sử dụng phương pháp mô hình hoá chủ đề và đếm tần suất từ để phân tích các lĩnh vực con

Kết quả trực quan hóa dữ liệu là một phần để giúp chúng ta hiểu dữ liệu, phát hiện xu hướng và tình hình thị trường tuyển dụng hiện nay. Ngoài ra, để giúp người dùng biết và nắm được những yêu cầu và kỹ năng cần thiết cho công việc mình dự định ứng tuyển, chúng em sử dụng phương pháp mô hình hóa chủ đề áp dụng lên bộ dữ liệu thu thập được.

Dưới đây là một số kết quả đạt được trong quá trình phân tích các lĩnh vực con sử dụng phương pháp mô hình hóa chủ đề được đề xuất.

Bảng 5.1: Một số kết quả trong việc phân tích lĩnh vực con dựa trên phương pháp mô hình hóa chủ đề.

Vị trí	Kĩ năng phổ biến	Những yêu cầu phổ biến
architect	"Software Architect", "Java", "Cloud", "AWS", "Python", "DevOps", "C++", "Aws", "Solution Architecture", "Golang",	"System application performance tuning", "Infrastructure Architecture and Design", "Experienced IT Team Management", "Effective Collaboration Skills", "English proficiency in client presentations", "Agile Methodologies with Kanban", "OOP Design Patterns", "Java EE Server Architectures", "Financial Banking Experience", "Recognition of Contributions", "SQL Server and Database",
cloud	"DevOps", "Cloud", "AWS", "Linux", "Python", "Azure", "Java", "SQL", "Networking", "CI/CD",	"Security Best Practices and Ethical Hacking", "DevOps Methodology Fundamentals", "Windows and Linux Operating Systems", "Virtualization and Hypervisors", "Strong Troubleshooting Skills", "Experienced CI/CD with Jenkins Git", "Microservices Architecture Experience", "Configuration Management Automation Tools", "Networking and Routing Protocols", "AWS SysOps Architect Certification", "CICD Tools Deployment", "Database Technology Experience",
data	"Python", "SQL", "Business Analyst",	"Analytical problem solving skills", "Software Development Process", "Python Programming Skills",

Vị trí	Kĩ năng phổ biến	Những yêu cầu phổ biến
	"Agile", "Data Analyst", "Java", "Database", "Business Analysis", "ERP", "Ruby on Rails",	"Thành thạo tin học văn phòng", "Advanced Data Analytics with SQL and Python", "Tốt nghiệp liên quan tài chính", "\"Machine Learning Frameworks\"", "Database Design and Optimization", "Data Visualization Tools", "Cấu trúc dữ liệu và giải thuật", "Optimization of ETL and Data Pipelines", "Business Analyst Experience",
database	"Database", "Oracle", "SQL", "MySQL", "Linux", "Agile", "Python", "Data Analyst", "DBA Oracle", "MS SQL DBA",	"Implementation of CRM and ERP", "Knowledge of Database Design", "Bachelor's in Computer Science", "Financial Banking Experience", "Strong Negotiation Skills", "Performance Optimization for MySQL and PostgreSQL", "Quản lý hướng dẫn và khả năng", "Data analysis and collection", "Experience with Oracle and MySQL", "Network Systems Certifications", "Windows and Linux Operating Systems", "Highly Detail-Oriented Assistant",
designer	"UIUX", "Designer", "User Experience", "Prototyping", "Visual Design", "Photoshop", "Illustrator", "Figma", "UI Design", "HTML", "CSS",	"Wordpress Plugin Mastery", "Network System Knowledge", "User Process Support and Assistance", "Mobile Responsive Development", "\"Passionate UI/UX Designer\"", "Web application development framework", "Software Securities Advantage", "Active Creative Thinking", "Camera Entertainment Activities", "Computer Hardware Experience", "Understanding Design Patterns", "Equivalent experience required",
embedded	"Embedded", "C++", "Linux", "AUTOSAR", "Python", "Automotive", "Hardware", "QT Framework", "C Language", "Docker",	"Regular Training for Professional Development", "Software Development Life Cycle (SDLC) Understanding", "Honesty and Responsibility", "Python Programming Skills", "English Skills", "Automotive Quality Evaluation", "IoT Wireless Stacks", "Autosar and Autocad BIM",
web	"JavaScript", "ReactJS", "Java", ".NET",	"\"Net Core Experience\"", "Backend Development Experience", "Experienced JavaScript Developer", "Database Design and Optimization",

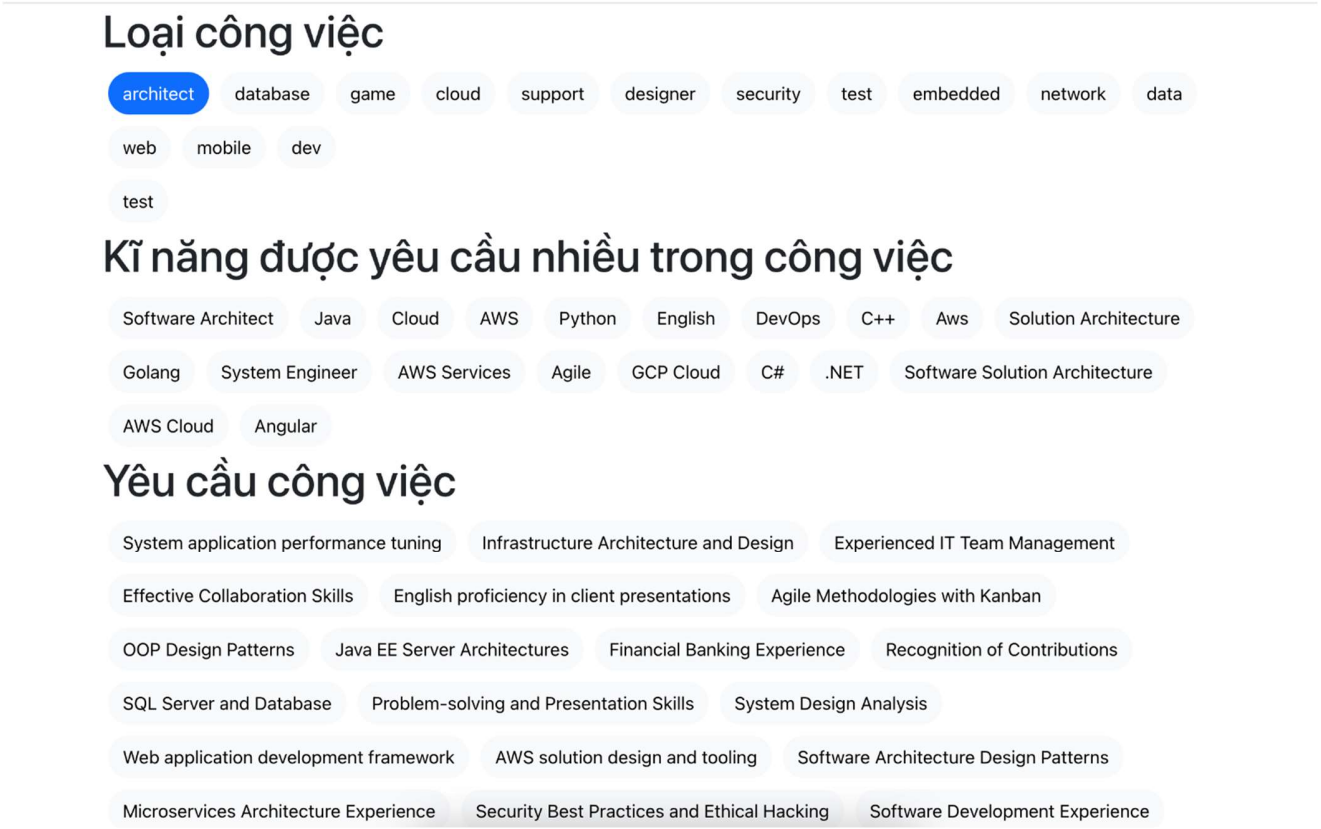
Vị trí	Kĩ năng phổ biến	Những yêu cầu phổ biến
	"NodeJS", "PHP", "CSS", "Python", "MySQL", "SQL",	"Web Development Experience", "\"Java Spring Framework Experience\"", "Frontend Development with Typescript React Vuejs", "Unit Test Experience", "Software Development Experience", "Experienced with git", "Realistic ASPNET MVC Experience",

5.3. Kết quả triển khai hệ thống

Chúng em triển khai hệ thống trên nền tảng web, sử dụng React làm front-end, Node.js làm back-end và lưu trữ dữ liệu trên cơ sở dữ liệu MongoDB.

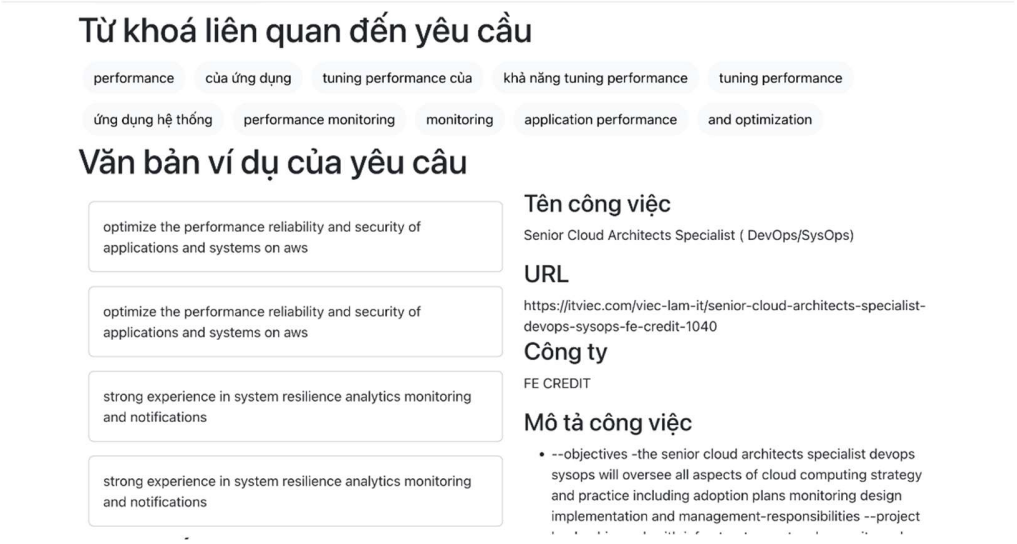
Các tính năng chúng em xây dựng:

1. Bằng việc người dùng lựa chọn một nhãn trong "Loại công việc" tương ứng với ngành nghề đang quan tâm, hệ thống sẽ trả về danh sách các kỹ năng được yêu cầu nhiều trong công việc theo thứ tự giảm dần về mức độ phổ biến.



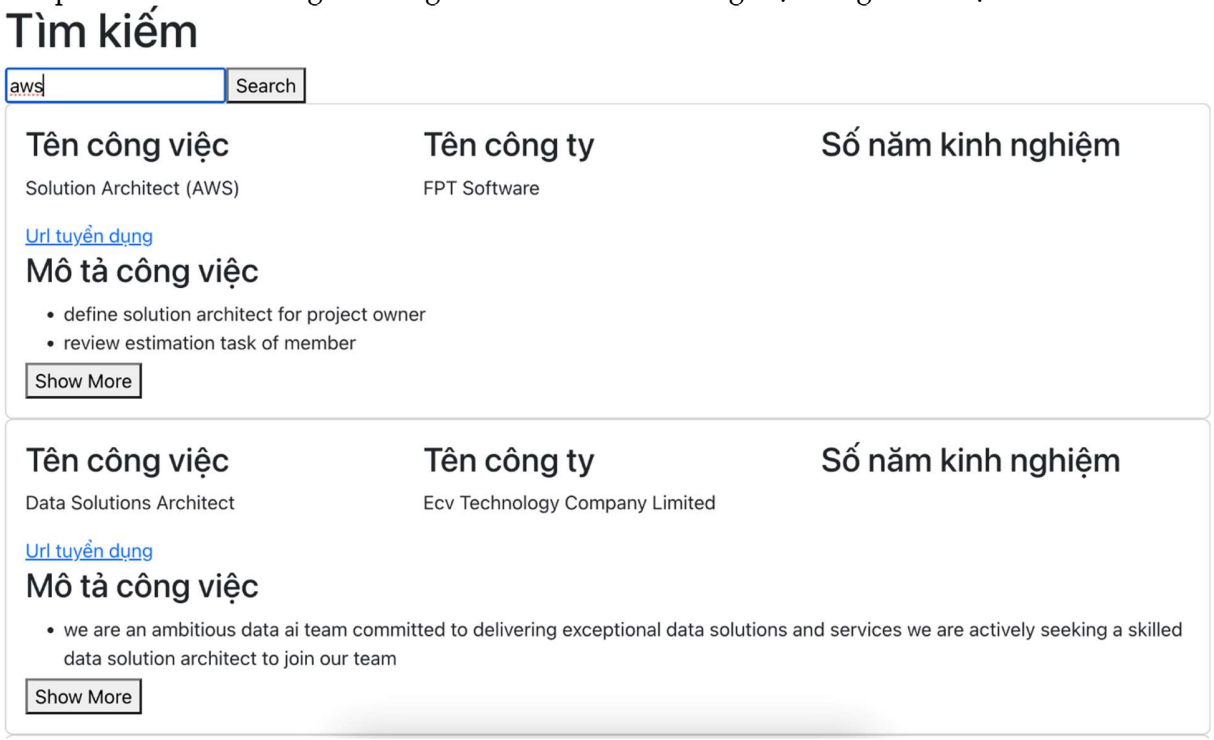
Hình 5.10: Giao diện hiển thị các kỹ năng và yêu cầu công việc phổ biến cho một công việc cho trước.

2. Bằng việc người dùng lựa chọn vào một yêu cầu công việc được hiển thị, hệ thống sẽ trả về danh sách các từ khóa và văn bản liên quan đến yêu cầu. Người dùng có thể tương tác với văn bản ví dụ để xem cụ thể các công việc có liên quan đến yêu cầu, bao gồm các thông tin như một bài đăng tuyển dụng.



Hình 5.11: Giao diện hiển thị các từ khóa và văn bản có liên quan đến yêu cầu được chọn. Người dùng có thể tương tác với văn bản ví dụ để xem chi tiết công việc có liên quan.

3. Bằng việc người dùng nhập vào một từ khóa để tìm kiếm, hệ thống sẽ trả về danh sách các công việc có liên quan đến từ khóa. Người dùng có thể xem chi tiết công việc bằng cách chọn **Show More.**



Hình 5.12: Giao diện cho chức năng tìm kiếm công việc theo từ khóa.

CHƯƠNG 6. KẾT LUẬN

6.1. Kết luận

Qua khảo sát thực trạng và với mong muốn những người đang tìm việc, nhất là sinh viên ngành Công nghệ thông tin có cái nhìn tổng quan về thị trường tuyển dụng và nắm được những kỹ năng, yêu cầu của công việc mình đang chuẩn bị ứng tuyển, chúng em đã thực hiện đề tài "*Phân tích thị trường tuyển dụng trong lĩnh vực Công nghệ thông tin*". Quá trình tìm hiểu thực trạng cũng như những bước đầu trong việc thu thập và tiền xử lý dữ liệu, tích hợp dữ liệu, trực quan hóa dữ liệu và sử dụng một số kỹ thuật để phân tích dữ liệu đã được trình bày cụ thể trong báo cáo này.

Nội dung đã đạt được

Dự án của chúng em đã đáp ứng được mục tiêu đã đặt ra. Trước hết, chúng em đã tiến hành khảo sát và phân tích hiện trạng về những khó khăn, thách thức trong quá trình đi tìm một công việc của những người học ngành Công nghệ thông tin, từ đó đề ra mục tiêu cần giải quyết và rút ra những yêu cầu quan trọng về tập dữ liệu. Tiếp theo, chúng em sử dụng các công cụ, thư viện có sẵn để thu thập dữ liệu là các bài đăng tuyển dụng trên các nguồn trang web khác nhau. Sau khi có dữ liệu, tiến hành các thao tác tiền xử lý dữ liệu: làm sạch, khôi phục và điền giá trị thiếu, tích hợp dữ liệu từ các nguồn. Sau đó, tiến hành sử dụng các thư viện để trực quan hóa dữ liệu, phục vụ cho nhu cầu hiểu dữ liệu. Đồng thời, chúng em cũng sử dụng một kỹ thuật, gọi là kỹ thuật mô hình hóa chủ đề, để đưa ra các yêu cầu và kỹ năng cần thiết cho một công việc cụ thể.

Việc thực hiện bài tập lớn đã giúp chúng em vận dụng được những kiến thức đã học trong học phần vào thực tế, cho chúng em cái nhìn cụ thể hơn về quy trình giải quyết một vấn đề trong khoa học dữ liệu, hiểu được tầm quan trọng của các bước trong luồng nghiệp vụ mà trước đây chỉ được nghe trong lý thuyết.
Bên cạnh đó, việc thực hiện bài tập lớn cũng giúp chúng em phát triển thêm các kỹ năng mềm, bao gồm kỹ năng phân chia thời gian, lập kế hoạch cụ thể cho từng công việc, kỹ năng tìm kiếm, nghiên cứu tài liệu và viết báo cáo. Những kiến thức trên là vô cùng quan trọng và quý báu, giúp chúng em hoàn thiện bản thân hơn và phát triển công việc sau này.

Những khó khăn và hạn chế trong quá trình thực hiện bài tập lớn

Trong quá trình thực hiện bài tập lớn, chúng em gặp phải một số khó khăn nhất định.

Khó khăn lớn nhất mà chúng em gặp phải là về nguồn dữ liệu và cách thu thập dữ liệu. Các trang web mà chúng em lựa chọn tuy được nhiều người dùng truy cập, nhưng các trang web này hạn chế người khác cào dữ liệu. Các thông tin cung cấp không được đầy đủ. Ví dụ, đối với trang web https://itviec.com, nếu muốn biết được thông tin về lương thì yêu cầu phải đăng nhập (khắc phục bằng cách sử dụng FormRequest của Scrapy), hay có trang web thì một số thông tin lại hiển thị trong mục pop-up (khắc phục bằng cách sử dụng giả lập tương tác của Selenium). Mặt khác, lượng đơn tuyển dụng trên những trang web này lại khá khiêm tốn, có rất nhiều trường thuộc tính bị rỗng hoặc ẩn, gây ảnh hưởng không nhỏ đến việc tiền xử lý và tích hợp dữ liệu, trực quan hóa và kết quả phân tích công việc.

Một khó khăn khác mà chúng em gặp phải là việc phân tích chủ đề, tìm từ khoá vẫn còn thủ công khi kết quả vẫn cần phải lọc bởi người dùng, mà điều này có thể xuất phát từ vấn đề của tập dữ liệu.

Hạn chế lớn nhất trong quá trình thực hiện bài tập lớn là kiến thức của chúng em còn hiểu chưa sâu, một số phần còn đang bị nhầm lẫn, gây mất thời gian.

6.2. Hướng phát triển trong tương lai

Mặc dù chúng em đã đạt được mục tiêu đề ra, nhưng vẫn còn một số hạn chế. Trong tương lai, có thể khắc phục vấn đề về tập dữ liệu bằng cách tìm một số nguồn có nhiều dữ liệu về công việc ngành Công nghệ thông tin hơn. Ngoài ra, có thể sử dụng một số kỹ thuật tối ưu hơn, giúp tăng hiệu suất cho việc đưa ra yêu cầu và kỹ năng cần có của công việc.

Cuối cùng, có thể đi đến xây dựng một ứng dụng hay một trang web có nhiều tính năng được tích hợp, ngoài việc cho biết một số thống kê, biểu đồ cố định (định nghĩa bởi người lập trình), thì cho phép người dùng tự định nghĩa yêu cầu để đưa ra biểu đồ phân tích phù hợp. Tích hợp thêm một số mô hình ngôn ngữ lớn hiện nay như GPT-4, LaMDA,… có thể cải thiện tác vụ đưa ra yêu cầu, kỹ năng cần có để có thể ứng tuyển vào một công việc nào đó.

Trên đây là toàn bộ báo cáo Khoa học dữ liệu với đề tài "*Phân tích thị trường tuyển dụng trong lĩnh vực Công nghệ thông tin*". Với thời gian có hạn, nguồn lực và kiến thức còn hạn chế nên chắc chắn chúng em không tránh khỏi sai sót. Chúng em rất mong nhận được ý kiến đóng góp từ thầy và các thầy cô trong nhóm chuyên môn để giúp bản báo cáo của chúng em được hoàn thiện hơn.

TÀI LIỆU THAM KHẢO

[1] Grootendorst, M.R. (2022). BERTopic: Neural topic modeling with a class-based TF-IDF procedure. ArXiv, abs/2203.05794.