Mid-Term Report

1. Task Description

The soul aim of my Machine Learning project is to build a candidate selection model.The model will shortlist most appropriate candidates for a particular Job Description among a huge collection of candidates.It is a Supervised Learning Model and as per my research my model comes under the NLP(Natural Language Processing) domain of Machine Learning.My model utilises various NLP techniques ranging from basic NLP techniques for text preprocessing to advanced NLP techniques for generating embeddings. I will use NLP techniques such as Stop Word Removal,tokenization,lemmatization for processing the input.I will make use of mainly SpaCy to implement these.A large part of my model is based on the Named Entity Recognition(NER) which is used for extracting important fields(which earlier i was planning to do through RegEx) from Resumes and JDs which will make use of SpaCy's blank English model for implementation.As of now I will use the Doc2Vec approach(which is an extension of the famous Word2vec developed by Google) for generating embeddings.But I am also planning to use the transformer based pre trained BERT mode for embeddings due to its accuracy(fun fact: It was also developed Google).Earlier I was determined to use TF-IDF for generating embeddings but as suggested by my mentors and also during the learning phase I got to know about Word2Vec and dropped the idea of using it.The last part of my model includes generating similarity scores and i am planning to use cosine similarity(this might be subject to change).

---

2. Dataset

I had to divide my training dataset into two parts,one for training the Resume Screener and other one to train the embeddings model which is currently based on Doc2Vec.

- **Dataset 1:** The dataset which I am using to train the resume screener model was quite easily accessible. I basically combined two datasets of pre annotated resumes.One of 220 resumes which i took from Kaggle,and the other one of 200 resumes was from a GitHub Repo.I am currently searching for a pre annotated Job Descriptions dataset on various sites to make my overall training dataset comprehensive and diverse.I had to convert both the sets into a common JSON format and then fed it to my model as pickle file.

- **Dataset 2:** Now the dataset which I am using to train the Doc2Vec model has to first go through the Resume Screener which will give the extracted entities of the resumes in a json format which will be then fed to Doc2Vec model for training.This dataset includes a compilation of various resumes and JDs from kaggle,indeed.com,enhancv.com and various other sites.All the resumes and JDs will be first parsed by a PDF parser to

extract all the text from pdf,docx files and then given to the resume Screener.This Dataset currently includes 100 Resumes and around 70 JDs.

One of the major challenges faced was during the time of training the resume screener.Earlier my dataset for this consisted only of 200 resumes which probably resulted in overfitting.Upon discussing with mentors I decided make my dataset more vast and diverse and also tuned some of the parameters to improve my model's performance.

---

3. Model Architecture

My model basically comprises of three parts first part being the Resume Screener(trained using spacy's blank english model) it includes training a custom NER model using a pre annotated dataset and will make use of the NER technique to extract important fields from resumes and JDs like Name,Experience,Skills,Degree,etc.Second being the Embeddings model(trained using Doc2Vec) Doc2Vec model is known for generating embeddings for the entire document ,I am also planning to use a Pre Trained transformer based model BERT wich will provide me with more rich and contextualised embeddings which would improve the overall accuracy greatly, and third includes calculating the similarity score between resumes and JDs and ranking resumes accordingly.The third part so far will use the basic cosine similarity on Embeddings generated by the second model to calculate the similarity scores.

- **Input**: Input of the model would include Job Descriptions and Resumes mainly in pdf,doc,txt format.
- **Processing Stages**: The input data will be first parsed through a PDF parser(currently I am using PyMuPDF) for extracting all the text from resumes and JDs this text will be then fed to the resume screener to extract all the important fields of the input data in a JSON format.This parsed data will go through basic text processing which includes stop word removal,tokenization,lemmatization and converting to lower case before going into the Doc2Vec model.
- **Output**: The model is expected to produce a list of resumes ranked on the basis of their similarity scores with the provided Job Description.

---

4. Implementation

I have completed the learning part which includes learning the basic NLP techniques as well as advanced NLP topics such as RNNs,LSTMs,Transformers and learning in detail about BERT. I am currently implementing the Resume Screener model .My key focus is improving the accuracy of the Resume Screener(since accuracy of subsequent parts of the pipeline is dependent on how accurate my resume screener is) which takes most of my time.Earlier my model was overfitted on the training data therefore I had to work on my training data making it more diverse I also have a backup option to use the “Bert-base NER” model provided by Hugging Face incase my current model is not giving reliable results.I have prepared the code of the Doc2Vec model and will start its training once my screener model is trained.

Here's a link to a GitHub Repo which includes all the learning and implementation codes so far.

---

5. End-Term Goals

My updated End-Term goals will include building the Doc2Vec as well BERT based model for generating the word embeddings.I have dropped the idea of using TF-IDF for generating embeddings.Since my Resume Screener model is not completed yet i would also count training and fine tuning my resume screener as an End-Term Goal,and if the Spacy’s NER model doesn't work well for it I would switch to BERT- Base NER as mentioned earlier.I would also like to research more on a similarity technique another cosine similarity for comparing JDs and resumes.Learn more about Bagging as suggested by my mentor.

---

6. References

- Completed Andrew Ng’s ML Specialization course on Coursera to get into Machine Learning.
- YouTube Playlist. I have used this YouTube playlist to understand the basic concepts of NLP like: RegEx,Stemming,Tokenization,Word Embeddings ,etc.
- Read this article on PDF parsing which mainly talks about PyPDF2 but since I wanted to parse not only PDFs but also doc and other files so I decided to use PyMuPDF.
- I used this article for learning the implementation of PyMuPDF.
- I mainly followed the YouTube videos of 3blue1brown for understandingDeep Learning and advanced NLP concepts of RNN,LSTMs,Attention,Transformers,BERT.
- 3B1B Deep Learning playlist for Neural Networks.
- I watched CodeBasics yt videos on Word2Vec to understand the underlying concept and also its implementation. Vid1 , Vid2
- Watched this video for the implementation of BERT.

- Read an article provided by my mentor which talks about Doc2Vec and uses it in a similar way as i am going to use it for resume matching.
- Watched this yt video to understand Doc2Vec(PV-DM and PV-DBOW) and also for the implementation.
- Used SpaCy's documentation for learning how to add custom NER labels to a blank Spacy pipeline.
