ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI OF UNIVERSITY OF SCIENCE AND TECHNOLOGY

VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC
SCHOOL OF APPLIED MATHEMATICS AND INFORMATICS

Các mô hình ngẫu nhiên và ứng dụng – MI5040

Hệ thống khuyến nghị dựa trên NLP trong lĩnh vực tuyển dụng.

Sinh viên: *Nguyễn Văn Nghiêm* - *20206206*
Nguyễn Hoàng Sơn - *20206165*
Tạ Duy Hải - *20206197*

Giảng viên: *TS. Nguyễn Thị Ngọc Anh*

Viện: *Toán ứng dụng và Tin học*
Đại học Bách khoa Hà Nội

Hà Nội, Ngày 06 tháng 08 năm 2023

Lời nói đầu

Quá trình tuyển dụng nhân sự là một phần quan trọng trong hoạt động của bất kỳ doanh nghiệp nào. Tìm kiếm và lựa chọn nhân tài phù hợp không chỉ quan trọng mà còn ảnh hưởng trực tiếp đến hiệu suất và thành công của tổ chức. Tuy nhiên, với số lượng lớn hồ sơ ứng viên và yêu cầu chính xác của từng vị trí, quá trình tuyển dụng có thể trở nên phức tạp và tốn nhiều thời gian.

Chính vì vậy, hệ thống khuyến nghị cho lĩnh vực tuyển dụng ra đời như một giải pháp hiệu quả. Được xây dựng dựa trên NLP - công nghệ xử lý ngôn ngữ tự nhiên, hệ thống này có khả năng tự động phân tích và hiểu ngôn ngữ tự nhiên của các hồ sơ ứng viên. Điều này cho phép đánh giá các kỹ năng, kinh nghiệm và học vấn của ứng viên một cách chính xác và nhanh chóng.

Trong báo cáo này, nhóm sẽ đi sâu vào nghiên cứu về cách xây dựng hệ thống khuyến nghị dựa trên các kỹ thuật NLP. Bài báo trình bày ba nội dung chính như sau:

1. **Chương 1 - Tổng quan đề tài** : Chương 1 sẽ nêu ra tổng quan bài toán gợi ý nhân sự và một số nghiên cứu liên quan.

2. **Chương 2 - Phương pháp tiếp cận đề tài**: Chương 2 sẽ đi vào chi tiết về hệ gợi ý và một số phương pháp được đề xuất giải quyết bài toán dựa vào các mô hình xử lý ngôn ngữ tự nhiên.

3. **Chương 3 - Ứng dụng các kỹ thuật NLP cho hệ thống khuyến nghị trong lĩnh vực tuyển dụng**: Trong chương này, ta sẽ vận dụng các kiến thức đã được trình bày để ứng dụng xây dựng mô hình trong thực tiễn.

Chúng em xin gửi lời cảm ơn sâu sắc đến TS. Nguyễn Thị Ngọc Anh đã truyền đạt cho chúng em những kiến thức nền tảng để hoàn thành bài báo cáo này. Báo cáo không tránh khỏi những thiếu sót, chúng em kính mong nhận được sự nhận xét, góp ý của cô để báo cáo được hoàn thiện hơn.

Mục lục

1 Tổng quan đề tài.

1.1 Giới thiệu bài toán

Bài toán gợi ý nhân sự là một trong những lĩnh vực phát triển nổi bật trong quản lý nguồn nhân lực. Với sự phát triển của công nghệ và ứng dụng trí tuệ nhân tạo (AI), các công ty và tổ chức ngày càng quan tâm đến việc tối ưu hóa quy trình tuyển dụng.

Hệ gợi ý trong nhân sự là một hệ thống thông minh dựa được xây dựng để giúp tổ chức trong quá trình tuyển dụng. Hệ thống này sử dụng dữ liệu về ứng viên và các thông tin liên quan khác để đưa ra các gợi ý thông minh và chính xác.

Trong quá trình tuyển dụng, hệ gợi ý có khả năng phân tích và đánh giá hồ sơ ứng viên, từ đó đề xuất danh sách ứng viên phù hợp với các vị trí công việc cụ thể. Hệ thống cũng có thể phân loại các ứng viên dựa trên các tiêu chí quan trọng như kỹ năng, kinh nghiệm và trình độ học vấn.

Hệ gợi ý trong nhân sự sử dụng các thuật toán học máy và xử lý ngôn ngữ tự nhiên để hiểu và phân tích dữ liệu nhân sự một cách hiệu quả. Nó cũng có thể được tích hợp vào các hệ thống quản lý tài nguyên nhân lực (HRM) hiện có, tạo ra môi trường làm việc thông minh và hiệu quả hơn.

1.2 Một số nghiên cứu về đề tài.

Với sự phát triển của internet và các môi trường truyền thông, các tin tức tuyển dụng dễ tiếp cận với mọi người, kéo theo đó là số lượng lớn hồ sơ ứng viên được gửi tới nhà tuyển dụng. Việc nắm bắt các thông tin chính của mỗi hồ sơ trở nên khó khăn và tốn nhiều thời gian công sức để sàng lọc toàn bộ lượng lớn ứng viên. Vì vậy trong những năm gần đây, hệ thống khuyến nghị trong lĩnh vực nhân sự là một đề tài hấp dẫn với các nhà nghiên cứu.

Các mô hình được công bố đều sử dụng các mô hình xử lý ngôn ngữ tự nhiên, do đặc thù các thông tin tuyển dụng và ứng viên đều ở dạng văn bản. Với nghiên cứu của Alsaif và các công sự [1], họ sử dụng một mô hình có sẵn trong SpaCy để huấn luyện NER, sau đó trích xuất các kỹ năng của hồ sơ ứng viên và hồ sơ tuyển dụng. Để đánh giá độ tương đồng, họ sử dụng độ đo Jaccard giữa tập kỹ năng của ứng viên với hồ sơ tuyển dụng, sau đó xếp hạng và đưa ra các ứng viên thích hợp. Với mô hình của nhóm nghiên cứu Shovon [2], họ xây dựng hệ thống khuyến nghị trên lĩnh vực khoa học máy tính. Sử dụng thư viện NLTK và biểu thức chính quy, họ trích xuất được kỹ năng trong các hồ sơ ứng viên, ngoài ra họ còn trích được họ tên, năm kinh nghiệm, chứng chỉ của ứng viên. Tuy nhiên khi so khớp lại chỉ dùng kỹ năng và phần mô tả công việc, bằng cách vector hóa TF-IDF và sử dụng độ đô cosine. Sau đó sử dụng thuật toán KNN để đề xuất các hồ sơ ứng viên thích hợp. Một cách làm khác của Tejaswini K và các cộng sự [3] đó là trực tiếp vector hóa TF-IDF hồ sơ ứng viên và hồ sơ tuyển dụng sau khi loại bỏ các từ dừng và các số. Sau đó sử dụng thuật toán KNN dựa trên độ đo cosine để đề xuất các hồ sơ thích hợp.

Nhìn chung, hệ thống mà các nghiên cứu trên đều chỉ sử dụng kỹ năng trong hồ sơ mà không sử dụng được các thông tin khác như số năm kinh nghiệm, hay thông tin về công việc đã từng làm. Việc sử dụng TF-IDF hay độ đo Jaccard sẽ không hiệu quả khi một số kỹ năng có nhiều cách viết. Ngoài ra việc tìm kiếm trên tập dữ liệu lớn, các kỹ năng trong hồ sơ lặp lại nhiều, việc vector hóa nhiều lần cũng khiến hệ thống hoạt động chậm.

Từ những nhược điểm trên, nhóm tác giả xây dựng một hệ thống khuyến nghị lọc dựa trên nội dung, ngoài so khớp kỹ năng mà còn sử dụng các thông tin khác như năm kinh nghiệm, vị trí làm việc, yêu cầu trình độ. Bằng thuật toán phù hợp và lưu trữ những thông tin được dùng nhiều lần, hệ thống hoạt động nhanh trên tập dữ liệu lớn.

2 Phương pháp tiếp cận đề tài

2.1 Hệ thống khuyến nghị (Recommendation System).

Hệ thống khuyến nghị là một dạng của hệ thống lọc thông tin, được sử dụng để dự đoán sở thích hay xếp hạng của người dùng cho một sản phẩm nào đó mà họ chưa xem xét tới trong quá khứ (sản phẩm ở đây có thể là bài hát, sách, quần áo,...). Một số hệ thống khuyến nghị thường gặp trong thực tế như:

- Facebook hiển thị quảng cáo những sản phẩm có liên quan đến từ khóa người dùng vừa tìm kiếm.
- Youtube tự động chuyển đến các video liên quan đến video người dùng vừa xem.
- Netflix gợi ý phim cho người dùng.

Nhìn chung, mục đích chính của các hệ thống khuyến nghị là dự đoán mức độ quan tâm của người dùng tới một sản phẩm nào đó rồi đề xuất các sản phẩm phù hợp, Vì vậy hệ thống khuyến nghị có vai trò quan trọng trong nhiều lĩnh vực như thương mại điện tử, mạng xã hội, giải trí,..., tạo ra nguồn thu khổng lồ cũng như lượng người dùng đông đảo.

Hệ thống khuyến nghị thường được chia thành hai loại:

- Lọc dựa trên nội dung: hệ thống đề xuất các sản phẩm tương đồng với người dùng thông qua các đặc trưng. Cách tiếp cận này yêu cầu phân loại các sản phẩm hoặc tìm đặc trưng của chúng.
- Lọc công tác: hệ thống gợi ý các sản phẩm dựa trên sự tương đồng của người dùng và sản phẩm. Các sản phẩm được đề xuất tới một người dùng dựa trên những người dùng có hành vi hoặc sở thích tương tự.

2.1.1 Lọc dựa trên nội dung

Lọc dựa trên nội dung là phương pháp đề xuất các sản phẩm mà người dùng có thể quan tâm dựa trên các đặc tính nội dung của chính sản phẩm đó. Lọc dựa trên nội dung xây dựng hồ sơ (thường là một vector) cho sản phẩm và người dùng dựa trên các đặc trưng của chúng. Sau khi xây dựng xong hồ sơ, lọc dựa trên nội dung sẽ sử dụng các phương pháp tính độ tương đồng giữa người dùng và sản phẩm thông qua hồ sơ tương ứng. Các sản phẩm có độ tương đồng cao sẽ được gợi ý cho người dùng.

Hệ thống khuyến nghị sử dụng lọc dựa trên nội dung có ưu và nhược điểm như sau:

- Ưu điểm:
  - Có khả năng gợi ý các sản phẩm cá nhân hóa cho người dùng.
  - Không cần sử dụng thông tin về hành vi của người dùng hoặc thông tin từ người dùng khác.
  - Có khả năng đề xuất các sản phẩm mới mà người dùng chưa biết trước đó.
- Nhược điểm:
  - Không sử dụng đánh giá phản hồi của người dùng về sản phẩm, do đó có thể bỏ qua những thông tin quan trọng về sở thích của họ.
  - Cách xây dựng hồ sơ ảnh hưởng lớn đến kết quả gợi ý.
  - Đề xuất sản phẩm hạn chế do chỉ sử dụng thông tin hiện có của sản phẩm.

Ví dụ trong một hệ thống xem phim trực tuyến, hồ sơ của người dùng và sản phẩm (bộ phim) gồm hai trường là thể loại và thời lượng. Hồ sơ được biểu diễn dưới dạng vector có độ dài 2, mỗi phần tử tương ứng với một trường. Hồ sơ của tài khoản `hai3k` được biểu diễn như sau:

$$v_{hai3k} = [1 \quad 1].$$

Hồ sơ của 3 bộ phim tương ứng là:

$$\begin{aligned} v_1 &= [1 \quad 1], \\ v_2 &= [1 \quad 0], \\ v_3 &= [0 \quad 1]. \end{aligned}$$

Hệ thống khuyến nghị tính độ tương đồng bằng cách tính cosin giữa hồ sơ người dùng và hồ sơ bộ phim và đề xuất bộ phim có độ tương đồng lớn hơn 0.7. Với tài khoản `hai3k` và 3 bộ phim trên ta có:

$$\begin{aligned} cosin_1 &= 1, \\ cosin_2 &= \frac{1}{\sqrt{2}}, \\ cosin_3 &= \frac{1}{\sqrt{2}}. \end{aligned}$$

Vậy bộ phim thứ nhất sẽ được đề xuất cho tài khoản `hai3k`.

2.1.2 Lọc cộng tác

Lọc cộng tác là cách phổ biến để thiết kế hệ khuyến nghị. Lọc cộng tác dựa trên giả định rằng sở thích của người dùng là không thay đổi, các nhóm người dùng chung sở thích sẽ cùng thích các sản phẩm. Cơ chế hoạt động của lọc cộng tác dựa trên hai phương pháp chính:

- Lọc cộng tác dựa trên người dùng: phương pháp này xác định sự tương đồng giữa người dùng bằng cách so sánh các sản phẩm đã đánh giá trong quá khứ. Nếu hai người dùng có các đánh giá tương tự với một tập hợp các sản phẩm, hệ thống sẽ đề xuất các sản phẩm mà một người dùng đã đánh giá đến cho người dùng khác.
- Lọc cộng tác dựa trên sản phẩm: phương pháp này xác định sự tương đồng giữa các sản phẩm dựa trên cách người dùng đánh giá với chúng. Nếu hai sản phẩm có độ tương đồng cao, khi người dùng mua một trong hai sản phẩm, hệ thống sẽ đề xuất sản phẩm còn lại.

Hệ thống khuyến nghị sử dụng lọc cộng tác có ưu điểm và khuyết điểm như sau:

- Ưu điểm:
  - Không cần thông tin chi tiết về nội dung sản phẩm, chỉ cần thông tin đánh giá giữa người dùng và sản phẩm. Điều này giúp dễ dàng triển khai và tổng hợp dữ liệu.
  - Hiệu quả khi số lượng sản phẩm lớn vì lọc cộng tác không cần xử lý nội dung của sản phẩm.
- Nhược điểm:
  - Với những người dùng và sản phẩm mới sẽ không có dữ liệu để đưa ra đề xuất chính xác.
  - Với hệ thống khuyến nghị lớn với hàng triệu người dùng và sản phẩm, cần một hệ thống tính toán mạnh mẽ.
  - Người dùng chỉ đánh giá với một số lượng ít sản phẩm. Đôi khi những sản phẩm được ưa chuộng lại không có nhiều đánh giá.

Để biểu diễn sự đánh giá giữa người dùng và sản phẩm, lọc cộng tác thường sử dụng ma trận tiện ích (utility matrix). Ma trận này được xây dựng từ dữ liệu thu thập được về việc người dùng đã đánh giá như thế nào với các sản phẩm trong hệ thống. Giả sử trong hệ thống có m người dùng và n sản phẩm, ma trận tiện ích lúc này sẽ có cỡ là $m \times n$. Giá trị ở mỗi ô (i, j) biểu thị đánh giá của người dùng i với sản phẩm j.

Quay trở lại với ví dụ về hệ thống xem phim trên, giả sử hệ thống có 4 tài khoản và 4 bộ phim, đánh giá của người dùng về bộ phim là một số nguyên từ 0 đến 5 tương ứng với mức độ ưa thích tăng dần. Các phần tử có dấu $*$ nghĩa là chưa có đánh giá giữa bộ phim với người dùng tương ứng.

0	*	1	5
5	5	0	*
*	*	*	1
1	4	*	0

Bảng 1: Ví dụ về ma trận tiện ích trong hệ thống xem phim.

Như đã nêu trên, người dùng thường chỉ đánh giá một lượng nhỏ số lượng sản phẩm trong hệ thống nên ma trận tiện ích thường là ma trận thưa. Số lượng ô được đánh giá càng nhiều thì hệ thống sẽ đưa ra gợi ý càng chính xác. Có hai cách phổ biến để xây dựng các giá trị trong ma trận tiện ích:

- Nhờ người dùng đánh giá sản phẩm. Các sàn thương mại điện tử luôn nhờ người dùng đánh giá các sản phẩm của họ. Rất nhiều hệ thống khác cũng làm việc tương tự. Tuy nhiên, cách tiếp cận này có một vài hạn chế, vì thường thì người dùng ít khi đánh giá sản phẩm. Và nếu có, đó có thể là những đánh giá thiên lệch.
- Hướng tiếp cận thứ hai là dựa trên hành vi của người dùng. Rõ ràng, nếu một người dùng mua một sản phẩm trên Shopee, Lazada, xem một clip trên Youtube (có thể là nhiều lần), hay đọc một bài báo,thì có thể khẳng định rằng người dùng đó thích sản phẩm đó.

2.1.3 nDCG@K

Để đánh giá chất lượng của một hệ khuyến nghị, ta thường dùng chỉ số nDCG@K. Chỉ số này đánh giá chất lượng của một truy vấn, trong ngữ cảnh của hệ khuyến nghị, nó đánh giá chất lượng của một dãy K sản phẩm được đề xuất cho người dùng. Khi một sản phẩm thuộc dãy và phù hợp với người dùng, được xếp hạng cao, sẽ đóng góp vào chỉ số nDCG@K nhiều hơn khi sản phẩm này được xếp hạng thấp. Điều này là hợp lý vì một hệ khuyến nghị tốt khi nó đề xuất những sản phẩm hợp với người dùng đầu tiên.

Xét một dãy đề xuất có K sản phẩm. Chỉ số nDCG@K được tính thông qua hai chỉ số:

$$nDCG@K = \frac{DCG_K}{IDCG_K}.$$

Trong đó:

$$DCG_K = \sum_{i=1}^{K} \frac{rel_i}{\log_2(i+1)},$$

$$IDCG_K = \sum_{i=1}^{|REL_K|} \frac{rel_i}{\log_2(i+1)}.$$

Trong đó rel_i là mức độ liên quan của sản phẩm thứ i và REL_K là dãy có K sản phẩm sao cho chỉ số DCG_K của dãy này là lớn nhất. Như vậy chỉ số nDCG@K luôn nhỏ hơn 1 và chỉ số này càng cao thì hệ khuyến nghị càng hiệu quả.

2.2 Biểu diễn từ trong NLP

Phương pháp nhúng từ (Word Embedding) được sử dụng để số hóa các từ trong ngôn ngữ tự nhiên để máy có thể hiểu và xử lý dữ liệu văn bản. Trong ngôn ngữ tự nhiên, đầu vào của mô hình là các từ và dấu câu. Để máy có thể hiểu được, ta cần biểu diễn các từ dưới dạng số.

2.2.1 Túi từ - Bag of words

Mô hình túi từ (bag of words) là một cách trích xuất các đặc trưng từ văn bản. Cách tiếp cận này có ưu điểm là rất đơn giản và dễ cài đặt. Một túi từ là một đại diện của văn bản mô tả sự xuất hiện của các từ trong một tài liệu, phụ thuộc vào hai yếu tố: một từ điển chứa các từ vựng đã biết; sự xuất hiện của từ trong từ điển trên tài liệu.

Tuy nhiên, nhược điểm của phương pháp này là bất kỳ thông tin nào về thứ tự hoặc cấu trúc của các từ trong tài liệu đều bị loại bỏ. Mô hình chỉ quan tâm đến việc các từ đã biết có xuất hiện trong tài liệu hay không, chứ không phải ở vị trí nào trong tài liệu.

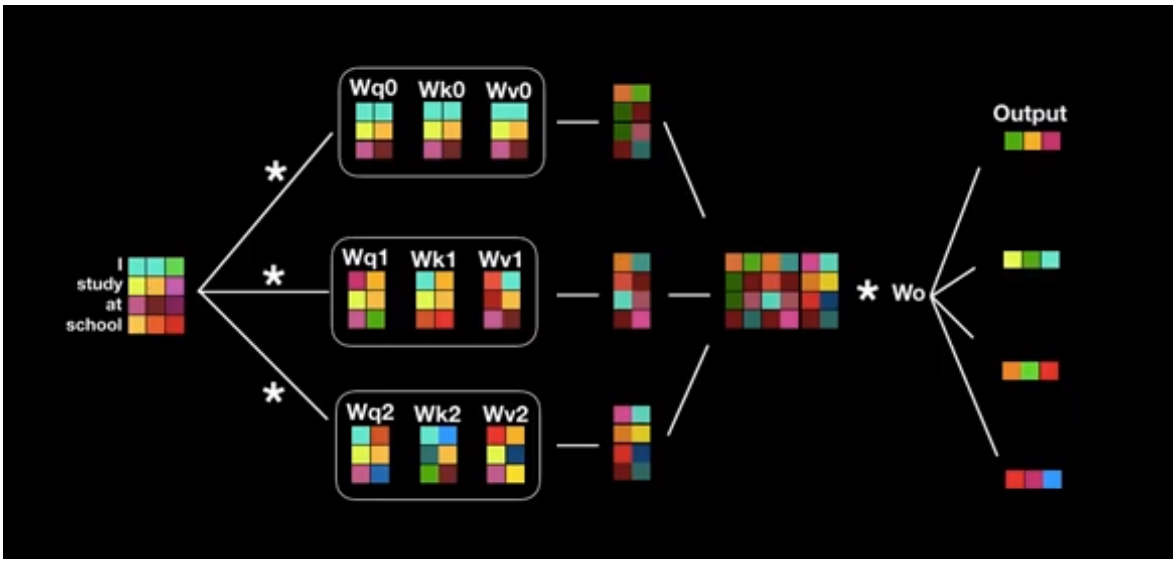
Ví dụ 2.1. *Với từ điển:{"it", "was", "the", "best", "of", "times","worst", "age","wisdom", "foolishness"} và câu cần biểu diễn: "it was the worst of times".*

Ta thu được vector biểu diễn: [1, 1, 1, 0, 1, 1, 1, 0, 0, 0]

2.2.2 Mô hình Sentence-BERT

Trước hết, ta sẽ nói sơ qua về mô hình BERT [4]. BERT là một model biểu diễn ngôn ngữ (Language Model- LM) được Google giới thiệu vào năm 2018. Trước khi BERT ra đời thì các tác vụ như: phân loại cảm xúc văn bản (tốt hay xấu, tích cực hay tiêu cực), sinh văn bản, dịch máy,... đều sử dụng kiến trúc RNN. Kiến trúc này có nhiều nhược điểm như train chậm, mất quan hệ giữa các từ xa nhau,... Với sự ra đời của BERT thì các tác vụ nêu trên đều được giải quyết với hiệu suất được cải thiện hơn.

Kiến trúc của BERT là một kiến trúc đa tầng bao gồm nhiều lớp Transformer Encoder. Chính kiến trúc như vậy khiến cho BERT có thể học được mối liên hệ giữa các từ xung quanh với nhau nhờ vào cơ chế Attention.



Hình 1: Kiến trúc Multi-head Attention với 3 head

Hai cấu hình phổ biến của BERT gồm:

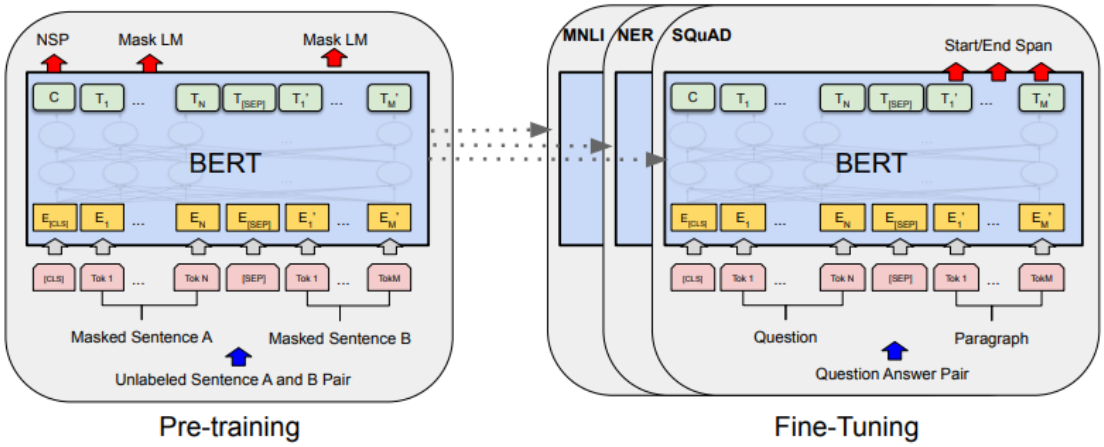
- $BERT_{BASE} : L = 12, H = 768, A = 12$
- $BERT_{LARGE} : L = 24, H = 1024, A = 16$

Trong đó:

- L: số lớp Transformer
- H: kích thước của vector biểu diễn

- A: số head ở lớp Attention

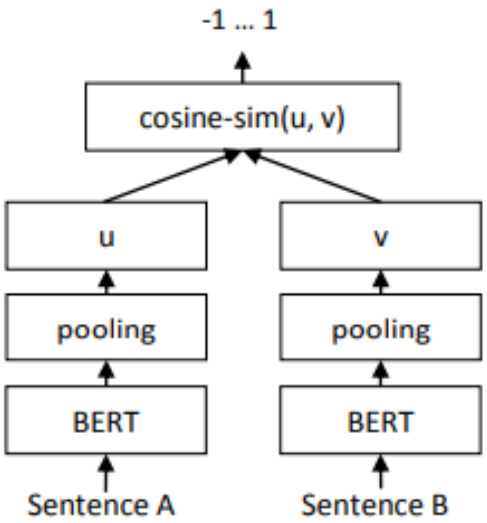
Một điểm thú vị ở BERT kết quả huấn luyện có thể fine-tuning được. Chúng ta sẽ thêm vào kiến trúc model một output layer để tùy biến theo tác vụ huấn luyện.



Hình 2: Tiến trình pre-training và fine-tuning của BERT

Quá trình pre-training của BERT được thực hiện dựa trên hai tác vụ chính là Masked LM (dự đoán từ thiếu trong câu) và Next Sentence Prediction (NSP – dự đoán câu tiếp theo câu hiện tại) với đầu vào được embedding bằng WordPiece - một từ điển chứa 30000 từ.

Một trong các mô hình fine-tuning có thể kể đến chính là Sentence-BERT [5] (S-BERT). Đây là một mô hình được fine-tuning dựa trên mô hình BERT để thực hiện tác vụ so sánh độ trùng khớp giữa các câu với nhau. Mô hình cố gắng ánh xạ các câu đầu vào có ý nghĩa tương đương sang một không gian vector mà ở đó chúng có biểu diễn gần giống nhau dựa trên kiến trúc Siamese.



Hình 3: Kiến trúc mô hình S-BERT

Kiến trúc Siamese sử dụng kết hợp hai mạng đầu vào giống hệt nhau mục đích là để có thể tính toán độ tương đồng giữa hai vector đầu ra qua độ đo cosine với hàm mất mát là hàm $L = \frac{1}{n}\sum_{i=1}^{n}(u_i - v_i)^2$.

Có một cách khác là chúng ta có thể sử dụng hàm mất mát là hàm Triplet. Mục tiêu là chúng ta sẽ tối thiểu độ tương đồng giữa hai câu không cùng nghĩa (anchor - negative) và tối đa độ tương đồng giữa hai câu cùng nghĩa (anchor - positive). Hàm có dạng như sau:

$$L = \max(\|s_a - s_p\| - \|s_a - s_n\| + \epsilon, 0).$$

Trong đó:

- s_x: vector embedding của các câu $a/p/n$
- $\|\cdot\|$: chuẩn độ dài
- ϵ: một lề được thêm vào để đảm bảo s_p gần s_a hơn s_n ít nhất một đoạn ϵ.

Thực nghiệm đã chỉ ra được kết quả khá tốt của S-BERT khi so sánh với những mô hình cũ như Glove hay FastText. Chính vì lí do như vậy và cùng với các đánh giá thực nghiệm nên mô hình được chọn để ứng dụng trong bài báo cáo là một biến thể của S-BERT: mô hình `paraphrase-MiniLM-L6-v2` [5][1].

2.3 Các độ đo tương đồng (Similarity Measures)

2.3.1 Cosine Similarity.

Độ đo Cosine là một phép đo độ tương đồng giữa hai vector trong không gian nhiều chiều. Độ đo này đo lường sự tương đồng hướng của hai vector thay vì khoảng cách euclidean giữa chúng.

Giả sử chúng ta có hai vector A và B trong không gian nhiều chiều với các thành phần tương ứng là $[A_1, A_2, ..., A_n]$ và $[B_1, B_2, ..., B_n]$. Độ đo Cosine giữa hai vector A và B được tính bằng cách sử dụng công thức sau:

$$Cosine(A, B) = \frac{(A.B)}{\|A\| * \|B\|}.$$

Trong đó:

- $(A.B)$ là tích vô hướng của hai vector A và B.
- $\|A\|$ và $\|B\|$ là độ lớn (norm) của hai vector A và B, được tính bằng căn bậc hai của tổng bình phương các phần tử của vector.

Kết quả của độ đo Cosine nằm trong khoảng [-1, 1]. Độ đo Cosine càng gần 1 tức là hai vector gần nhau, ngược lại càng gần về -1 tức là hai vector trái ngược nhau.

Độ đo Cosine rất hữu ích trong nhiều bài toán xử lý ngôn ngữ tự nhiên (NLP) và thị giác máy tính. Trong NLP, nó thường được sử dụng để tính độ tương đồng giữa các văn bản, từ đó tạo ra các biểu diễn vector cho văn bản và tính toán tương đồng giữa chúng. Trong thị giác máy tính, độ đo Cosine có thể được sử dụng để so sánh các đặc trưng của hình ảnh.

Vì tính đơn giản và hiệu quả của nó, độ đo Cosine là một công cụ quan trọng trong việc đo lường độ tương đồng và tính toán vector biểu diễn trong nhiều bài toán khác nhau.

2.3.2 Jaccard Similarity.

Độ đo Jaccard là một phép đo độ tương đồng giữa hai tập hợp. Độ đo này đo lường sự tương đồng giữa hai tập hợp dựa trên tỉ lệ phần tử chung của chúng so với tổng số phần tử của hai tập.

Cho hai tập hợp A và B, độ đo Jaccard được tính bằng cách lấy số phần tử chung của hai tập và chia cho tổng số phần tử của hai tập:

$$Jaccard(A, B) = \frac{|A \cap B|}{|A \cup B|}.$$

Trong đó:

[1] https://huggingface.co/sentence-transformers/paraphrase-MiniLM-L6-v2

- $|A \cap B|$ là số phần tử chung của hai tập hợp A và B.
- $|A \cup B|$ là tổng số phần tử của hai tập hợp A và B, bao gồm cả phần tử chung.

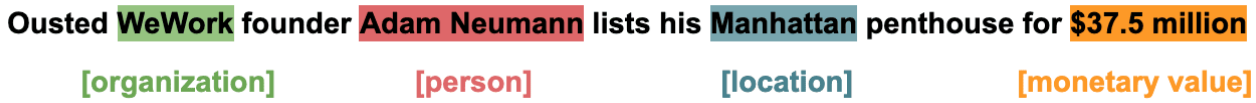
Kết quả của độ đo Jaccard nằm trong khoảng [0, 1]. Nếu độ đo Jaccard càng gần 1, tức là hai tập hợp A và B hoàn toàn giống nhau, có các phần tử giống nhau. Ngược lại, nếu độ đo Jaccard càng gần 0 tức là không có nhiều phần tử chung nào, độ tương đồng thấp.

2.4 Trích xuất thông tin văn bản trong NLP

Hiện nay, dữ liệu thường được trình bày dưới dạng nguyên bản (không có cấu trúc, sử dụng ngôn ngữ tụ nhiên) từ nhiều lĩnh vực như kinh tế, y tế, đời sống,... Do đó việc tóm tắt, tìm kiếm, trích xuất thông tin, rút ra kết luận và phân tích thống kê là những nhiệm vụ khó khăn với con người. Vì vậy, chúng ta sử dụng các mô hình xử lý ngôn ngữ tự nhiên NLP (natural language processing) để xử lý văn bản và giải quyết các nhiệm vụ trên. Điều này làm NLP có vai trò quan trọng trong việc xử lý dữ liệu. Một trong những tác vụ cơ bản của NLP là nhận dạng và phân loại thực thể trong văn bản hay còn gọi là NER (Named Entity Recognition).

2.4.1 Tổng quan về NER

NER là là một nhiệm vụ con của trích xuất thông tin nhằm tìm cách định vị và phân loại các thực thể được đề cập trong văn bản phi cấu trúc vào các loại đã được xác định trước, chẳng hạn như tên người, tổ chức, địa điểm, mã y tế , biểu thức thời gian, số lượng, giá trị tiền tệ, tỷ lệ phần trăm,...



Hình 4: Minh họa nhận diện thực thể trong NER.

Để xây dựng một hệ thống NER, hiện nay có một số cách tiếp cận phổ biến sau:

- Cách tiếp cận dựa trên quy tắc.
- Cách tiếp cận học có giám sát.
- Cách tiếp cận sử dụng mô hình có sẵn.

Hướng tiếp cận dựa trên quy tắc

Cách tiếp cận này dựa trên việc sử dụng các quy tắc và các mẫu ngữ pháp để xác định và phân loại các thực thể định danh trong văn bản. Thay vì sử dụng các thuật toán máy học để học từ dữ liệu, phương pháp này dựa vào các quy tắc đã được xác định trước bởi con người. Các quy tắc có thể được xây dựng dựa trên các luật ngữ nghĩa, một từ điển chứa danh sách các từ được định danh, hoặc các biểu thức chính quy để tìm kiếm các mẫu đặc trưng trong văn bản. Ví dụ một quy tắc đơn giản là "Nếu một từ bắt đầu bằng chữ hoa và tiếp theo là các chữ thường, thì đó là tên riêng."

Một số ưu nhược điểm của các tiếp cận này có thể kể tới như:

- Ưu điểm:
  - Đơn giản và dễ triển khai: không yêu cầu nhiều dữ liệu huấn luyện và có thể được triển khai nhanh chóng chỉ bằng cách định nghĩa các quy tắc thủ công, không cần huấn luyện mô hình, điều này tiết kiệm thời gian và tài nguyên trong quá trình triển khai.

  - Chĩnh xác cao: khi được thiết kế đúng, các quy tắc có thể đảm bảo mức độ chính xác cao trong việc xác định các thực thể. Đặc biệt trong các loại văn bản có cấu trúc đơn giản và nhiều từ vựng chuyên ngành.

- Nhược điểm:

  - Khó xử lý các mẫu phức tạp: các quy tắc có giới hạn trong khả năng biểu diễn các mẫu phức tạp và không thể tự động học từ dữ liệu mới. Điều này làm cho phương pháp này không hiệu quả khi đối mặt với các ngôn ngữ phức tạp hoặc không rõ ràng.
  - Không tổng quát hóa: các quy tắc được thiết kế dựa trên ngôn ngữ và lĩnh vực cụ thể, do đó khó áp dụng sang các lĩnh vực khác.

Hướng tiếp cận học có giám sát

Cách tiếp cận này thường sử dụng khi dữ liệu đã được đánh nhãn. Mô hình được huấn luyện dựa trên dữ liệu gán nhãn bằng nhiều thuật toán khác nhau như Hidden Markov Model, Support Vector Machines,... Sau đây là một số ưu nhược điểm của cách làm này:

- Ưu điểm

  - Tích hợp thông tin ngữ cảnh: học có giám sát cho phép mô hình học từ các quan hệ ngữ nghĩa và ngữ cảnh trong dữ liệu huấn luyện. Điều này giúp nâng cao khả năng hiểu ngữ cảnh và xử lý các mẫu phức tạp trong văn bản.
  - Tính tổng quát hóa cao: mô hình có khả năng tổng quát hóa sang các văn bản mới và các lĩnh vực khác. Nó có thể xử lý các thực thể không xuất hiện trong dữ liệu huấn luyện nhờ vào việc học các đặc trưng chung của các loại thực thể.
  - Tự động học từ dữ liệu: mô hình có khả năng tự động học từ dữ liệu huấn luyện, giúp giảm thiểu sự can thiệp của con người..

- Nhược điểm

  - Đòi hỏi dữ liệu huấn luyện đủ lớn: mô hình yêu cầu một lượng lớn dữ liệu huấn luyện đã được gán nhãn để đảm bảo tính chính xác và hiệu quả. Việc thu thập và gán nhãn dữ liệu có thể tốn nhiều thời gian và công sức.

Hướng tiếp cận sử dụng mô hình có sẵn

Cách làm này sử dụng mô hình ngôn ngữ đã được huấn luyện trước với một lượng lớn dữ liệu và kiến thức ngôn ngữ rộng lớn từ nhiều nguồn khác nhau. Các mô hình này thường được huấn luyện trên nhiều tác vụ khác nhau và có khả năng hiểu ngữ cảnh và ngữ nghĩa của văn bản. Người dùng có thể tiếp tục huấn luyện mô hình có sẵn trên dữ liệu của mình để nâng cao khả năng xác định các thực thể trong lĩnh vực cụ thể. Một số mô hình có sẵn phổ biến như BERT, XLNet, ALBERT...

2.4.2 Một số thư viện tích hợp NER

SpaCy

SpaCy là một gói thư viện trong Python được sử dụng cho xử lý ngôn ngữ tự nhiên cấp cao và mã nguồn mở. Nó hỗ trợ việc tạo ra các chương trình có khả năng "hiểu" và xử lý lượng lớn văn bản. SpaCy hỗ trợ hơn 70 ngôn ngữ và đi kèm với các mô hình huấn luyện đã có sẵn cho nhiều ngôn ngữ khác nhau.

Natural Language Tool Kit (NLTK)

Natural Language Tool Kit (NLTK) là một thư viện mã nguồn mở phổ biến trong ngôn ngữ lập trình Python, được phát triển để hỗ trợ xử lý ngôn ngữ tự nhiên (NLP) và nghiên cứu về xử lý ngôn ngữ tự nhiên. NLTK được phát triển bởi Steven Bird và Edward Loper từ năm 2001 và tiếp tục được cải tiến và phát triển bởi cộng đồng người dùng và nhà phát triển NLP.

TensorFlow

TensorFlow là một thư viện mã nguồn mở mạnh mẽ trong lĩnh vực học máy và trí tuệ nhân tạo. Được giới thiệu lần đầu vào năm 2015 bởi Google Brain Team, TensorFlow đã trở thành một trong những công cụ phổ biến nhất trong cộng đồng học máy và AI. Với ngôn ngữ lập trình Python, C++ và CUDA, TensorFlow cung cấp một bộ công cụ toán học mạnh mẽ, cho phép xây dựng và triển khai các mô hình học máy và trí tuệ nhân tạo đa dạng.

2.5 Tổng quan mô hình

2.5.1 Mô hình hóa bài toán

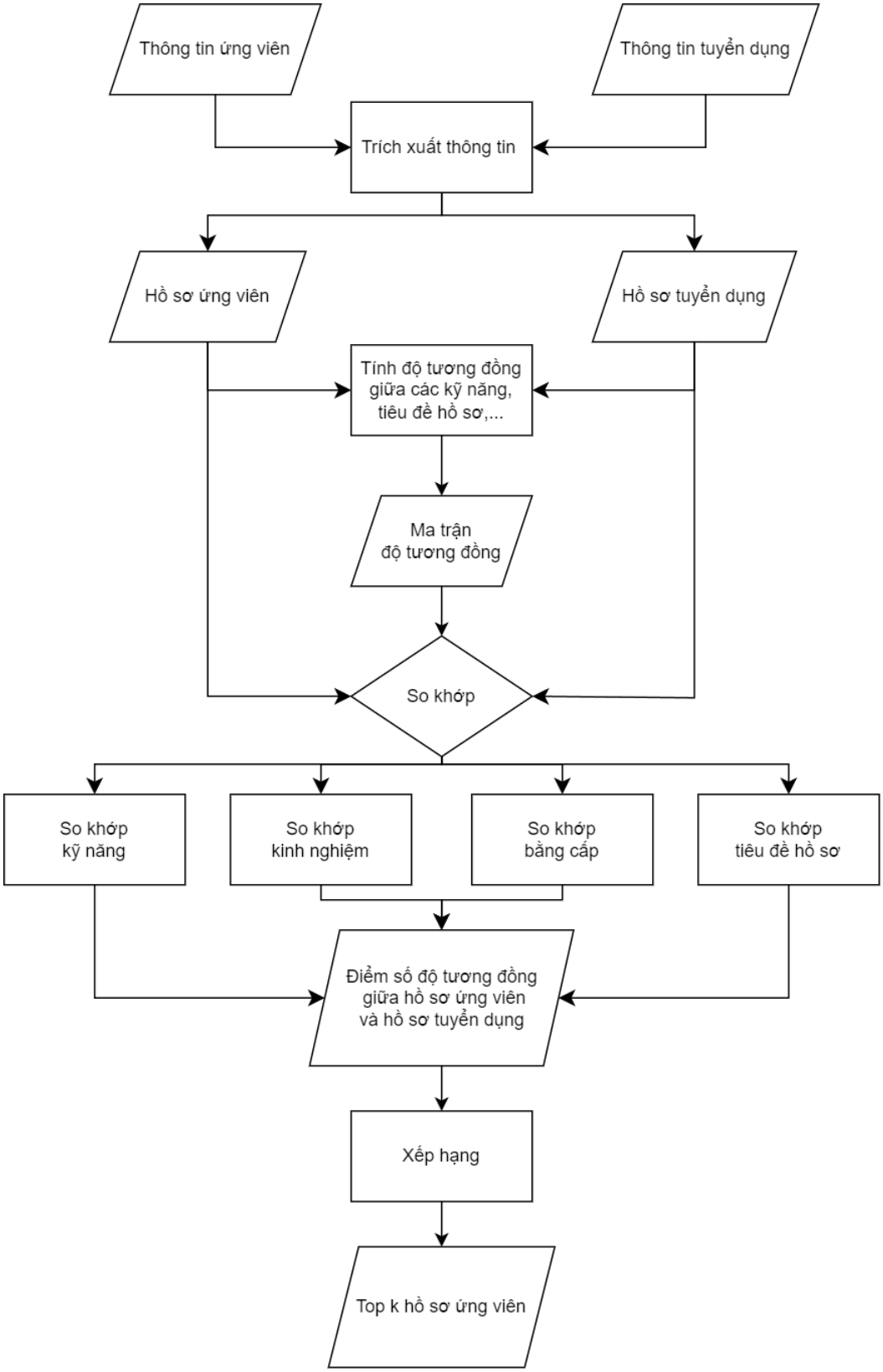
Với những yêu cầu của bài toán gợi ý nhân sự, xây dựng mô hình cho hệ khuyến nghị như sau:

- Đầu vào:
  - $\mathbf{R} = \{r_1, \cdots, r_n\}$ là tập các hồ sơ ứng viên.
  - $\mathbf{J} = \{j_1, \cdots, j_m\}$ là tập các hồ sơ tuyển dụng.
- Đầu ra:
  - Với mỗi hồ sơ tuyển dụng $j \in \mathbf{J}$, đề xuất k hồ sơ ứng viên $r \in \mathbf{R}$.

2.5.2 Mô hình hệ thống khuyến nghị

Hệ thống mà nhóm tác giả xây dựng là hệ thống khuyến nghị sử dụng lọc dựa trên nội dung. Do thông tin về các ứng viên và thông tin tuyển dụng thường được chứa trong tệp loại PDF hoặc JSON nên cần phải trích xuất thông tin các tệp này để tạo hồ sơ tuyển dụng và ứng viên. Sau khi có hồ sơ, ta tiến hành so khớp, do các thông tin đều ở dạng văn bản nên nhóm tác giả sẽ vector hóa các trường cần thiết như kỹ năng, tiêu đề, vị trí đã làm việc,... và sử dụng độ đo cosine để tính độ tương đồng giữa chúng. Do các trường trên xuất hiện lặp lại trong hồ sơ ứng viên cũng như tuyển dụng, tác giả sẽ tính sẵn các độ tương đồng này và lưu vào các ma trận để sử dụng cho lần so khớp tiếp theo. Điểm của mỗi hồ sơ ứng viên sẽ được tính bằng tổng các độ tương đồng của các trường được so khớp với trọng số thích hợp. Tổng quan, mô hình hoạt động theo 4 bước sau:

1. Trích xuất thông tin và tạo các hồ sơ ứng viên, hồ sơ tuyển dụng.
2. Vector hóa và tính độ tương đồng giữa các trường thông tin như kỹ năng, tiêu đề hồ sơ,... sau đó lưu vào các ma trận.
3. Sử dụng các ma trận để tiến hành so khớp và tính điểm giữa hồ sơ ứng viên và hồ sơ tuyển dụng.
4. Sắp xếp các hồ sơ ứng viên theo điểm và đưa ra danh sách k hồ sơ phù hợp với hồ sơ tuyển dụng.



Hình 5: Tổng quan mô hình hệ thống khuyến nghị.

3 Ứng dụng các kỹ thuật NLP cho hệ thống khuyến nghị trong lĩnh vực tuyển dụng

3.1 Datasets

Trong báo này, nhóm sử bộ dữ liệu JD-Resume 897 mẫu do giảng viên TS. Nguyễn Thị Ngọc Anh cung cấp. Bộ dữ liệu bao gồm 897 job decripstion, 897 resumes và 1 file csv chứa thông tin về độ đánh giá, nhãn của jd với resume.

Thông tin về job decripstion
Các trường quan trọng được sử dụng trong job decripstion (JD) bao gồm là:

- *id*: Chỉ số id của JD.
- *title*: Tên của vị trí muốn tuyển dụng.
- *decripstion*: Mô tả công việc của vị trí tuyển dụng.
- *requirements*: Các yêu cầu về công việc của vị trí tuyển dụng.
- *required_skills*: Các kĩ năng chuyên môn yêu cầu của công việc.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 897 entries, 0 to 896
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   id               897 non-null    object
 1   title            897 non-null    object
 2   description      885 non-null    object
 3   requirements     897 non-null    object
 4   required_skills  897 non-null    object
dtypes: object(5)
memory usage: 35.2+ KB
```

Hình 6: Thông tin về JD.

Thống kê số lượng các vị trí cho thấy có tất cả 261 vị trí tuyển dụng và vị trí được tuyển nhiều nhất là Senior IOS developer, các vị trí Senior khác cũng được tuyển rất nhiều:

```
title
Senior IOS developer                                17
Senior .NET .Netcore developer                      16
Senior Front End Developer (ReactJs or VueJS …)     16
Senior Magento/Shopify developer (Remote)           15
Senior Front-end                                    14
                                                    ..
Front-end Developer (Middle Level - Senior )         1
PRODUCT MANAGER, FIN-TECH (ZALOPAY E-WALLET)         1
Trưởng Phòng Nhân Sự                                 1
REACT NATIVE FRONT END DEV                           1
Mobile Developer (Android/IOS)                       1
Name: count, Length: 261, dtype: int64
```

Hình 7: Thống kê các vị trí tuyển dụng.

Có tất cả 487 kĩ năng chuyên môn mà các nhà tuyển dụng yêu cầu và 5 kĩ năng xuất hiện nhiều nhất là về javascript, communication skill, management và git.

```
number skill = 487
javascript: 213
communication skill: 189
management: 183
git: 182
css: 167
problem solving: 149
html: 148
java: 142
```

Hình 8: Thống kê các kĩ năng yêu cầu.

Thông tin về resume
Một số trường chính của dữ liệu về resume là:

- *id*: Chỉ số id của resume.
- *fulltext*: Chứa thông tin về ứng viên.
- *degree*: Cấp bậc học vấn của ứng viên.
- *educations_gpa*: Chứa điểm của ứng viên tương ứng với các cấp bậc.
- *education_major*: Ngành học của ứng viên tương ứng với các cấp bậc.
- *positions*: Các vị trí làm việc mà ừng đã đảm nhiệm ở các công ty cũ.
- *years_positions*: Thông tin về thời gian làm việc tương ứng các positions.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 897 entries, 0 to 896
Data columns (total 8 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   id                897 non-null    object
 1   skills            897 non-null    object
 2   fulltext          897 non-null    object
 3   degree            897 non-null    object
 4   educations_gpa    897 non-null    object
 5   educations_major  897 non-null    object
 6   positions         897 non-null    object
 7   years_positions   897 non-null    object
dtypes: object(8)
memory usage: 56.2+ KB
```

Hình 9: Thông tin về resume.

Thống kê các kỹ năng xuất hiện nhiều nhất trên các resume cho thấy các kĩ năng như: management, javascript, database, html, css phổ biến trên các resumes.

```
skills
management    604
javascript    424
database      413
html          377
css           368
mysql         357
git           352
sql           348
java          335
analysis      295
Name: count, dtype: int64
```

Hình 10: Các kỹ năng phổ biến xuất hiện trong resumes.

Thông tin về bộ đánh giá

Trong file chứa các thông tin quan trọng để đánh giá mức độ phù hợp của JD với resume gồm các đánh giá quan trọng như experience, title score, overall score, cosine score skill và label.

job_description_id	resume_id	experience	education	language	title_score	description_score	yoe_score	overall_score	cosine_score_matching_skill	label
4288	4288	0.698490	0	0.0	0.566124	0.830855	NaN	0.483967	0.636829	1
4385	4385	0.905469	0	1.0	1.000000	0.810938	NaN	0.898703	0.452085	1
4515	4515	0.808731	0	0.0	0.696288	0.921175	NaN	0.692099	0.292798	1
5184	5184	0.209289	0	0.0	0.209289	NaN	NaN	0.355047	0.523493	1
4334	4334	0.669163	0	0.0	0.669163	NaN	NaN	0.740888	0.677293	1
...	...	...	...	...	...	...	...	...	...	...
4982	4287	0.538705	0	0.0	0.374487	0.702924	NaN	0.538705	0.218218	0
4982	4448	0.557140	0	0.0	0.390805	0.723474	NaN	0.557140	0.113961	0
4982	4329	0.543104	0	0.0	0.377300	0.708909	NaN	0.543104	0.150188	0
4982	4289	0.521524	0	0.0	0.327424	0.715624	NaN	0.521524	0.218218	0
4982	4595	0.512574	0	0.0	0.320549	0.704598	NaN	0.512574	0.188982	0

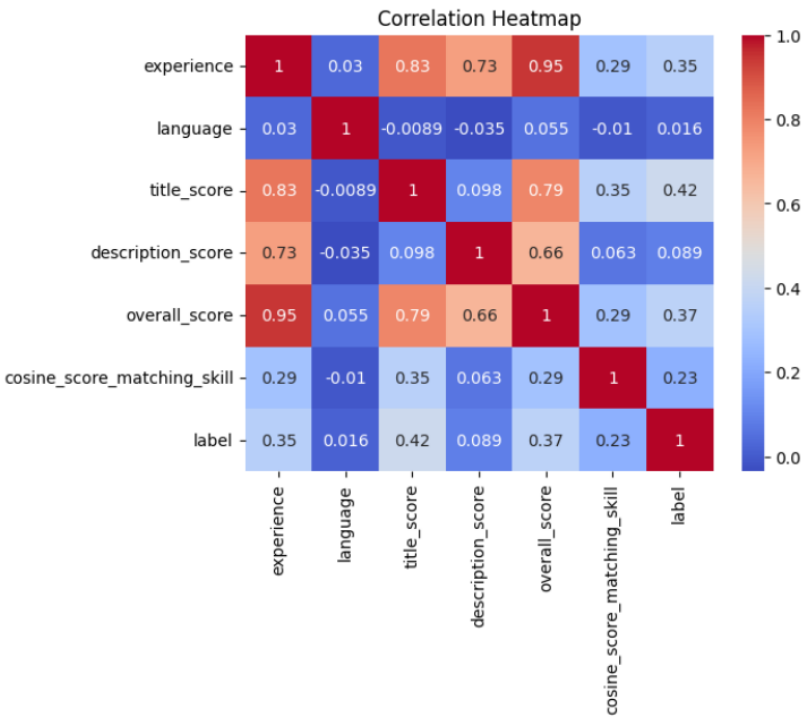
Hình 11: Độ đánh giá và nhãn của các JD và resume.

Có tất cả 9728 hàng tương ứng với 9728 labels, các JD và resume cùng có số id thường mang label 1 còn lại là 0. Các trường education và yoe_score không có giá trị nào hết.

	job_description_id	resume_id	experience	education	language	title_score	description_score	yoe_score	overall_score	cosine_score_matching_skill	label
count	9728.000000	9728.000000	9728.000000	9728.0	9728.000000	9722.000000	3849.000000	0.0	9728.000000	9574.000000	9728.000000
mean	4881.405530	4524.008224	0.486118	0.0	0.007093	0.432963	0.699106	NaN	0.487609	0.262358	0.090255
std	283.266915	299.544167	0.151959	0.0	0.083925	0.151900	0.141372	NaN	0.157041	0.195220	0.286561
min	4287.000000	4287.000000	0.000000	0.0	0.000000	0.057000	0.000000	NaN	0.000000	0.000000	0.000000
25%	4648.000000	4311.750000	0.372895	0.0	0.000000	0.319336	0.654406	NaN	0.375953	0.096693	0.000000
50%	4934.000000	4385.000000	0.505141	0.0	0.000000	0.417628	0.710449	NaN	0.503303	0.236443	0.000000
75%	5132.000000	4595.000000	0.585836	0.0	0.000000	0.540197	0.776837	NaN	0.584627	0.390704	0.000000
max	5322.000000	5322.000000	1.000000	0.0	1.000000	1.000000	0.936667	NaN	1.000000	1.000000	1.000000

Hình 12: Thống kê bảng đánh giá JD và resume.

Dựa theo ma trận tương quan có thể nhận xét được rằng các chỉ số đánh giá như title score, skill score, experience score đóng vai trò quan trọng, ảnh hưởng nhiều tới label.



Hình 13: Ma trận độ tương quan.

3.2 Xây dựng mô hình

Mô hình của nhóm được xây dựng theo hướng Lọc dựa trên nội dung (content-base) kết hợp với model Sentence-BERT và xây dựng dựa trên ngôn ngữ tiếng Anh nên đối với các JD, resume tiếng Việt sẽ có thể bị kém hiệu quả.

3.2.1 Trích xuất thông tin

Do dữ liệu chưa được gán nhán phù hợp theo NER cũng như thời gian có hạn, nên nhóm sử dụng phương pháp NER theo hướng tiếp cận dựa trên quy tắc bằng cách xây dựng các từ điển chứa danh sách các từ định danh và kết hợp sử dụng thư viện **Spacy** để trích xuất. Cụ thể, nhóm xây dựng quy tắc để trích yêu cầu về bằng cấp, trích yêu cầu về số năm kinh nghiệm và loại kinh nghiệm trong trường *requirements* của các JD.

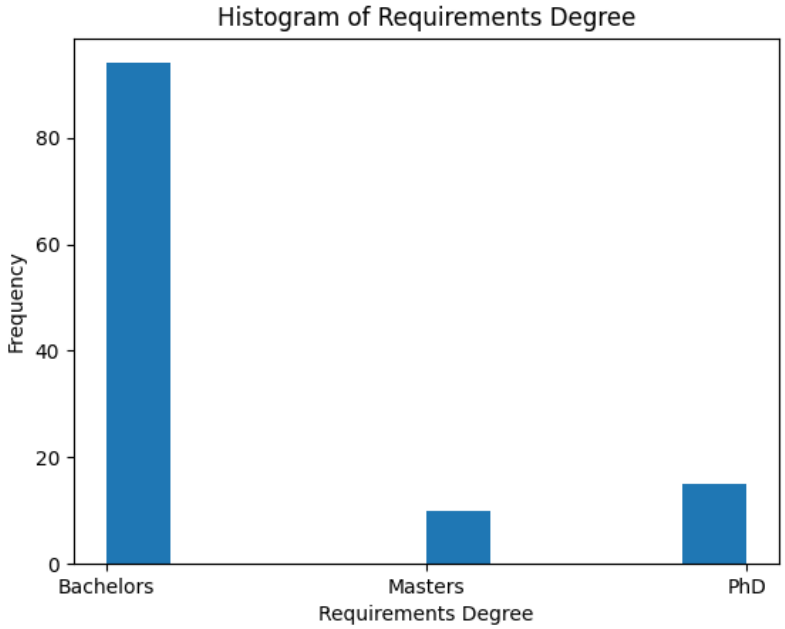
Trích yêu cầu về bằng cấp

Từ điển được sử dụng để trích về bằng cấp:

```
{"label": "DEGREE|Bachelors", "pattern": [{"LOWER": "bachelors"}]}
{"label": "DEGREE|Bachelors", "pattern": [{"LOWER": "high school diploma"}]}
{"label": "DEGREE|Bachelors", "pattern": [{"LOWER": "engineer"}]}
{"label": "DEGREE|Bachelors", "pattern": [{"LOWER": "colleges"}]}
{"label": "DEGREE|Bachelors", "pattern": [{"LOWER": "bachelor's"}]}
{"label": "DEGREE|Bachelors", "pattern": [{"LOWER": "bachelor"}]}
{"label": "DEGREE|Bachelors", "pattern": [{"LOWER": "college"}]}
{"label": "DEGREE|Bachelors", "pattern": [{"LOWER": "undergraduate"}]}
{"label": "DEGREE|Masters", "pattern": [{"LOWER": "master"}]}
{"label": "DEGREE|Masters", "pattern": [{"LOWER": "master's"}]}
{"label": "DEGREE|PhD", "pattern": [{"LOWER": "phd"}]}
{"label": "DEGREE|PhD", "pattern": [{"LOWER": "ph.d"}]}
{"label": "DEGREE|PhD", "pattern": [{"LOWER": "doctorate"}]}
```

Hình 14: Từ điển về bằng cấp.

Kết quả thu được có 119 JD có yêu cầu về bằng cấp, trong đó có 15 yêu cầu về bằng PhD và 10 yêu cầu về bằng Master.



Hình 15: Biểu đồ yêu cầu về bằng cấp.

Trích năm kinh nghiệm và loại kinh nghiệm

Về trích xuất số năm kinh nghiệm, nhóm sử dụng kĩ thuật regex expression để nhận diện được kí tự số và phần ngay sau đó (đến khi xuất hiện dấu câu) là loại kinh nghiệm tương ứng.

	id	requirements_year	requirements_major
2	4289	None	None
3	4290	[2 years-]	[]]
4	4291	[5-year, 1 year]	[working experience and 1 -2 experience in team management , experience in E - commerce or Tech startup]
5	4292	None	None
6	4293	[1,5 years]	[' experience at web applications]
7	4294	[3- years]	[' experience as a Software Developer]

Hình 16: Trích xuất thông về kinh nghiệm.

Nhóm trích được tất cả 555 yêu cầu về kinh nghiệm trong JD. Các trường hợp không trích được có thể do năm kinh nghiệm ở dạng chữ, năm kinh nghiệm đứng cuối cầu và do jd thuộc loại tiếng Việt.

3.2.2 Xây dựng ma trận tương đồng

Word Embedding

Nhóm sử dụng model `paraphrase-MiniLM-L6-v2` [5][2] để thực hiện nhúng từ. Đây là mô hình được huấn luyện bởi sentence-transformer, mô hình ánh xạ các câu và đoạn văn tới một không gian vectơ 384 chiều và có thể được sử dụng cho các nhiệm vụ như phân cụm hoặc tìm kiếm ngữ nghĩa.

[2] https://huggingface.co/sentence-transformers/paraphrase-MiniLM-L6-v2

Ma trận tương đồng

Nhóm sử dụng độ đô Cosine làm độ đo chính để xác định độ tương đồng giữa các từ. Xây dựng các ma trận của các yếu tố:

- Skill - Skill: Ma trận kích thước 2284 tương ứng với độ tương đồng của các cặp skills (bao gồm cả skill của jd và resume).
- Title - Positions: Ma trận kích thước 2181 tương ứng độ đương đồng của các cặp title của JD và position của resume.

	asp	digital skill	automation testing tool	corporate stationery	swing	dojo	sps	frequent pattern mining	apache ignite	service management	reinforcement learning	sphinx	support and troubleshooting	sqlyog
asp	1.000000	0.205543	0.298829	0.295037	0.025158	0.199707	0.347172	0.193305	0.025532	0.327306	0.127913	0.058435	0.234748	0.253937
digital skill	0.205543	1.000000	0.403837	0.200126	0.078734	0.108810	0.127988	0.210041	-0.000158	0.101842	0.350093	-0.062782	0.154481	0.110563
automation testing tool	0.298829	0.403837	1.000000	0.113393	0.045746	0.002950	0.142275	0.201642	0.016586	0.243479	0.214352	0.027263	0.286459	0.102743
corporate stationery	0.295037	0.200126	0.113393	1.000000	0.001300	0.129273	0.226180	0.207767	0.013028	0.323987	0.090983	0.244812	0.123339	0.088839
swing	0.025158	0.078734	0.045746	0.001300	1.000000	0.169582	0.242992	0.073762	0.115854	0.095967	0.119468	-0.037656	-0.013908	-0.031119
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
woocommerce	0.180572	0.118439	0.063146	0.214821	0.093377	0.214434	0.299357	0.083210	0.029762	0.211114	0.107021	0.067130	0.137004	0.210899
graphql	0.075934	0.156540	0.120463	0.134118	0.051416	0.001837	-0.007757	0.307172	-0.048843	0.138398	0.151965	0.137682	0.108874	0.345729
data visualization	0.251521	0.383565	0.220866	0.131599	0.016293	0.012139	0.087654	0.259714	-0.097372	0.139418	0.249877	0.090423	0.068090	0.284916
rancher	0.001835	0.020450	0.025626	0.108275	0.303309	0.250623	0.106443	0.008062	0.085741	0.138813	0.177736	0.035883	0.140949	0.043531
disaster recovery	0.103502	0.064324	0.066991	0.089223	0.025562	0.056967	0.135547	-0.028488	0.153042	0.251557	0.169405	0.132010	0.245125	-0.045701

Hình 17: Ma trận tương đồng của skills.

	Category Operation Assistant Manager - MT & DT	Waiter / Food Runner	Nhân Viên Kế toán nội bộ kiêm hành chính văn phòng	Scrum Master\nSoftware Engineer	Business Analyst/ Project Manager	Cán bộ truyền thông	Assistant to Legal & Admin Manager	Admin - HR & Personal Assistant to Country Manager	Customs Brokerage / Warehouse Leader	Trợ lý	Senior Logistics Associate	SALES ACCOUNT MANAGER/ ACTING CEO	Senior Developer/ Technical Team Leader	BrSE/Project Manager	SOCIAL MEDIA MANAGER
Category Operation Assistant Manager - MT & DT	1.000000	0.236814	0.103736	0.315784	0.447104	0.012199	0.479340	0.440485	0.261948	0.135117	0.385912	0.463926	0.434701	0.383705	0.401738
Waiter / Food Runner	0.236814	1.000000	0.235905	0.097907	0.232740	0.262653	0.173171	0.212205	0.288457	0.210557	0.216355	0.352748	0.109340	0.161908	0.352931
Nhân Viên Kế toán nội bộ kiêm hành chính văn phòng	0.103736	0.235905	1.000000	0.124473	0.148535	0.396513	0.183147	0.292304	0.257987	0.376221	0.123853	0.153263	0.115353	0.229765	0.160548
Scrum Master\nSoftware Engineer	0.315784	0.097907	0.124473	1.000000	0.480804	0.069887	0.387441	0.274415	0.208607	0.171512	0.339782	0.224045	0.501409	0.394613	0.292936
Business Analyst/ Project Manager	0.447104	0.232740	0.148535	0.480804	1.000000	0.082377	0.464925	0.489133	0.404455	0.118720	0.469069	0.557036	0.604144	0.653652	0.518249
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Hình 18: Ma trận tương đồng của cặp title-positon.

3.2.3 Các luật so khớp

Với mỗi jd được yêu cầu, thực hiện so khớp với resume theo các luật so khớp như sau:

Xếp hạng bằng cấp:

Mục đích của điểm bằng cấp giống như là một điểm cộng cho ứng viên, tuy nhiên điểm bằng cấp sẽ không mang trọng số cao cho xếp hạng.

$$degree = \begin{cases} 0 & \text{nếu không có về bằng cấp.} \\ 0.5 & \text{với Bachelors, Colleges.} \\ 0.75 & \text{với Engineer, Masters.} \\ 1 & \text{với PhD.} \end{cases}$$

và

$$degree\_score = \begin{cases} 1 & \text{nếu } degree(resume) \geq degree(jd). \\ degree(resume)/degree(jd) & \text{nếu } degree(resume) < degree(jd). \end{cases}$$

So khớp skills

Skill là một yếu tố quan trọng trong việc ứng tuyển, do đó điểm skill sẽ mang trọng số cao. Với mỗi skill của jd (n skill) tiến hành so khớp với từng skill của resume (m skill) và lấy kết quả lớn nhất, nếu kết quả đó lớn hơn mọt ngưỡng cho trước thì đem tổng hợp lại và sau đó tính trung bình cộng để ra được skill_score:

$$skill\_score(i) = \max(cosine\_skill(skill\_jd_i, skill\_resume_j)\ \forall j = \overline{1, m}$$

$$skill\_score = \frac{1}{n}\sum_{i=1}^{n} skill\_score(i) \text{ v.đ.k } skill\_score(i) \geq threshold$$

So khớp title - position và số năm kinh nghiệm

Giống như skill, các vị trí đã từng làm ở công ty cũ và kinh nghiệm cũng là một yếu tố quan trọng trong việc xét tuyển. Mỗi title của JD sẽ được đo độ tương đồng với từng vị trí của position, nếu độ tương đồng đạt một ngưỡng cho trước thì sẽ được tính số năm kinh nghiệm của vị trí đó.

$$title\_score = \frac{1}{n}\sum_{i}^{n} cosine\_major(title, position_i)$$

$$year\_exp = \sum_{i}^{n} year\_position_i \text{ nếu } cosine\_major(title, position_i) \geq threshold$$

$$exp\_score = \begin{cases} 1 & \text{nếu } year\_exp \geq requirement\_year. \\ year\_exp/requirement\_year & \text{nếu } year\_exp < requirement\_year. \end{cases}$$

Cuối cùng tính tổng điểm dựa trên một trọng số cho trước và đem đi xếp hạng.

3.3 Kết quả mô hình

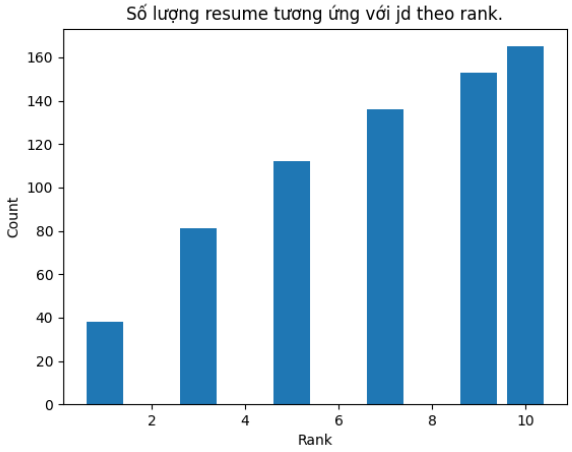
Sau nhiều lần thực nghiệm trên bộ dataset, thông số mô hình cho ra kết quả tốt nhất như sau:

- So khớp toàn bộ 897 JD với 897 resumes.
- Trọng số: [6, 1.5, 1.5, 1]

$$overall\_score = \frac{1}{10}(6.title\_score + 1.5.skill\_score + 1.5.exp\_score + degree\_score).$$

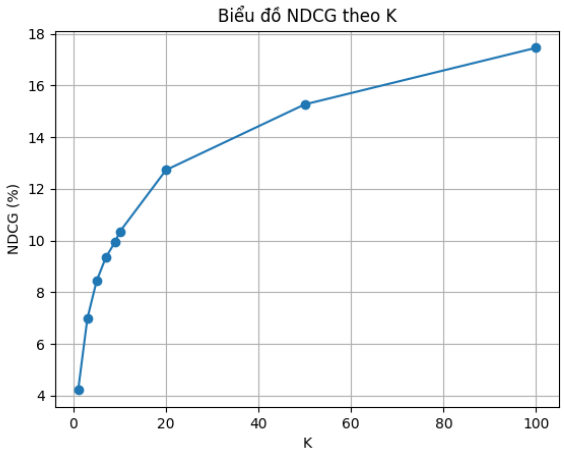
- $threshold = 0.5$
- Kết quả thu được:
  - Số lượng resume tương ứng với JD nằm trong rank 1: 38.

- Số lượng resume tương ứng với JD nằm trong rank 3: 81.
- Số lượng resume tương ứng với JD nằm trong rank 5: 112.
- Số lượng resume tương ứng với JD nằm trong rank 7: 136.
- Số lượng resume tương ứng với JD nằm trong rank 9: 153.
- Số lượng resume tương ứng với JD nằm trong rank 10: 165.



Hình 19: Số lượng resume tương ứng với JD theo rank.

- Kết quả chỉ số nDCG@K:
  - nDCG@1 = 4.22 %
  - nDCG@3 = 7.00 %
  - nDCG@5 = 8.45 %
  - nDCG@7 = 9.36 %
  - nDCG@9 = 9.96 %
  - nDCG@10 = 10.35 %



Hình 20: Biểu đồ NDCG@K.

3.4 Đánh giá mô hình

Do chưa có nhiều tiêu chí đánh chất lượng mô hình nên vẫn chưa đủ khả năng kết luận tính hiệu quả của mô hình. Tuy nhiên, dựa theo các nhãn label đã được cung cấp có thể nhận thấy rằng mô hình cho ra chỉ số nDCG@K không cao, nDCG@5 chỉ xấp xỉ khoảng 8.44%. Chỉ số nDCG@K thấp một phần là do sự thiếu xót của các nhãn label vì mỗi JD chỉ có ứng với 1 resume duy nhất còn lại đều là nhãn 0.

Qua quá trình kiểm tra các kết quả có thể cho thấy mô hình vẫn đưa ra những cặp JD và resume tốt. Đối với các JD tiếng Việt hầu hết sẽ cho ra kết quả resume tốt cũng là resume tiếng Việt.

Một điểm mạnh của mô hình là tốc độ xử nhanh, 1 JD cho ra kết quả chỉ tốn khoảng 1 giây. Tổng tất cả thời gian cho 897 JD nằm trong khoảng 15 - 17 phút. Nhược điểm của mô hình là xử lý các cặp JD và resume khác ngôn ngữ nhau có thể cho ra độ đo âm và cần tìm trọng số phù hợp để tính điểm $overall\_score$ tốt nhất.

4 Kết luận.

Qua quá trình làm báo cáo, nhóm tác giả đã đạt được một số kết quả như sau:

- Trình bày được khái quát lý thuyết của hệ thống khuyến nghị, các độ đo tương đồng, các kỹ thuật NLP sử dụng như NER, model SBERT.
- Thiết kế, xây dựng được mô hình hệ thống khuyến nghị cho lĩnh vực tuyển dụng.
- Tiến hành cài đặt, đánh giá thử nghiệm mô hình.

Tài liệu

[1] S. A. Alsaif, M. Sassi Hidri, I. Ferjani, H. A. Eleraky, and A. Hidri, "Nlp-based bi-directional recommendation system: Towards recommending jobs to job seekers and resumes to recruiters," *Big Data and Cognitive Computing*, vol. 6, no. 4, 2022. [Online]. Available: https://www.mdpi.com/2504-2289/6/4/147

[2] S. F. Shovon, K. T. J. Tama, J. Ferdaous, and M. A. B. Mohsin, "Cvr: An automated cv recommender system using machine learning techniques," in *Data Science and Algorithms in Systems*, 2023, pp. 312–325.

[3] T. K, U. V, S. M Kadiwal, and S. Revanna, "Design and development of machine learning based resume ranking system," *Global Transitions Proceedings*, vol. 3, pp. 371–375, 2022.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.

[5] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. [Online]. Available: http://arxiv.org/abs/1908.10084

[6] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*, 1st ed. USA: Cambridge University Press, 2010.

[7] B. Li and L. Han, "Distance weighted cosine similarity measure for text classification," in *Intelligent Data Engineering and Automated Learning – IDEAL 2013*, H. Yin, K. Tang, Y. Gao, F. Klawonn, M. Lee, T. Weise, B. Li, and X. Yao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 611–618.

[8] C. Sternitzke and I. Bergmann, "Similarity measures for document mapping: A comparative study on the level of an individual scientist," *Scientometrics*, vol. 78, pp. 113–130, 12 2009.

[9] W. Qader, M. M. Ameen, and B. Ahmed, "An overview of bag of words;importance, implementation, applications, and challenges," 06 2019, pp. 200–204.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[11] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *J. Am. Soc. Inf. Sci.*, vol. 41, pp. 391–407, 1990.

[12] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, no. null, p. 993–1022, mar 2003.

[13] A. Fantechi, S. Gnesi, S. Livi, and L. Semini, "A spacy-based tool for extracting variability from nl requirements," in *Proceedings of the 25th ACM International Systems and Software Product Line Conference - Volume B*, ser. SPLC '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 32–35. [Online]. Available: https://doi.org/10.1145/3461002.3473074

[14] S. Bag, S. K. Kumar, and M. K. Tiwari, "An efficient recommendation generation using relevant jaccard similarity," *Information Sciences*, vol. 483, pp. 53–64, 2019.

[15] S. Al-Otaibi, N. Altwoijry, A. Alqahtani, L. Aldheem, M. Alqhatani, N. Alsuraiby, S. Alsaif, and S. Albarrak, "Cosine similarity-based algorithm for social networking recommendation," *Int. J. Electr. Comput. Eng*, vol. 12, no. 2, pp. 1881–1892, 2022.

[16] "Paraphrase-minilm-l6-v2." [Online]. Available: https://huggingface.co/sentence-transformers/paraphrase-MiniLM-L6-v2