

Project Report on

**“Resume Categorization System Using Natural Language Processing (NLP)”**

Submitted in fulfilment of the requirement for the award of the internship certificate

as

## “Data Scientist Intern”

Submitted by

### Mr. THILAK RAJU S

### Ms. ANKITA BIRARI

### Mr. SUDEEP

Under the guidance of

## Mrs. Aishwarya Mate

## Ai Variant – Mentor

## Certificate Code – AIV/23-24/Q3/09/13432

## #49, Ground Floor, 27th Main Near IQRA International School, opposite to WIF Hospital, 1st Stage, BTM Layout, Bengaluru, Karnataka 560068

## ABSTRACT

The resume categorization solution should significantly reduce the manual human effort in the HRM. It should achieve a higher level of accuracy and automation with minimal human intervention.

**INTRODUCTION**

The selection of a suitable job applicant from the pool of thousands of applications is often daunting job for an employer. The categorization of job applications submitted in form of Resumes against available vacancy(s) takes significant time and efforts of an employer. Thus, Resume Classification System (RCS) using the Natural Language Processing (NLP) and Machine Learning (ML) techniques could automate this tedious process. Moreover, the automation of this process can significantly expedite and transparent the applicants' screening process with mere human involvement. This experimental study presents an automated NLP and ML-based RCS that classifies the Resumes according to job categories with performance guarantees.

To demonstrate the significance of NLP techniques for RCS, the extracted features were evaluated on ML classification models namely Support Vector Machine - SVM, Naïve Bayes K-Nearest Neighbor (KNN), and Logistic Regression (LR). The Term-Frequency-Inverse-Document-Frequency (TF-IDF) feature representation scheme was proved suitable for RCS. The developed models were evaluated using the Confusion Matrix, F-Score, Recall, Precision, and overall Accuracy.

**Project Workflow**

1. Business Objective
2. Data Collection
3. Data Preparation
4. Exploratory Data Analysis
5. Model Building
6. Model Evaluation
7. Deployment

**Data Collection**

The first step is to collect a dataset of resumes. Resume files can be of any format like .pdf, .doc, .docx.

**Data Preparation**

Data preparation involved conversion of .pdf and .doc files into .docx files. Used textract module to extract the text from documents. Saved the file name, category, and text into data frame.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Exploratory Data Analysis**

EDA involves –

* Text Pre-processing - removal of spaces, special characters, stop words, Pos-tagging etc.

A screenshot of a computer program

Description automatically generated

**Categorizing Resumes**

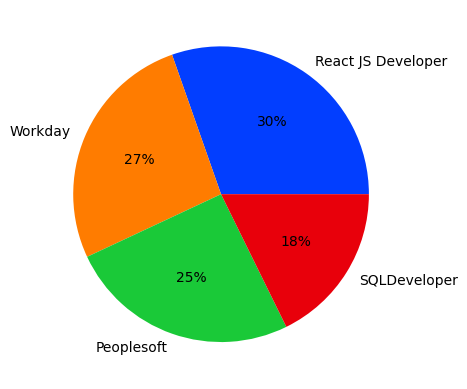
A screenshot of a computer

Description automatically generated

**Visualizations performed on the dataset**

1. Bar Plot
2. Pie Chart
3. Count-Vectorizer (Bigrams & Tri-Grams)
4. Word Cloud

A graph of a bar graph

Description automatically generated with medium confidence 

A graph of different colored bars

Description automatically generated with medium confidence A graph of different colored bars

Description automatically generated with medium confidence



**Model Building**

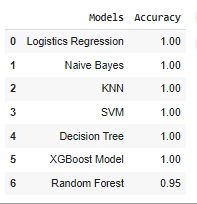
1. Logistics Regression
2. Naive Bayes Algorithm
3. K-Nearest Neighbors Classification
4. Support Vector Machine
5. Decision Tree
6. Random Forest
7. XG-Boost Classification

**Decision Tree Model**

**A screenshot of a computer program

Description automatically generated**

**Model Accuracy Table**



**Final Model Selection**

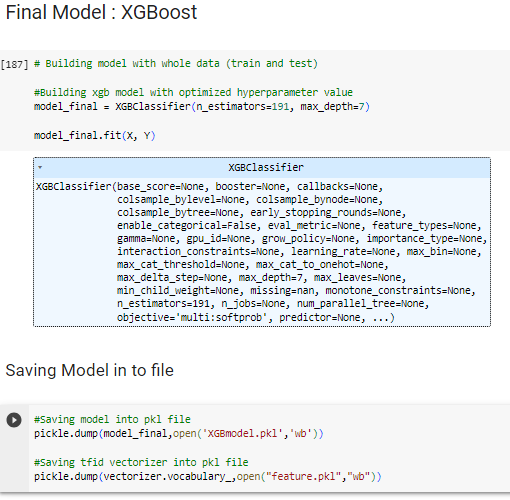
Final model is selected based on the accuracy of the models. From the table multiple models have performed well and has an accuracy of 100%. Out of the models XG-Boost model is selected. short for eXtreme Gradient Boosting, is an ensemble machine learning technique that combines multiple weak predictive models, called decision trees, to create a stronger and more accurate model. It iteratively trains these decision trees by focusing on the errors made by the previous trees in the sequence, effectively minimizing the residual errors.

**Advantages over other models:**

* Exceptional Performance
* Handling Sparse Data and Missing Values
* Regularization
* Parallel and Distributed Computing
* Flexibility and Customization

**XGBoost Model**

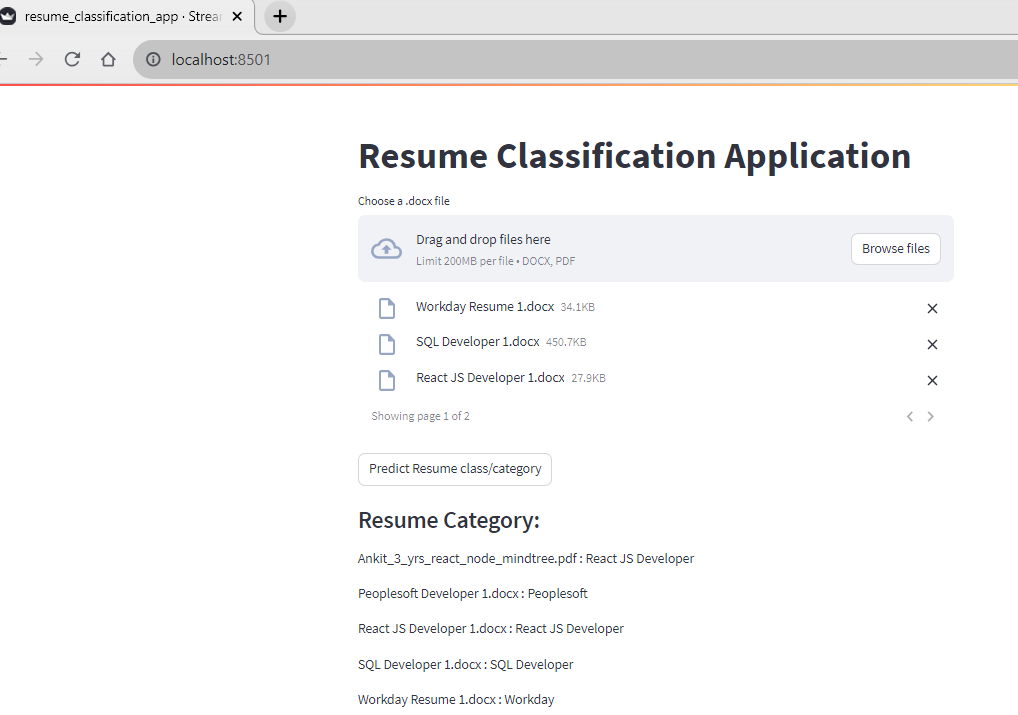
* Trained the model with whole data.
* Saved the model in Pickle file.
* Saved the TfidVectorizer.



**Deployment**

* Created .py file in Spyder for deployment.
* Loaded model and Vectorizer from pickle file.
* Streamlit Library for UI elements.
* file uploader object to upload resume.
* Displayed predicted Category.

**Resume Categorization System Application**

****

**Conclusion**

Multiple categorical models were built among them XG-Boost model is selected as it has a model accuracy of 100% and has an upper hand when compared to other models and performed very well. The data is trained on XG-Boost model and model is loaded into pickle file. Trained model is deployed using Stream-lit Library for UI elements and an application is created for Resume categorization system. The resumes are dropped or uploaded into the application, the application categorizes accordingly, and the result is provided with almost 100% accuracy.

**References and Supporting File Links**

Introduction:[http://ui.adsabs.harvard.edu/abs/2022MURJE..41...65A/abstract#:~:text=Thus%2C%20Resume%20Classification%20System%20(RCS,process%20with%20mere%20human%20involvement.](http://ui.adsabs.harvard.edu/abs/2022MURJE..41...65A/abstract%23:~:text=Thus%2C%20Resume%20Classification%20System%20(RCS,process%20with%20mere%20human%20involvement.%20)

GitHub Link : <https://github.com/Thilakraju/Ai-Variant-Intern-NLP-Project>