RESUME-MATCHING SYSTEM

Report on resume matching with Job-Description

SEPTEMBER 17, 2023

NILESH NAYAK
Pune - 411025, India

Report: Resume Matching with Job Descriptions Using PDF CVs

Project Description:

The aim of this project is to develop a tool for resume parsing and cosine similarity calculation between skills and job-description using Python. The project involves the utilization of PDF extractor libraries like “pyresparser” in Python for parsing resumes, word embeddings through “DistilBERT” from Hugging Face's Transformers library, and calculating the cosine similarity between two embedded sentences using the Natural Language Toolkit (NLTK).

Objective:

1. **Resume Parsing**: Develop a script to extract relevant information from resumes in PDF format. This information may include but is not limited to:
    a. Job Role
    b. Education
    c. Skills
2. **Word Embeddings**: Implement word embeddings using DistilBERT, a state-of-the-art pre-trained transformer model from Hugging Face's Transformers library. This will convert words or sentences into numerical vectors that capture their semantic meaning.
3. **Cosine Similarity Calculation**: Calculate the cosine similarity between two embedded sentences using SciKit-Learn library to measure the similarity between Job-Description and resume skills. Cosine similarity is a common metric used in natural language processing to determine how similar two vectors are.

Dataset Description:

- Kaggle resume dataset is having 24 category and 1284 pdf resume.
- Hugging face job-description dataset has 853 job-description.

Tools/Techniques Used:

- **Python**: The primary programming language for this project.
- **PDF Extractor Libraries**: Utilize Python libraries such as PyPDF2 or pdfminer and ‘pyresparser’ for parsing PDF resumes.
- **Hugging Face Transformers**: Use the DistilBERT model for word embeddings.
- **NLTK**: Employ the Natural Language Toolkit for text pre-processing.
- **SciKit-Learn:** To calculate cosine-similarity between two embedded sentences vector.

Methodology:

Resume Parsing-

- **PDF Extraction:** PDF resumes will be processed using PDF extraction libraries. These libraries will extract text and structural information from the resumes.
- **Text Pre-processing:** The extracted text will undergo pre-processing to remove any irrelevant information, such as headers and footers, and to improve the quality of the extracted text.

- **Information Extraction:** Relevant information such as job-role, education, and skills will be extracted using pyresparser and text analysis techniques.

Word Embeddings -

- **Word Embedding with DistilBERT:** Implement DistilBERT from Hugging Face's Transformers library to convert words or sentences from skills and job-description text into numerical vectors. This will create embeddings that represent the semantic meaning of the text.

Cosine Similarity Calculation -

- **Cosine Similarity Calculation:** Calculate the cosine similarity between Skills and Job-Description embedded sentences using the SciKit-Learn library. Cosine similarity values range from -1 (completely dissimilar) to 1 (completely similar), with 0 indicating no similarity.

Result:

The project aims to produce the following results:

- Extracted and structured resume information, including Job-Role, education, and skills.
- Word embeddings for resume text and job-description data using DistilBERT.
- Cosine similarity scores for pairs of sentences or sections of the parsed resumes. This will provide a quantitative measure of similarity.

Challenges:

- Version Mismatch for pyresparser, PyPDF2 library.
- Listing of Skills to extract from resume.
- Aggregation of cosine-similarity matrix to get one value for score.
- Hardware limitations (requires GPU for faster calculations)

Conclusion:

This project combines the power of PDF extraction, word embeddings, and cosine similarity calculation to develop a resume parsing and matching tool that can assess the similarity between different job-description and skills of resumes. This tool can be valuable for various applications, such as candidate matching in recruitment processes and identifying skill gaps in job seekers' profiles. By successfully implementing these components, we have demonstrated the potential for automated resume analysis and similarity assessment with job-description.

Future Scope:

- Improve resume parsing accuracy by incorporating more advanced natural language processing techniques and named entity recognition.
- Extend the project to support a broader range of document formats beyond PDF.

- Enhance the efficiency of DistilBERT techniques to convert the given text into vector.
- Implement a database to store and retrieve parsed resume data for more comprehensive analysis and reporting.

Acknowledgements:

I would like to express my gratitude to the open-source community for their contributions to the Python libraries like PyPDF2, pyresparser, scikit-learn and tools that made this project possible. I also thank Hugging Face for providing access to state-of-the-art transformer models.

References:

- https://huggingface.co/docs/transformers/model_doc/distilbert
- https://www.scaler.com/topics/nlp/huggingface-transformers/
- https://omkarpathak.in/pyresparser/
- https://pypdf2.readthedocs.io/en/3.0.0/

Dataset links

- https://www.kaggle.com/datasets/snehaanbhawal/resume-dataset
- jacob-hugging-face/job-descriptions · Datasets at Hugging Face