CSCE 5214: Software development for AI, Fall 2022

Resume Classification Using NLP

Project Proposal

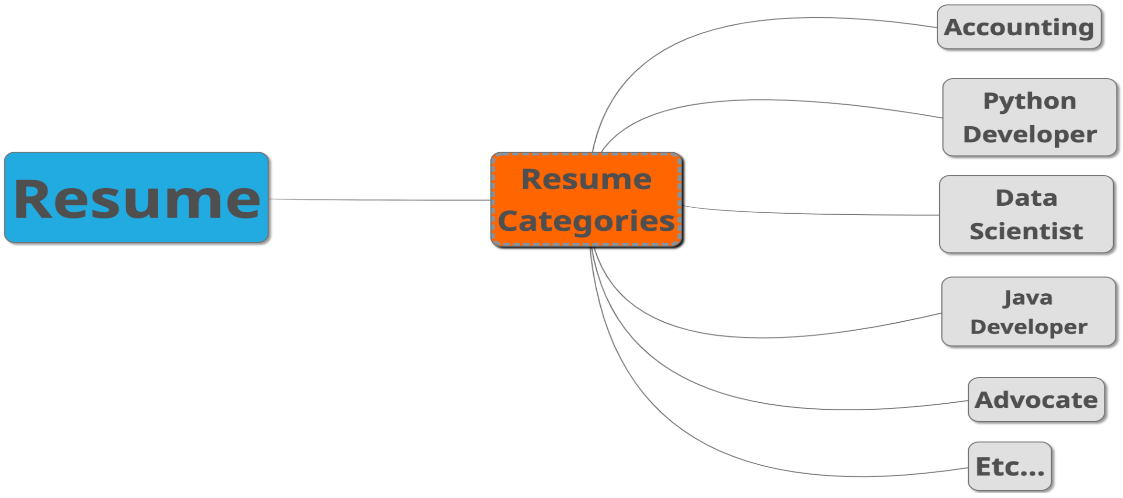
Instructor: *Dr. Russel Pears* (Russel.Pears@unt.edu)

Team Members:

Bhashwitha Kolluri - 11526264 (bhashwithakolluri@my.unt.edu)
Harish Kashyap Vutukuru - 11518964 (harishkashyapvutukuru@my.unt.edu)
Manoj Kolluri - 11524958 (manojkolluri@my.unt.edu)
Santhoshi Kareddy - 11600006 (SanthoshiKareddyVIII@my.unt.edu)

TABLE OF CONTENT:

Abstract



A typical job opening on the Internet receives many applications in a short period of time. The process of manually screening resumes is impractical because it takes a considerable amount of time and incurs a lot of expenditures that recruitment organizations cannot afford to incur. Furthermore, there are many qualified candidates who do not receive the attention they need because of the process of evaluating resumes. There is a possibility that unsuitable applicants will be hired, or that appropriate candidates may be rejected because of this. Our project introduces a system that is aimed at addressing these challenges by automatically recommending the best candidates based on the job description to address these challenges.

As part of our project, we use Natural Language Processing to extract key information from resumes, such as skills, education, experience, and so on, to create a simplified version of each application based on the extracted information. Recruiters will be able to properly analyze each resume in less time when all irrelevant information is removed, simplifying the screening process, and making the screening process simpler. To match the resumes with the job description after the text mining process has been completed, the method applies a vectorization model and a cosine similarity model. It is then possible to use the derived ranking scores to identify the most suitable candidates for the job, based on their fit with that role.

Motivation & Significance

An applicant's CV is provided to recruiters in the form of a PDF or a Word document. There is nothing difficult about reading tone resumes in these formats, but it is a challenge if the recruiter receives dozens of fresh resumes every day, and it is hard to keep up with this volume. That is when the use of Natural Language Processing (NLP) technology in resume classification comes into play. We offer resume categorizer solutions that can save recruiters time and effort by automating the process of processing CVs by hand. There is a tool called a resume categorizer that uses artificial intelligence to recognize and extract information from CVs in different formats, and it provides the information to you in an organized and understandable manner based on this information.

Dataset

For this project, we used the Kaggle Resume Dataset [4]

This dataset contains a total of 962 Resumes belonging to 25 different categories.

Some of those categories are:

1. **Java Developer,**
2. **Testing,**
3. **DevOps Engineer,**
4. **Python Developer,**
5. **Web Designing and so on.**

For each category of the resume, there are an average of 40 to 50 resumes belonging to the same category.

Learning Goals & Objective

The major objective and learning goal of our application are to create a fully connected AI model that can learn and recognize which job category the resumes belong to base on the words and the context written in the resume by the applicant and categorize it accordingly with optimal accuracy. We then plan to leverage this model by trying to categorize an entirely new resume that is uploaded by the user and estimate the job category the resume belongs to. Since we also plan to show the user the working functionality of such an application, we also intend to develop a friendly user interface. The dataset mentioned above seems to be almost ideal for this task of multiclass classification.

Goals:

1. Cleaning and changing the text into a vectorized format which can be understood by the learning algorithm.
2. Creating a Fully trained model which can learn from this vectorized data and predict the label.
3. Integrating the model within a GUI to display the predictions and metrics of the evaluated model when applied in real time.

Project Design & Milestones

For this project, we propose to achieve our goals and milestones mentioned below by using Natural Language Processing techniques and Python Programming Language in a local environment using Anaconda Navigator and Jupiter Notebook. For the first part of the project, we plan to work on the data analysis and pre-processing parts, we plan to perform Exploratory Data Analysis on the raw data first and observe the most common words preferred in each category of the resume then we would perform an analysis on how evenly the dataset is distributed into all its categories. We plan to achieve this using python's NumPy, pandas, matplotlib, and seaborn libraries after the initial analysis is done the next step is to perform data pre-processing, for this part of the project we plan to implement techniques such as stop words removal, lemmatization, and stemming using the NLTK library. After the pre-processing part of the project is done, we then plan to change the cleaned data into vector form and then create a simple neural network using TensorFlow and train it on the dataset, we then need to Evaluate the performance metrics of the model and integrate it into a user-friendly Interface.

Milestones:

1. Perform EDA on the Dataset
2. Perform Preprocessing techniques to clean and balance the dataset if not balanced
3. Using the NLTK library to apply NLP techniques to the data such as Lemmatization or Stemming
4. Convert the data into Vector Format.
5. Split the data into Training and Testing formats
6. Create a Fully connected Neural Network
7. Perform Hyper Parameter Tuning
8. Train and test the model on the dataset and evaluate it using evaluation metrics.
9. integrate the model within a GUI

Requirements

When it comes to selecting a candidate for any position, one of the most crucial requirements is a resume. When the company's recruiting team gets a candidate's resume, the skills, if extracted via automation, will save the hiring team a significant amount of time because they will no longer need to sit and go through each word. However, first, a dataset must be scraped or created, and then preprocessing must be performed on that dataset. Once the words have been preprocessed, the data must be sorted into distinct job profile classes based on the expertise using various machine learning methods.

In terms of software and packages, the following are required:

Cloud tools: Google Cloud Platform

Jupyter/Colab notebooks: Jupyter

Mobile/client-side integration: client-side integration

- Python 3.6 or above
- Pandas for creating data frames and data analysis
- Numpy for an n-dimensional array
- Seaborn for visualization
- Sklearn for ML libraries
- Keras for creating deep models
- TensorFlow

Exploratory and Extensible

In this project, we not only propose to use simple machine learning to classify the resumes into different categories we also propose to experiment with different types of learning algorithms and neural networks to achieve this prediction and compare the performance metrics of these different algorithms against each other and integrate the Neural network with the best metrics into a user-friendly interface. This type of analysis and training process can further be used to develop other applications such as analyzing the quality of the resume or identifying the similarity between the candidate's skills and the job profile the candidate is applying for.

Literature Review

1. **Domain Adaptation for Resume Classification Using Convolutional Neural Networks** [1]

This article proposes a convolutional neural network method for categorizing resumes into 27 different job categories. Due to resume data's sensitive nature, domain adaptation provides a cost-effective and reliable solution. Our approach involves training a classifier on a variety of freely available job descriptions and then using it to classify resume data. Despite a limited amount of labelled resume data, we demonstrate a reasonable classification performance for our approach.

2. **Robotic Process Automation for Resume Processing System** [2]

In this paper, the author uses robotic process automation and machine learning techniques to automate such interview processes to help with recruitment. Whenever a new email or file is received, the RPA bot downloads the attachments, and then classifies the attachments to determine if they are resumes. The resumes are shortlisted according to their skills, experience, and educational background, and named entity recognition (NER) is applied to extract the pertinent details from the attachments. Naive Bayes Classifier has a precision value of 92.08% on text classification, 91.36% on document classification on BERT, 86.2 % on custom-trained NER, and 91.5 % on web scanning.

3. **Resume classification system using natural language processing and machine learning techniques** [3]

In the paper using machine learning algorithms and natural language processing techniques, the author proposes a solution with better accuracy and reliability in various settings. For demonstrating the importance of NLP and machine learning techniques in RCS, nine ML classification models were applied to evaluate the extracted features. These models included Support Vector Machine, Naive Bayes, KNN, and Logistic Regression. Confusion Matrix, F-Score, Accuracy, Precision, and Recall were calculated to assess the developed models. On the study dataset of over nine hundred sixty-plus parsed resumes, the SVM class of Machine Learning classifiers performed better with an accuracy of 96% than the rest classifiers using the One-vs-Rest-Classification strategy. This study demonstrated that NLP and Machine Learning can be used to develop an efficient RCS using promising results.

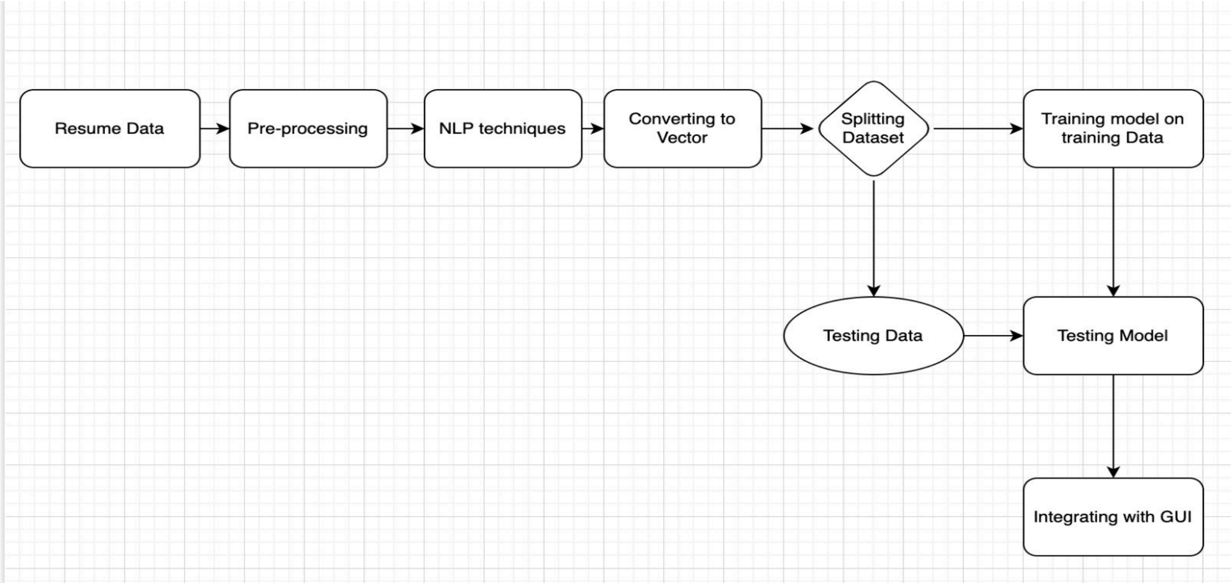
Workflow

EDA, Pre-Processing, Model Training:

- In the first stage, an Exploratory Data Analysis is performed on the dataset and the dataset is also checked for any missing data and imbalance in data distribution.
- In the initial preprocessing phase, methods like sampling are used if the dataset is imbalanced and any null values if present are removed.
- In the next phase of preprocessing NLP techniques such as stop words removal and other techniques such as stemming, and lemmatization are used before converting it into vector form.
- A record will be kept of the standard metrics of performance, such as the confusion matrix, the ROC, F1-score, precision, sensitivity, and accuracy to evaluate the models.

UI Integration:

- When a user uploads the resume all the important preprocessing and NLP techniques are applied in the backend.
- Also, the content of the resume then goes through a fully trained model and predicts its result.
- The UI then displays the result of the resume on the front end.



Expected Outcome

In the end, our project is going to result in the creation of a Graphical User Interface that takes a resume belonging to one of the 25 job categories mentioned earlier and feeds it into a pre-trained artificial intelligence model that predicts the job title for it.

References

1. Sayfullina, L., Malmi, E., Liao, Y., & Jung, A. (2017). Domain adaptation for resume classification using convolutional neural networks. Retrieved from https://arxiv.org/abs/1707.05576

2. Roopesh, N., & Babu, C. N. (Aug 27, 2021). Robotic process automation for the resume processing system. Paper presented at the 180-184. doi:10.1109/RTEICT52294.2021.9573595 Retrieved from https://ieeexplore.ieee.org/document/9573595

3. Ali, I., Mughal, N., Khan, Z. H., Ahmed, J., & Mujtaba, G. (2022). Resume classification system using natural language processing and machine learning techniques. Mehran University Research Journal of Engineering and Technology, 41(1), 65-79. doi:10.22581/muet1982.2201.07

4. https://www.kaggle.com/datasets/gauravduttakiit/resume-dataset