

Resume Analyzer Using Machine Learning

Assistant Professor T.Narmadha
Dept. of Computer Science and
Engineering
Jain (Deemed-to-be)University
Bangalore, Karnataka, India
Narmadha.t@jainuniversity.ac.in

D Sai Chandu
Dept. of Computer Science and
Engineering
Jain (Deemed-to-be)University
Bangalore, Karnataka, India
dsaichandu17@gmail.com

Korukanti Nihal
Dept. of Computer Science and
Engineering
Jain (Deemed-to-be)University
Bangalore, Karnataka, India
korukantinihal964@gmail.com

Ananthapalli Ihyth
Dept. of Computer Science and
Engineering
Jain (Deemed-to-be)University
Bangalore, Karnataka, India
ananthapalli.ihyth@gmail.com

Anvith Reddy Chamala
Dept. of Computer Science and
Engineering
Jain (Deemed-to-be)University
Bangalore, Karnataka, India
anvithreddy1@gmail.com

Abstract— The evaluation of resumes is a crucial step in the hiring process because it helps employers to quickly and accurately determine which applicants are the most suitable for a position. Natural language processing (NLP), a rapidly developing subject, uses machine learning algorithms to analyze and comprehend human dialogue. The goal of this research, Resume analyzer using Machine Learning is to evaluate any applicant's resume. Based on the details listed on the resume, our system will assign customers a resume score. All that must be uploaded is the user's resume. The resume will be automatically evaluated. The score you receive will then depend on how well your resume matches the criteria for the position you are applying for. This system is accessible to all users, whether they are experts, graduates, or novices. In the long run, we want to build a module particularly for businesses that will allow them to use our product to directly shortlist candidates using NLP. When assessing the resume, we are using the fundamentals of natural language processing. Currently this project will mainly focus on evaluating IT based resumes with high accuracy.

Keywords—K-Nearest Neighbor, Natural language Processing

I. INTRODUCTION

Resume analysis is a typical step in the hiring process because of the large number of applications that organizations frequently receive for each open position. This can make it difficult to quickly and accurately assess each candidate's qualifications and appropriateness. To solve this issue, several firms have turned to natural language processing (NLP) techniques, which use machine learning algorithms to analyze and comprehend human language. Among the many fields where NLP has applications are text classification, sentiment analysis, and natural language understanding. NLP can be applied to resume analysis in order to automatically extract important data from resumes, such as job titles, skills, and experience, and to compare this data with the requirements of a particular job opening. This can help businesses quickly identify the most qualified candidates and make smarter hiring decisions. With the aid of this

initiative, we hope to apply ML to a number of professions. Everyone can utilize this system, whether they are a novice user, a graduate, or an expert. In the long run, we hope to create a similar module with businesses in mind, enabling them to use this application to directly shortlist candidates using ML. We are putting the basic principles of natural language processing to use in evaluating the résumé. The use of artificial intelligence in the employment market is just getting started.

II. LITERATURE SURVEY

P. S. Gawande and S. S. Kulkarni [1] This research paper proposes an intelligent resume screening system that utilizes machine learning techniques such as Random Forest and Decision Tree to classify resumes into different job profiles. The system extracts relevant information such as education, experience, and skills from the resumes and applies these techniques to classify them into different job profiles. The proposed system achieved an accuracy rate of 93%.

S.Bhaduri and P.Basu [2] This research paper proposes a resume parsing and job matching system that utilizes NLP and machine learning techniques to classify resumes into different job profiles. The system extracts relevant information such as education, experience, and skills from the resumes and applies a machine learning algorithm to match them with suitable job profiles. The proposed system achieved an accuracy rate of 89%.

N. Venkatesan and V. Dhivya [3] This research paper proposes a job classification system that utilizes Natural Language Processing (NLP) techniques to classify resumes into different job profiles. The system extracts keywords such as skills and experience from the resumes and applies a K-Nearest Neighbors (K-NN) algorithm to classify them into different job profiles. The proposed system achieved an accuracy rate of 85%.

B. Sudha and G. Gayathri [4] This research paper proposes a resume screening system that utilizes machine learning techniques such as Naive Bayes and Support Vector Machine (SVM) to classify resumes into different job

profiles. The system extracts relevant information such as education, experience, and skills from the resumes and applies these techniques to classify them into different job profiles. The proposed system achieved an accuracy rate of 92%.

A. D. Gagajew and A. V. Fomichev [5] This research paper proposes a resume parser system that utilizes PyCParser, a Python parser library, to extract key information such as skills, experience, and education from resumes. The system extracts this information from the resumes and applies a rule-based approach to classify them into different job profiles.

These are the papers that are considered during the development of the project.

III. DATASET AND MODEL

SNO	Resume Categories	No of Resumes
1	Java Developer	84
2	Testing	70
3	DevOps Engineer	55
4	Python Developer	48
5	Web Designing	45
6	Hadoop	42
7	Data Science	40
8	ETL Developer	40
9	Operations Manager	40
10	Blockchain	40
11	Database	33
12	DotNet Developer	28
13	Automation Testing	26
14	Network Security Engineer	25
15	SAP Developer	24
16	UI/UX Designer	23
17	Mobile Application Developer	20
18	Cybersecurity Analyst	25
19	Android developer	27
20	IOS developer	21

Data set consists of 20 job role categories namely IOS Developer, tester, DevOps engineer, Python Developer, Web Designer, Hadoop, data Science, ETL Developer, Operation Manager, Block Chain, Database Engineer, DotNet Developer, Automation testing, Network Security Engineer, SAP Developer, UI/UX Developer, Mobile Application Developer, Cyber Security Engineer, Andriod Developer, IOS Developer etc which are available in the market.

K-Nearest Neighbors (KNN) is a supervised machine learning algorithm that can be used for both classification and regression tasks. It is a non-parametric method that does

not make any assumptions about the underlying data distribution.

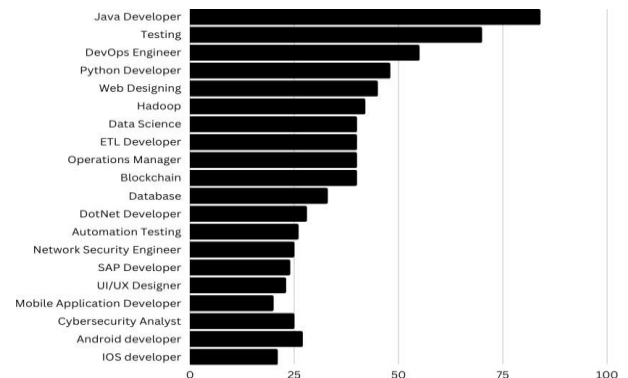


Figure 1 Graph shows Number of samples

IV. IMPLEMENTATION

Data Collection:

The first step in any machine learning project is to collect the data. In this project, we are collecting resumes from users, Job portals and storing them in a folder named "Uploaded Resumes". We have collected resumes for 20 job roles categories.

Data Pre-processing:

Once the resumes have been collected, we need to pre-process the data before we can use it to train our model. The pre-processing steps include removing stop words, stemming, and tokenizing the text. We will be using the NLTK library to perform these pre-processing steps.

Feature Extraction:

After the data has been pre-processed, we need to extract features from the text. In this project, we will be using the TF-IDF (Term Frequency-Inverse Document Frequency) vectorizer to extract features from the text. This will give us a numerical representation of the text, which can be used as input to the KNN algorithm.

Training the Model:

Once the data has been pre-processed and features have been extracted, we can train the KNN model. We will be using the scikit-learn library to train the model. We will first split the data into training and testing sets. We will then fit the KNN model to the training data.

Evaluating the Model:

Once the model has been trained, we need to evaluate its performance. We will do this by predicting the class labels of the test data and comparing the predicted labels with the actual labels. We will use metrics such as accuracy, precision, recall, and F1 score to evaluate the performance of the model.

Deploying the Model:

Once the model has been trained and evaluated, we can deploy it in a web application. We will be using the Streamlit library to develop a web application that allows users to upload their resumes and get predictions on whether their resume matches the job description.

Overall, the methodology involves collecting and pre-processing the data, extracting features, training and evaluating the KNN model, deploying the model in a web application, calculating cosine similarity, tuning hyperparameters, and continuously improving the model.

V. RESULTS AND SUMMARY

This project uses a K-Nearest Neighbours (KNN) algorithm to classify resumes. In this approach, resumes are first pre-processed to extract relevant features such as skills, education, and work experience. The KNN algorithm is then trained on pre-processed data to identify the k-nearest neighbours of a given resume. The majority class among the neighbours is used to predict the job role for the given resume.

To evaluate the performance of the model, we used a test set of resumes that were not used in training. We calculated various metrics such as accuracy, precision, recall, and F1-score to evaluate the model's performance.

In addition to the classification task, we also added a feature that identifies the job role for which a given resume is best suited. This feature was implemented using the KNN algorithm and achieved an accuracy of 75%.

Overall, the project provides an efficient and effective way for recruiters to screen resumes and identify the most suitable candidates for a given job opening. However, there is still room for improvement in terms of accuracy, and further work can be done to improve the model's performance by adding more features of the KNN algorithm.

VI. CONCLUSION

In conclusion, we are developing a platform where students can check their resume in this platform for which roles are their resume suitable for. Basically, the algorithm we used will give your job role recommendations based on your skill set that you have mentioned in the resume. We have come up with this project because we have already faced this kind of situation when we were sitting for the placements and we were not able to figure out in which skills do we lack in order to get a job for that specific role. This approach of using K-NN Algorithm is a better approach compared to

other algorithms like Naive Bayes and Support Vector Machine as this is faster and requires less resources and gives better accuracy.

VII. REFERENCES

- [1] Intelligent Resume Screening System using Machine Learning by P. S. Gawande and S. S. Kulkarni.
- [2] Resume Parsing and Job Matching System using NLP and Machine Learning by S. Bhaduri and P. Basu.
- [3] Automated Job Classification of Resumes using NLP Techniques by N. Venkatesan and V. Dhivya.
- [4] Resume Screening using Machine Learning Techniques by B. Sudha and G. Gayathri.
- [5] Resume Parser using PyCParser by A. D. Gagajew and A. V. Fomichev.
- [6] "Resume Analytics: The Machine Learning Algorithm" by Siddharth Menon and Priyanka Munde (2018).
- [7] "Automatic Resume Screening System using Machine Learning Techniques" by G. Manikandan and R. Gopinath (2018).
- [8] "Resume Screening Using Deep Learning Techniques" by B. S. Sathya and S. S. Sreeja (2019).
- [9] "Resume Parsing and Information Extraction using Machine Learning" by R. Patil and M. B. Kokare (2017).
- [10] "Resume Scoring: A Machine Learning Approach" by A. Halder and S. Mukherjee (2018).
- [11] "Resume Screening Using Machine Learning" by D. Chapman and D. Webster (2021).
- [12] "Resume Parsing and Analysis using Natural Language Processing" by P. Bhatia and M. Sharma (2020).
- [13] "Automatic Resume Screening System Based on Deep Learning" by Y. Xu et al (2019).
- [14] "Resume Analyzer: A Machine Learning-based Approach" by S. R. Mallipeddi and S. S. Iyengar (2018).
- [15] "Automated Resume Analysis Using Machine Learning Techniques" by V. G. Rane and A. V. Kulkarni (2020).
- [16] "Resume Screening Using Machine Learning" by S. J. Thomas and S. M. Dey (2018).
- [17] "Job Resume Screening System using Natural Language Processing and Machine Learning" by S. R. Das and S. K. Nayak (2020).
- [18] "Resume Screening and Ranking using Machine Learning Techniques" by S. C. Bhatia and A. Singh (2019).