Resume Extraction Project

Alexis Collier

10/22/2023

Agenda

Executive Summary

Problem Statement

Data Comprehension

Data Preparation

EDA

Modeling and Evaluation

Deployment

Conclusion

Executive summary:

The Client:
HR departements.

Data Set:
A json file containing 200 resumes.

General Description :
This is a flask web app used for resume annotation and parsing.

Product-Service Description:
This app is dedicated to HR managers to help them:

- Converting hours of labor into seconds.
- Increase recruiters' efficiency and availability.
- Reducing the need for more employees.
- Avoiding errors.

Data science objectives:

- Identifying the suitable technologies for our business objectives.
- Training and deploying fast and efficient Deep Learning models.

Problem statement :

Problem:

Resumes contain a surplus of information irrelevant to the HR/authority, and they have to manually process the resumes to shortlist the promising candidates for them. This EDA is dedicated to them to help them enhance their performance.

Data comprehension :

The data that has been provided is a json file. It’s composed of 200 resumes. Each line represents a 2 keys dictionary:
“annotation” is the key to the labeled resume.
“content” is the key to the plain resume text.

Each resume feature is represented with a dictionary: dict_keys(['label', 'points'])
‘points’ is the key to a dictionary that looks like this: [{'start': 1749, 'end': 1754, 'text': 'Oracle'}]

Resume.json

Raw

This file has been truncated, but you can view the full file.


```
{"content": "Govardhana K\nSenior Software Engineer\n\nBengaluru, Karnataka, Karnataka - Email me on Indeed: indeed.com/r/Govardhana-K/\nb2
{"content": "Harini Komaravelli\nTest Analyst at Oracle, Hyderabad\n\nHyderabad, Telangana - Email me on Indeed: indeed.com/r/Harini-\nKoma
{"content": "Hartej Kathuria\nData Analyst Intern - Oracle Retail\n\nBengaluru, Karnataka - Email me on Indeed: indeed.com/r/Hartej-Kathuri
{"content": "Ijas Nizamuddin\nAssociate Consultant - State Street\n\nIrinchayam B.O, Kerala - Email me on Indeed: indeed.com/r/Ijas-\nNizam
{"content": "Imgeeyaul Ansari\njava developer\n\nPune, Maharashtra - Email me on Indeed: indeed.com/r/Imgeeyaul-Ansari/a7be1cc43a434ac4\n\n
{"content": "Jay Madhavi\nNavi Mumbai, Maharashtra - Email me on Indeed: indeed.com/r/Jay-\nMadhavi/1e7d0305af766bf6\n\nI look forward to b
{"content": "Jitendra Babu\nFI/CO Consultant in Tech Mahindra - SAP FICO\n\nChennai, Tamil Nadu - Email me on Indeed: indeed.com/r/Jitendra
{"content": "Jyotirbindu Patnaik\nAssociate consultant@SAP labs , Bangalore Karnataka\n\nBengaluru, Karnataka - Email me on Indeed: indeed.
{"content": "Karthihayini C\nSystems Engineer - Infosys Limited\n\nRajapalaiyam, Tamil Nadu - Email me on Indeed: indeed.com/r/Karthihayini
{"content": "Karthik GV\nArchitect - Microsoft India\n\nHyderabad, Telangana - Email me on Indeed: indeed.com/r/Karthik-GV/1961c4eff806e6f4
{"content": "Kartik Sharma\nSystems Engineer - Infosys Ltd\n\nDelhi, Delhi - Email me on Indeed: indeed.com/r/Kartik-Sharma/cc7951fd7809f35
{"content": "Kasturika Borah\nTeam Member - Cisco\n\nBengaluru, Karnataka - Email me on Indeed: indeed.com/r/Kasturika-\nBorah/9e71468914b3
{"content": "Kavitha K\nSenior System Engineer - Infosys Limited\n\nSalem, Tamil Nadu - Email me on Indeed: indeed.com/r/Kavitha-K/8977ce8c
```

Data preparation :

Steps:

- Importing the data and transforming it into a dataframe.
- Getting rid of the trailing white spaces from the plain text resumes
- Putting entities of the annotation into lists composed of label, starting position , ending position.
- Getting rid of trailing spaces and unnecessary punctuation from the entities .
- Tokenization.
- Splitting the data into testing and training sets.
- Transforming the data to be fed to the model to torch tensors.

```
[ ] for i in range(len(data)):
      tokenized_texts = tokenizer.texts_to_sequences(data["content"])

    print(tokenized_texts[0])
```

```
[1979, 893, 187, 49, 60, 168, 73, 73, 62, 57, 8, 11, 11, 13, 16, 1979, 893, 3170, 569,
```

EDA

Analysis plan:

- Graduation year analysis.
- Candidates' location analysis.
- Companies Candidates worked at.
- Candidates’ degrees and universities studied at analysis.

EDA summary:

- The EDA performed on our data has shown that the candidates who have applied are quite different; they don’t share the same field of study or degrees. They went to different colleges, and they didn’t have the same skills.
- Depending on these insights, the client’s HR department can request the elimination of some candidates’ categories depending on the job profile needed (for example, the client needs candidates with engineering degrees)

Recommendation:

This EDA has shown that those who applied for this job are quite different, especially when talking about the fields of study I recommend that the HR department takes more care of the job description and the requirements provided to have a more accurate candidates’ resumes.

Modeling and evaluation:

Depending on the data we have, our business, and data science objectives, I will use the BERT model to work with to build the resume extraction.

Bidirectional Encoder Representations from Transformers (BERT) is a transformer-based machine-learning technique for natural language processing (NLP) pre-training developed by Google.
Epochs: 6
Optimizer:
Adam

As we can see, the accuracy we had is: 0.9138

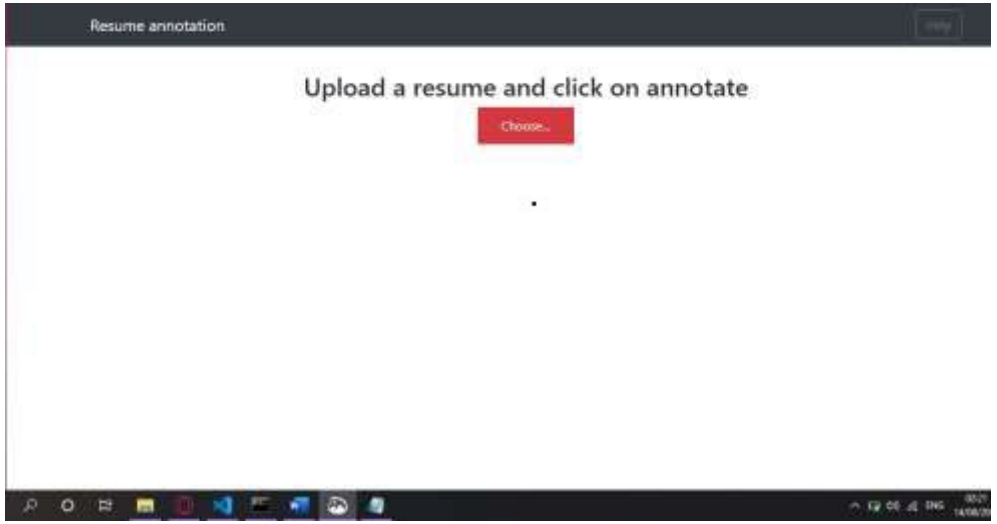
F1- score: 0.94

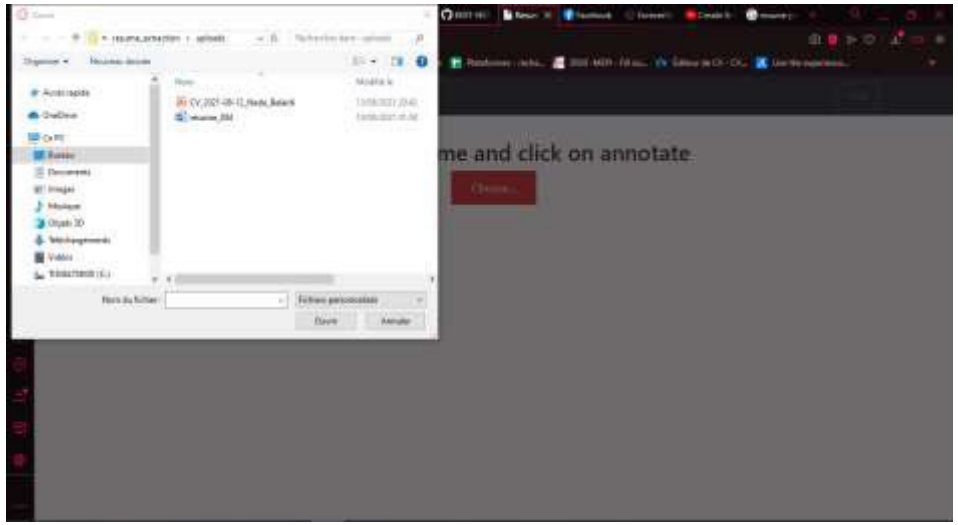
```
Epoch:   0%|          | 0/5 [00:00<?, ?it/s]Train loss: 0.8996409103274345
Epoch:  20%|██        | 1/5 [00:11<00:46, 11.65s/it]Validation loss: 0.5537128895521164
Validation Accuracy: 0.9138541666666666
F1-Score: 0.9417320178767181
Train loss: 0.5108052641153336
Epoch:  40%|████      | 2/5 [00:23<00:35, 11.67s/it]Validation loss: 0.517583355307579
Validation Accuracy: 0.9138541666666666
F1-Score: 0.9417320178767181
Train loss: 0.4752659574151039
Epoch:  60%|██████    | 3/5 [00:35<00:23, 11.75s/it]Validation loss: 0.4547186344861984
Validation Accuracy: 0.9138541666666666
F1-Score: 0.9417320178767181
Train loss: 0.4253872831662496
Epoch:  80%|████████  | 4/5 [00:47<00:11, 11.83s/it]Validation loss: 0.42342111468315125
Validation Accuracy: 0.9171875
F1-Score: 0.9430812041487477
Train loss: 0.40889669706424076
Epoch: 100%|██████████| 5/5 [00:59<00:00, 11.90s/it]Validation loss: 0.4165296256542206
Validation Accuracy: 0.9138541666666666
F1-Score: 0.9417320178767181
```

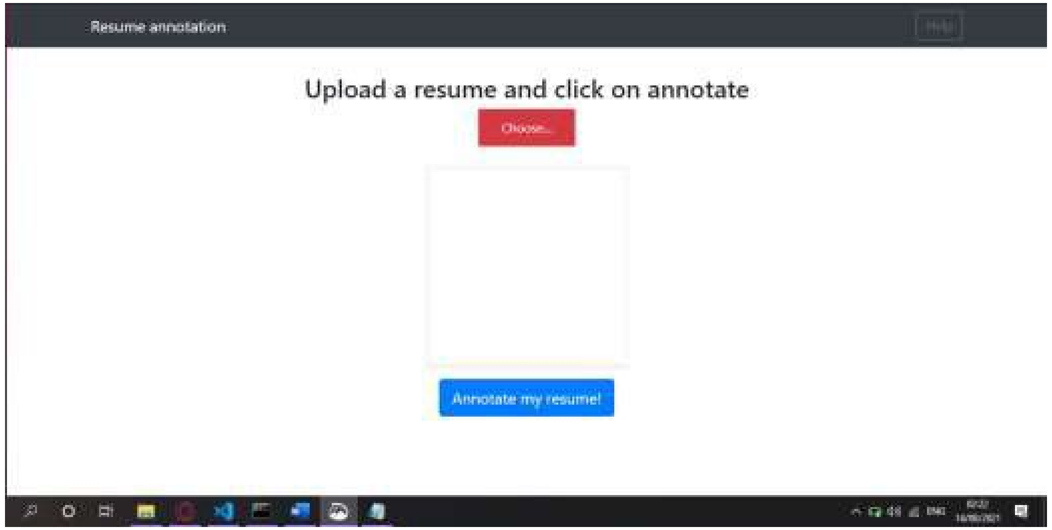
Deployement:

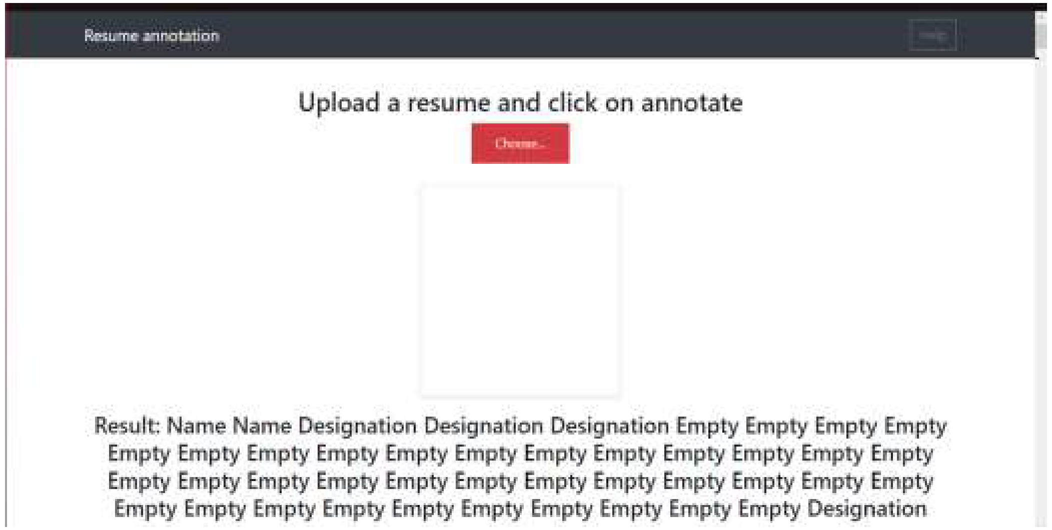
I used Flask, an open-source micro framework for web development in Python.
I started with creating additional functions dealing with mining text from files in different.

Formats (pdf, docx, and json)
I used HTML, CSS, and JS for the front-end part.
This is a quick demo:






Resume annotation
Upload a resume and click on annotate
Choose...
Annotate my resume!


Resume annotation
Upload a resume and click on annotate
Choose...
Result: Name Name Designation Designation Designation Empty Designation

Conclusion:

This was the NLP project that I worked on while being an intern at Data Glacier for one month.
This is an app that will make it easier for HR employers to find the perfect candidate for the job in less time while avoiding errors and with fewer employees' needs.

GitHub Repo link: https://github.com/colla00/NLP-Resume-Extraction-Project

Thank You