

HỆ THỐNG ĐÁNH GIÁ HỒ SƠ VIỆC LÀM DỰA TRÊN API MÔ HÌNH VÀ MULTI-AGENT

Huỳnh Nhật Minh - 22520862

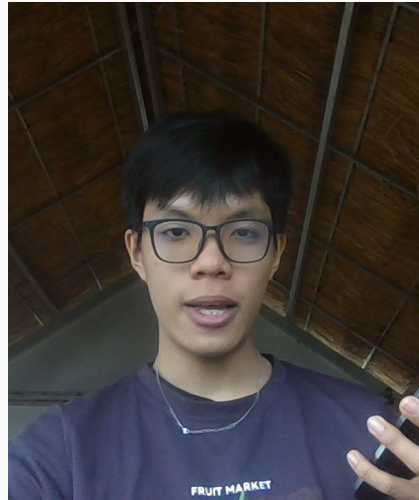
Lê Huỳnh Giang - 22520356

Tóm tắt

- Lớp: CS519.P11
- Link Github của nhóm: <https://github.com/lhgiang040504/JobHelper>
- Link YouTube video:
- Ảnh + Họ và Tên của các thành viên:



Huỳnh Nhật Minh
MSSV: 22520862



Lê Huỳnh Giang
MSSV: 22520356

Giới thiệu

Khảo sát hiện trạng:

Trong bối cảnh thị trường lao động ngày càng cạnh tranh, việc tuyển dụng hiệu quả là một thách thức lớn đối với các doanh nghiệp. Các nhà tuyển dụng thường phải xử lý một lượng lớn hồ sơ ứng viên để tìm kiếm người phù hợp với yêu cầu công việc (Job Description - JD). Tuy nhiên, phương pháp đánh giá thủ công không chỉ mất thời gian mà còn dễ dẫn đến sai sót, bỏ qua các ứng viên tiềm năng.

Mục tiêu

- Xây dựng một hệ thống có khả năng tự động nhận diện và trích xuất các đặc trưng quan trọng từ CV
- Phát triển thuật toán xử lý ngôn ngữ tự nhiên (NLP) để phân tích và trích xuất yêu cầu công việc, kỹ năng cần thiết, và các tiêu chí tuyển dụng từ JD.
- Thiết kế mô hình so sánh và đánh giá độ phù hợp giữa CV và JD
- Cùng với đó là hệ thống hỏi đáp giúp người dùng có thể tìm kiếm các thông tin

Nội dung và Phương pháp

- Frontend (Client-Side) là giao diện web tương tác, giúp:
 - Nhà tuyển dụng tải lên JD và nhận kết quả phân tích.
 - Ứng viên tải lên CV và nhận gợi ý cải thiện.
 - Công nghệ sử dụng: React.js
- Backend (Server-Side):
 - FastAPI làm framework chính để xây dựng các API RESTful cho hệ thống.
 - Tích hợp LLaMA để xử lý các tác vụ NLP (trích xuất thông tin từ CV và JD).

The screenshot shows a web application interface for 'My Application'. At the top, there are links for 'JOB MANAGEMENT' and 'CANDIDATE MANAGEMENT'. The main section is titled 'Job Management'. It features a form to 'Enter Job Description' with a text area containing the following text: 'AI Engineer', 'Kinh nghiệm: 2 năm.', 'Yêu cầu ứng viên: Tốt nghiệp Đại học chuyên ngành Công nghệ thông tin, Khoa học máy tính, Toán học ứng dụng hoặc các lĩnh vực liên quan.', and 'Có ít nhất 2 năm kinh nghiệm làm việc trong lĩnh vực Machine Learning và xử lý ngôn ngữ'. Below the text area are 'SAVE' and 'CANCEL' buttons. Below the form is a button labeled 'TOP 5 CANDIDATES'. To the right of the form, there is a section titled 'Top 5 Candidates' displaying a table of candidate information.

Nguyen Duy Tam Anh	Nguyen Duy Tam Anh	Nguyen Duy Tam Anh
Position: AI Engineer	Position: AI Engineer	Position: AI Engineer
Matching Skills: Machine Learning, PyTorch, Langchain	Matching Skills: Machine Learning, PyTorch, Langchain	Matching Skills: No matching skills
Degree Score: 1	Degree Score: 1	Degree Score: 1
Major Score: 1	Major Score: 1	Major Score: 1
Skill Score: 0.2727272727272727	Skill Score: 0.2727272727272727	Skill Score: 0
Total Score: 2.2727272727272725	Total Score: 2.2727272727272725	Total Score: 2
File: CV Nguyễn Duy Tâm Anh - AI Engineer Intern-TopCV.vn.pdf	File: CV Nguyễn Duy Tâm Anh - AI Engineer Intern-TopCV.vn.pdf	File: 4.pdf

The screenshot shows a chatbot interface. At the top, there is a blue header with the text 'Chatbot'. Below the header, there is a chat window. The chat window has a header with 'You' and a profile icon. The chat message from 'You' is 'how can improve my softskill?'. Below the chat message, there is a green box with the text 'Chatbot'. The chatbot response is 'To improve your soft skills, consider the following strategies: 1. Practice effective communication: Focus on clear and concise verbal and written communication, active listening, and asking clarifying questions. 2. Develop emotional intelligence: Recognize and manage your emotions, empathize with others, and build strong relationships. 3. Enhance teamwork and collaboration: Volunteer for group projects, participate in team-building activities, and learn to give and receive constructive feedback. 4. Improve time management and organization: Prioritize tasks, set realistic goals, and use tools like calendars or to-do lists to stay organized. 5. Seek feedback and learn from criticism: Ask for feedback from colleagues or mentors, reflect on areas for improvement, and work on implementing changes. Additionally, you can: - Take online courses or attend workshops to develop specific soft skills - Read books or articles on personal development and self-improvement - Join professional networks or clubs to practice networking and building relationships - Participate in volunteer work or community service to develop empathy and social skills Remember that improving soft skills takes time and practice. Be patient, persistent, and open to learning and growth.'

Nội dung và Phương pháp

- Model AI (LLaMA):
 - Dùng LLaMA để phân tích nội dung CV và JD, cụ thể:
 - Trích xuất thông tin quan trọng như kỹ năng, kinh nghiệm, học vấn.
- Database:
 - Lưu trữ thông tin người dùng, CV, JD, và kết quả phân tích.
 - Công nghệ sử dụng: MongoDB

```
1 FUNCTION calculate_degree_match(job_degree, candidate_degree):
2     IF job_degree IS A LIST:
3         RETURN 1.0 IF candidate_degree IS IN job_degree ELSE 0.0
4     ELSE:
5         RETURN 1.0 IF candidate_degree EQUALS job_degree ELSE 0.0
6
```

```
1 FUNCTION calculate_major_match(job_major, candidate_major):
2     candidate_major_lower = CONVERT candidate_major TO LOWERCASE
3
4     IF job_major IS A LIST:
5         IF candidate_major_lower IS IN job_major:
6             RETURN 1.0
7
8     FOR major IN job_major:
9         related_majors = LOOKUP major IN MAJOR_DICT
10        IF candidate_major_lower IS IN related_majors:
11            RETURN 0.5
12
13    RETURN 0.0
14 ELSE:
15    RETURN 1.0 IF candidate_major_lower EQUALS job_major ELSE 0.0
```

Nội dung và Phương pháp

```
1 FUNCTION calculate_match(job_skills, candidate_skills):
2     matched_skills = EMPTY_LIST
3     unmatched_skills = EMPTY_LIST
4
5     FOR skill IN candidate_skills:
6         IF skill IS IN job_skills:
7             ADD skill TO matched_skills
8         ELSE:
9             ADD skill TO unmatched_skills
10
11     INITIALIZE tfidf_vectorizer FOR job_skills + unmatched_skills
12     job_vectors = TRANSFORM job_skills USING tfidf_vectorizer
13     candidate_vectors = TRANSFORM unmatched_skills USING tfidf_vectorizer
```

```
14
15     similar_skills = EMPTY_LIST
16     FOR EACH candidate_vector IN candidate_vectors:
17         similarities = CALCULATE cosine_similarity(candidate_vector, job_vectors)
18         max_similarity = MAX(similarities) IF similarities IS NOT EMPTY ELSE 0
19         IF max_similarity > 0.7:
20             ADD corresponding unmatched_skill TO similar_skills
21
22     ADD similar_skills TO matched_skills
23     RETURN matched_skills
```

Nội dung và Phương pháp

Hệ Thống hỏi đáp: Áp dụng RAG cho hệ thống hỏi đáp

- LLM (Large Language Model):
 - Sử dụng mô hình tiên tiến từ Together AI meta-llama/Llama-3.3-70B-Instruct-Turbo để sinh nội dung.
- Dữ liệu (trending và mới nhất) Dữ liệu khóa học:
 - Từ Coursera, Udemy, LinkedIn Learning, v.v. Dữ liệu công việc: Từ LinkedIn, Glassdoor, hoặc các trang việc làm khác.
- Model embedding:
 - Sử dụng các model phổ biến và không cần quá phức tạp để tạo ra các vector embedding phục vụ cho việc truy vấn
- MongoDB:
 - Dữ liệu khóa học và công việc được lưu trữ dưới dạng JSON.
 - Vector embedding của dữ liệu được lưu trong một trường riêng (dùng vector search).
- Retriever:
 - Vector search dựa trên MongoDB Atlas Search hoặc tích hợp với thư viện FAISS để truy vấn nhanh.

Kết quả dự kiến

Hệ thống đề xuất các ứng viên tiềm năng dựa trên mức độ tương đồng giữa hồ sơ và yêu cầu công việc.

Cung cấp các khuyến nghị về khóa học hoặc kỹ năng cần cải thiện để tăng khả năng trúng tuyển của ứng viên.

Đưa ra các đánh giá phù hợp với tiêu chí tuyển dụng, hạn chế sai sót so với phương pháp thủ công.

Độ chính xác (Accuracy): Trên 80%

Hiệu suất:

Thời gian xử lý trung bình mỗi file (2 - 3 trang) : 3.0 giây / file.

Thời gian xử lý tìm kiếm CV phù hợp trung bình : 1.0 giây.

Tài liệu tham khảo

- [1]Brownlee, J. (2019). *Deep Learning for Natural Language Processing: Develop Deep Learning Models for Natural Language in Python*. Machine Learning Mastery.
- [2]Hugging Face Documentation. (n.d.). *Transformers Library*. Retrieved from <https://huggingface.co/docs>.
- [3]OpenAI API Documentation. (n.d.). Retrieved from <https://platform.openai.com/docs>.
- [4]Gupta, P., & Bhatia, R. (2020). *Job Recommendation System Using Natural Language Processing*. International Journal of Engineering Research & Technology.
- [5]JADE Documentation. (n.d.). Retrieved from <https://jade.tilab.com>.
- [6]European Parliament. (2016). *General Data Protection Regulation (GDPR)*. Retrieved from <https://gdpr-info.eu>.
- [7]Coursera API Documentation. (n.d.). Retrieved from <https://www.coursera.org/>.
- [8]SpaCy Documentation. (n.d.). *Industrial-Strength Natural Language Processing*. Retrieved from <https://spacy.io/>.
- [9]Microsoft Azure Cognitive Services. (n.d.). Retrieved from <https://azure.microsoft.com/en-us/services/cognitive-services/>.
- [10]MongoDB Documentation. (n.d.). *Building Scalable Database for HR Systems*. Retrieved from <https://www.mongodb.com/>.