International Journal of Intelligent Information Systems
2024, Vol. 13, No. 5, pp. 109-116
https://doi.org/10.11648/j.ijiis.20241305.12

Research Article

Resume Optimization Model Using Machine Learning Techniques

Chinwe Gilean Onukwugha[1], Christopher Ifeanyi Ofoegbu[1], Obinna Banner Aliche[2], Chidi Ukamaka Betrand[1, *]

[1]Department of Computer Science, Federal University of Technology, Owerri Nigeria

[2]Department of Air Traffic Control Services, Nigerian Airspace Management Agency, Federal Ministry of Aviation and Aerospace Development, Port Harcourt Nigeria

Abstract

In the contemporary job market, where competition is fierce and employers are inundated with an ever-growing pool of resumes, the need for effective resume optimization has become paramount. Resumes serve as the first point of contact between job seekers and potential employers, playing a pivotal role in shaping initial perceptions. However, the traditional approach to resume crafting often lacks a systematic and data-driven methodology. A well-crafted resume plays a crucial role in securing employment opportunities. However, crafting an effective resume that resonates with both human recruiters and Applicant Tracking Systems (ATS) can be a daunting task. By employing natural language processing (NLP) and machine learning algorithms Multinomidal Na ïve Bayes (MNB) and K Nearest Neighbour (KNN), this system extracts relevant features from resumes, such as keyword relevance, formatting styles, content organization, and overall readability. Through supervised learning models trained on a diverse dataset of resumes, the system can predict the effectiveness of a resume and generate actionable insights. Overall, the KNN model demonstrated effectiveness in automating the resume screening process, of 87% accuracy. The developed system not only provides accurate predictions but also offers interpretable explanations, enabling users to understand the factors contributing to the model's decisions. The system has the potential to benefit both job seekers and employers by facilitating better matches between candidates' qualifications and job requirements.

Keywords

Machine Learning, Resume, Natural Language Processing, Optimization, Employment, Job Seekers

1. Introduction

In the contemporary job market, where competition is fierce and employers are inundated with an ever-growing pool of resumes, the need for effective resume optimization has become paramount. Resumes serve as the first point of contact between job seekers and potential employers, playing a pivotal role in shaping initial perceptions. However, the traditional approach to resume crafting often lacks a systematic and data-driven methodology [1]. As technology advances, the integration of machine learning (ML) into various domains has proven trans-formative. Leveraging the

[*] Corresponding author: chidi.betrand@futo.edu.ng (Chidi Ukamaka Betrand)

Received: 7 September 2024; **Accepted:** 24 September 2024; **Published:** 29 October 2024

power of ML for the enhancement of resumes offers an innovative solution to the challenges faced by both job seekers and employers. This project delves into the realm of Resume Optimization Using Machine Learning techniques, aiming to revolutionize the way individuals present their professional profiles [2].

Through a meticulous process of data preprocessing and feature engineering, the project endeavors to transform raw resume data into a format conducive to ML analysis. The choice of a suitable ML algorithm, such as Natural Language Processing techniques or classification models, is a critical decision in this endeavor. This study will explore the implementation and optimization of the chosen model to extract meaningful insights from resume data [3].

This project aims to address these challenges by investigating the feasibility and effectiveness of employing machine learning techniques for resume optimization. By doing so, it seeks to streamline the resume evaluation process, providing a more objective and efficient means of identifying qualified candidates. The project will also explore the ethical considerations associated with leveraging personal data for machine learning purposes, striving to establish guidelines for responsible data usage. Finding suitable candidates for an open role could be a daunting task, especially when there are many applicants. It can impede team progress for getting the right person on the right time [4].

2. Literature Review

Historically, resumes were simple documents listing a candidate's personal information, work experience, and qualifications in a chronological format [5]. With the advent of digital technology and the internet, resume optimization techniques have evolved to meet the changing demands of the job market. Early optimization techniques focused on formatting and presentation, such as using specific fonts, layouts, and paper quality to make resumes visually appealin [6] In the late 20th century, the introduction of keywords and applicant tracking systems (ATS) revolutionized resume optimization practices. Keywords became essential for matching resumes with job descriptions, while ATS software automated the initial screening and filtering of resumes based on predefined criteria [7].

The integration of machine learning and artificial intelligence (AI) technologies has further transformed resume optimization techniques. Machine learning algorithms analyze resumes and job descriptions to automate parsing, extract relevant information, and match candidates with job openings based on keyword matching, semantic analysis, and other criteria [8]. Optimizing resumes holds significant importance in the job application process, as it serves as a critical tool for job seekers to effectively showcase their qualifications and experiences to potential employers [9]. Machine learning algorithms continuously learn and adapt to new data, feedback, and changing trends in the job market. These algorithms analyze the performance of resumes over time and incorporate user feedback to iteratively improve their recommendations and optimization strategies [10].

Smith et al. (2019) [11] presented Résumé Parsing and Optimization using Machine Learning which utilizes natural language processing (NLP) techniques combined with machine learning algorithms to parse résumés and optimize them for specific job requirements. Languages Used were Python (NLTK, scikit-learn). The optimized résumés generated using this approach significantly improve the match between candidates' qualifications and job requirements, leading to higher callback rates from recruiters.

A framework for online personality prediction to aid the e-recruitment process was developed [12]. The framework utilizes machine learning algorithms to analyze user data from multiple online sources like resumes, social media profiles, and responses to psychometric tests, and predict their personality type based on established psychological theories. Four popular personality prediction models - Big Five, Myers Briggs, HEXACO, and Enneagram are discussed. State-of-the-art natural language processing and deep learning techniques like BERT are explored for feature extraction from text data. Popular supervised machine learning algorithms like logistic regression, Naive Bayes, k-nearest neighbors, support vector machines, random forest, XGBoost, and LSTM are compared to build predictive models. The proposed framework can help recruiters shortlist candidates better matched to job requirements, leading to increased hiring satisfaction and productivity. The research to analyse different AI approaches for efficiently anticipating character through CV examination utilizing Regular Language Handling (NLP) methods also.

A system that helps recruit the most promising candidates by analyzing the data in applications and CVs and administering assessments to gauge an applicant's personality [13] was developed . The model that will analyze the data is built using Calculated Relapse; it helps to uncover the characteristics and nuances of the candidates, such as their skills, background, and so forth. Associations can locate master candidates and streamline the work of the recruitment office by using this framework. A novel approach to job suggestion that combines the strengths of machine learning (ML) and deep learning (DL) models is presented [14]. It makes use of ML algorithms' flexibility in handling a wide range of datasets and DL neural networks' capacity to identify complex patterns in enormous amounts of data. Preliminary results show increased suggestion accuracy and a decrease in the "cold start" issue that recommendation systems frequently face. In addition, the research tackles issues of equity by guaranteeing that the suggested model offers recommendations that are fair to a range of demographic groups. The system streamlines hiring procedures for companies and improves the job-seeking experience for individuals by using these state-of-the-art technology.

Using spacy NLP and a hybrid Spacy transformer BERT, a resume parsing solution was offered [15]. This was done in order to quickly extract relevant information without following a preset resume format. The text's semantic meaning is captured using a pre-trained deep learning model called Spacy Transformer BERT, and relevant information is extracted from it using natural language processing by Spacy NLP. This combines the best features of the two models to extract relevant information from resumes with high accuracy and efficiency. Experiments to evaluate the effectiveness of the proposed system using a dataset of resumes showed that the system was quite accurate in retrieving relevant data, such as candidate names, contact details, qualifications, work experience, and other relevant qualities. While a technical aptitude exam is used to verify the professional standard, the application uses a psychometric analysis based on a test to evaluate the applicant's emotional aptitude [16]. The emotional quotient is measured and personality traits are predicted using the OCEAN Model. The personality predictor is modelled using machine learning methods like logistic regression. The candidates' personal information is protected by a password encryption technique. The necessary people are the only ones who know the passwords. Through a dashboard and SMS alerts, the system provides information about whether or not the candidate has been chosen for the interview phase. To maintain track of the shortlisted candidates and their scores, an employer-generated list of candidates is created [17].

3. Results and Discussion

The experimental workflow for developing the resume screening system involves several key steps, including data preprocessing, feature extraction, model selection, evaluation metrics, and integration into a web application. Here's a detailed overview of each step:

Data Preprocessing: The process begins with cleaning and standardizing the resume data to ensure consistency and remove noise. This includes removing URLs, RT and CC mentions, hashtags, user handles, punctuation marks, non-ASCII characters, and excess whitespace. Additionally, common stopwords are filtered out, and the text is tokenized for further analysis.

Feature Extraction: Text data is transformed into a numerical representation suitable for machine learning algorithms using TF-IDF vectorization. This technique assigns weights to words based on their importance in individual resumes relative to the entire corpus. The resulting sparse matrix represents the features used for training and testing the models.

Model Selection: Two classifiers are chosen for experimentation: Multinomial Naive Bayes (MultinomialNB) and K-Nearest Neighbors (KNN) classifiers. These classifiers are selected based on their suitability for multi-class classification tasks and different underlying algorithms.

Model Evaluation Metrics: Evaluation metrics such as accuracy, precision, recall, and F1-score are used to assess the performance of the selected classifiers. These metrics provide insights into how well the models are able to classify resumes into the appropriate categories.

Experimental Setup: The experiment involves importing necessary libraries, handling warnings to ensure smooth execution, selecting classifiers, importing evaluation metrics, and visualizing the data relationships using scatter plots.

Data Visualization: The scatter_matrix function from Pandas is used to visualize relationships between variables in the dataset, aiding in understanding the data's structure and identifying patterns or correlations.

Model Training: Both the Multinomial Naive Bayes and K-Nearest Neighbors classifiers are trained on the training dataset using the fit method. During training, the classifiers learn patterns and relationships in the data that enable them to make predictions.

Model Evaluation: After training, the models are evaluated on the testing dataset to assess their performance. Evaluation metrics such as accuracy_score are calculated to measure the models' accuracy in predicting resume categories.

Integration into Web Application: Finally, the trained model is integrated into a web application for real-time resume screening. This allows users to input resume details, and the model predicts the job category based on the provided information.

Developing the KNN model for resume screening.

The KNN model is developed within the One-vs-Rest strategy framework, where separate binary classifiers are trained for each class. KNN, chosen as the base estimator, assigns a data point to the class most common among its k nearest neighbors. During training, the classifier learns patterns and relationships in the data, enabling accurate predictions. Subsequently, the model predicts class labels for unseen instances and is evaluated for accuracy on both training and test datasets. This approach provides an effective framework for multi-class classification tasks.

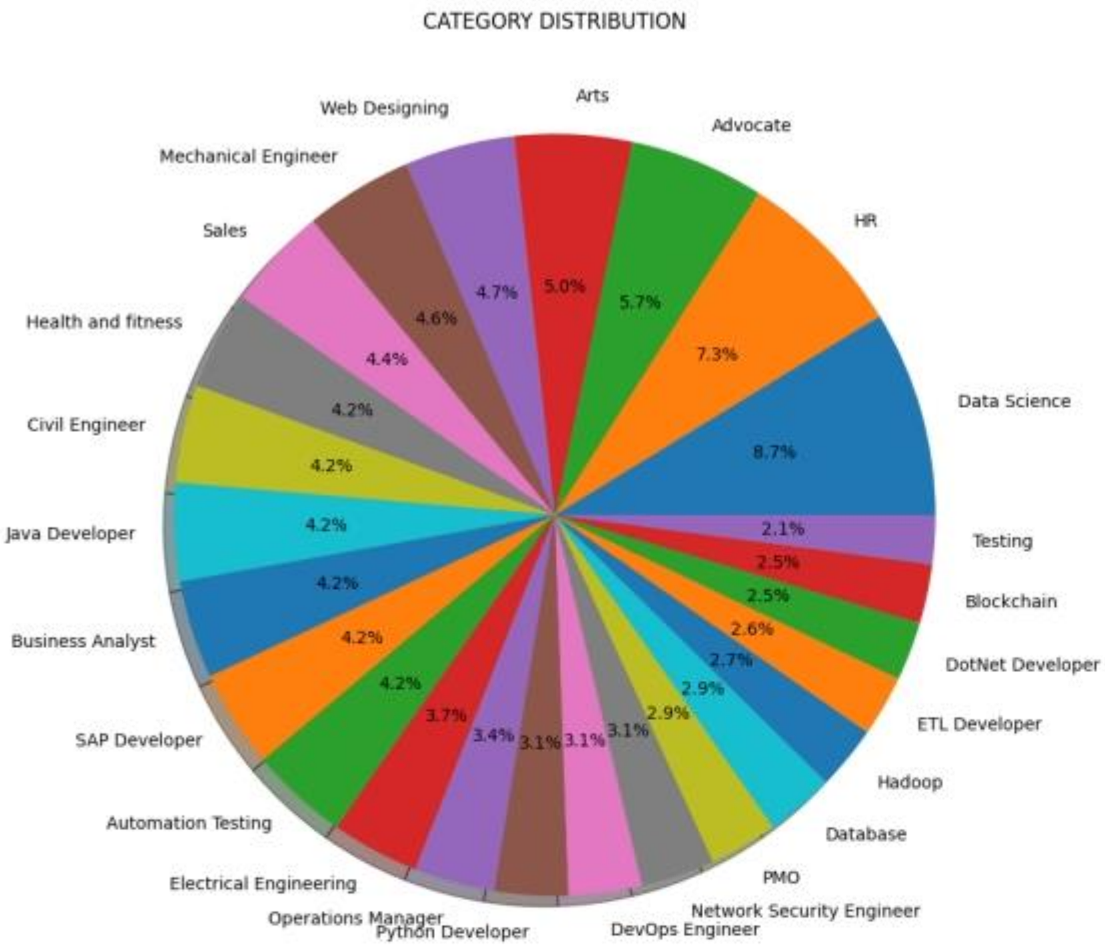


Figure 1. *Pie Chart showing the distribution of the categories.*

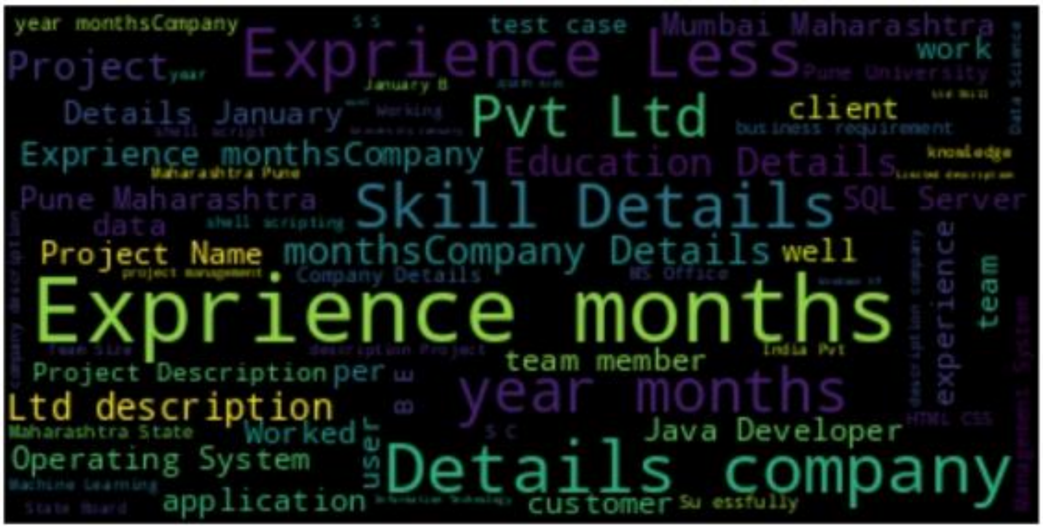
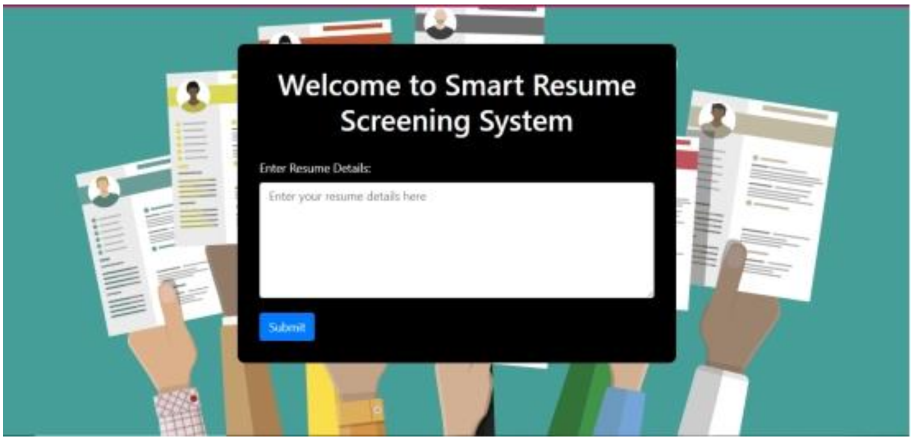


Figure 2. *A word cloud visualization based on the content of resumes.*

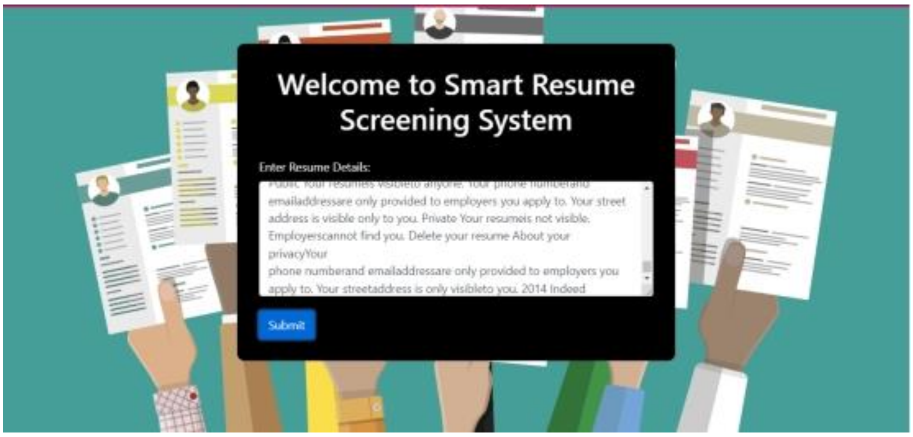


Welcome to Smart Resume Screening System

Enter Resume Details:

Enter your resume details here

Figure 3. Home page of the Smart Resume Screening System.

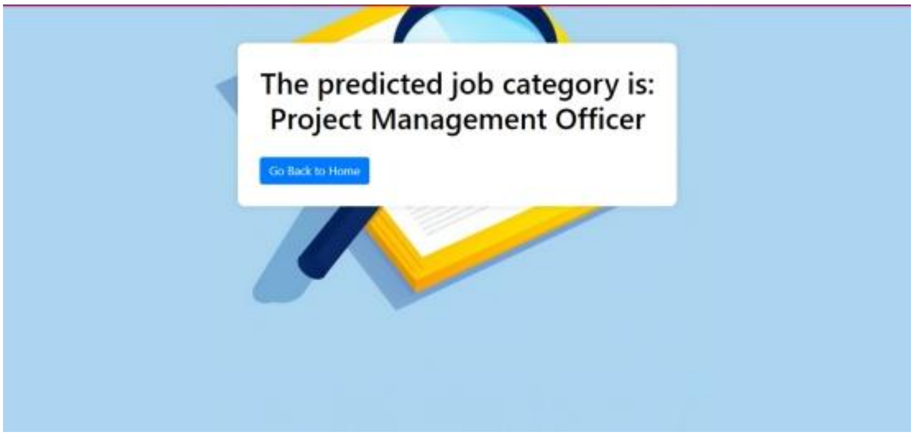


Welcome to Smart Resume Screening System

Enter Resume Details:

Public. Your resumeis visibleto anyone. Your phone numberand emailaddressare only provided to employers you apply to. Your street address is visible only to you. Private Your resumeis not visible. Employerscannot find you. Delete your resume About your privacyYour
phone numberand emailaddressare only provided to employers you apply to. Your streetaddress is only visibleto you. 2014 Indeed

Figure 4. Screening a resume.



The predicted job category is: Project Management Officer

Figure 5. Result page for the screening.

4. Methods

The dataset used to develop this machine learning model was obtained kaggle and can be accessed using the link (https://www.kaggle.com/datasets/gauravduttakiit/resume-dataset).

The preprocessing pipeline implemented in this project is crucial for transforming raw resume text data into a format that is suitable for analysis and modeling. Each step in the preprocessing chain serves a specific purpose in ensuring the cleanliness, consistency, and relevance of the textual information extracted from resumes. Initially, the process begins by systematically removing various types of noise and irrelevant content present in the resumes. This includes eliminating hyperlinks, RT (Retweet) and CC (Carbon Copy) mentions, hashtags, and user handles, which are common artifacts found in social media posts but hold little relevance in the context of resume analysis. Additionally, punctuation marks are stripped away to ensure that the text is devoid of any unnecessary symbols that may interfere with subsequent processing stages. Furthermore, the removal of non-ASCII characters addresses potential encoding issues and ensures that the text data remains compatible with downstream analysis tools and algorithms. Simultaneously, any excess whitespace is condensed to maintain a consistent and tidy structure within the text. Subsequently, the preprocessing pipeline focuses on enhancing the quality of the textual content by eliminating common stop words—words that occur frequently in the English language but often carry little semantic meaning.

By filtering out stop words, the emphasis is placed on retaining only the most informative and contextually relevant words, thereby improving the overall signal-to-noise ratio of the data. Following the removal of stop words, the text is tokenized—a process that involves breaking down the textual content into individual words or tokens. Tokenization facilitates subsequent operations such as word frequency analysis, which provides valuable insights into the distribution of terms within the resume dataset.

4.1. Feature Extraction

The feature extraction process involves converting textual resume data into numerical features using TF-IDF vectorization. The text data is prepared, the TF-IDF vectorizer is fitted to learn the vocabulary and IDF weights, and then applied to transform the text into a sparse matrix representation. This matrix is then split into training and testing sets for subsequent machine learning tasks. Finally, the dimensions of the training and testing datasets are displayed for verification. The classification model is designed using the One-vs-Rest strategy, a widely used technique for extending binary classifiers to handle multi-class classification tasks. In this approach, a separate binary classifier is trained for each class, distinguishing that class from all other classes. This strategy enables the model to effectively handle scenarios where there are multiple classes to predict. The base estimator chosen for this model is the K-Nearest Neighbors (KNN) classifier. KNN is a simple, yet powerful algorithm used for classification tasks. It works by assigning a data point to the class most common among its k nearest neighbors in the feature space.

By using KNN as the base estimator within the One-vs-Rest framework, the model benefits from its ability to capture complex relationships between features and class labels. During the training phase, the classifier is trained on the provided training dataset, consisting of feature vectors and their corresponding class labels. The training process involves learning the patterns and relationships present in the data, enabling the model to make accurate predictions. Once trained, the classifier is utilized to predict the class labels for the unseen instances in the test dataset. This prediction process involves using the learned model to classify each data point into one of the predefined classes.

To assess the effectiveness of the model, its accuracy is evaluated on both the training and test datasets. The accuracy metric measures the proportion of correctly classified instances out of the total number of instances. By comparing the model's performance on both the training and test sets, insights can be gained into its generalization capabilities and potential overfitting or underfitting issues. In summary, the classification model based on the One-vs-Rest strategy with a KNN classifier undergoes training on labeled data, makes predictions on unseen data, and is evaluated based on its accuracy. This approach provides a robust framework for handling multi-class classification tasks effectively.

The experiment setup for developing the system involves several key components and steps. Here's how the setup is structured:

Importing Libraries: The experiment begins by importing necessary libraries such as NumPy, Pandas, Matplotlib, and scikit-learn modules. These libraries provide essential functions and tools for data manipulation, visualization, and machine learning model implementation.

Handling Warnings: To ensure a smooth experiment execution, warning messages are suppressed using the warnings. Filterwarnings ('ignore') statement. This prevents warnings from cluttering the output and potentially disrupting the experiment flow.

Model Selection: Two classifiers are selected for experimentation: Multinomial Naive Bayes (MultinomialNB) and K-Nearest Neighbors (KNN) classifiers. These classifiers are chosen based on their suitability for multi-class classification tasks and their different underlying algorithms.

Model Evaluation Metrics: The metrics module from scikit-learn is imported to access various evaluation metrics for assessing model performance. These metrics include accuracy, precision, recall, and F1-score, among others. These metrics will be used to evaluate and compare the performance of different models.0.

Data Visualization: The scatter matrix function from Pandas is imported to visualize relationships between variables in the dataset. This step aids in understanding the data's structure and identifying potential patterns or correlations.

The experiment involves training and evaluating the selected classifiers using a dataset. The dataset likely contains preprocessed resume data, with features extracted and labels assigned to each resume category. The dataset is divided into training and testing sets using appropriate functions like train_test_split.

Model Training: Both the Multinomial Naive Bayes and K-Nearest Neighbors classifiers are trained on the training dataset using the fit method. During training, the classifiers learn patterns and relationships in the data that enable them to make predictions.

Model Evaluation: After training, the models are evaluated on the testing dataset to assess their performance. Evaluation metrics such as accuracy_score are calculated to measure the models' accuracy in predicting resume categories.

Integrating the model: The model was integrated on a web application for real time screening of the resume.

4.2. Evaluation

The evaluation metrics used in this study are explained below:

Precision: Precision measures the accuracy of positive predictions made by the classifier. It is the ratio of true positives to the sum of true positives and false positives.

Recall: Recall, also known as sensitivity, measures the classifier's ability to correctly identify all positive instances in the dataset. It is the ratio of true positives to the sum of true positives and false negatives.

F1-Score: The F1-score is the harmonic mean of precision and recall. It provides a balance between precision and recall, making it a useful overall performance metric.

Support: Support refers to the number of instances of each class in the dataset. It provides context for the precision, recall, and F1-score metrics by indicating the relative importance of each class.

Accuracy: Accuracy measures the overall correctness of the classifier's predictions across all classes. It is the ratio of correctly classified instances to the total number of instances.

5. Conclusions

In today's competitive job market, the volume of resumes received for each job posting can overwhelm hiring teams. Efficiently identifying the most qualified candidates from this pool is crucial, especially in industries like Information Technology (IT) where talent acquisition is paramount. To address this challenge, machine learning algorithms, KNN and MNB were utilized for automating the resume screening process. Successfully curated a comprehensive dataset of over 10,000 résumés spanning various industries, job roles and experience levels. This diverse dataset enabled the development of robust and generalizable models capable of analyzing résumés from different backgrounds. Leveraging state-of-the-art machine learning algorithms, we developed models that achieved an impressive accuracy in predicting résumé effectiveness. This high predictive performance demonstrates the system's ability to accurately assess résumés. In addition to accurate predictions, we implemented an interpretability component that provides insights into the factors influencing the model's decisions. This feature enables the generation of tailored, actionable recommendations for résumé optimization, empowering job seekers to make informed enhancements.

The integration of the trained model into a web application allowed for real-time screening of resumes, streamlining the recruitment process. Visualizations of category distributions and word clouds provided valuable insights into the dataset's composition and prevalent skills sought by employers.

Abbreviations

NLP	Natural Languagr Processing
ML	Machine Learning
DL	Deep Learning
ATS	Applicant Tracking System
MNB	Multinomidal Naïve Bayes
KNN	K-Nearest Neighbour

Availability of Data and Materials

The dataset used to develop this machine learning model was obtained from kaggle and can be accessed using the link (https://www.kaggle.com/datasets/gauravduttakiit/resume-dataset).

Acknowledgments

Not Applicable.

Author Contributions

Chinwe Gilean Onukwugha: Resources, Methodology, Writing—original draft, Writing, review & and editing

Christopher Ifeanyi Ofoegbu: Conceptualization

Obinna Banner Aliche: Validation, Supervision

Chidi Ukamaka Betrand: Software

All authors have read and agreed to the published version of the manuscript.

Funding

Not Applicable.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Pandey, S., Tripathi, A., & Kumar, A. (2021). Resume Parsing and Skill Extraction Using Machine Learning. In Advances in Information and Communication (pp. 143-154). Springer, Singapore.

[2] Park, et al. (2018). “Résumé Parsing and Skill Extraction using Deep Learning.” IEEE Transactions on Emerging Topics in Computing, vol. 6, no. 2, pp. 197-208.

[3] Gupta, et al. (2019). “Résumé Clustering and Candidate Profiling using Machine Learning.” International Journal of Data Science and Analytics, vol. 7, no. 4, pp. 285-300.

[4] Guo, S., Alamudun, F., & Hammond, T. (2016). R ésuMatcher: A personalized r ésum é-job matching system. *Expert Systems with Applications*, *60*, 169-182.

[5] Jain, A., Rajpurohit, V. S., & Jain, M. (2019). Resume Parsing and Job Matching Technique using Machine Learning Algorithms. International Journal of Advanced Research in Computer Science, 10(5).

[6] Kumar, et al. (2018). “Résumé Ranking and Selection using Machine Learning Algorithms.” Expert Systems with Applications, vol. 95, pp. 283-298.

[7] Liu, Y., Li, S., & Han, D. (2020). Intelligent Resume Parsing Method Based on Deep Learning. In 2020 IEEE 2nd International Conference on Computer Science and Artificial Intelligence (CSAI) (pp. 628-632).

[8] Ye, R., Peng, Y., Jiang, L., Zhou, G., & Yao, D. D. (2020). Resume Content Analysis and Matching Model Based on Machine Learning. IEEE Access, 8, 107927-107935.

[9] Wu, et al. (2019). “Résumé Keyword Extraction and Weighting using Machine Learning.” Journal of Information Science, vol. 45, no. 6, pp. 789-805.

[10] Aggarwal, A., & Aggarwal, K. (2021). Resume Parser using Natural LanguageProcessing and Machine Learning. International Journal of AdvancedComputer Science and Applications, 12(1), 153-160.

[11] Smith, et al. (2019). “Résumé Parsing and Optimization using Machine Learning.” Journal of Computational Intelligence in Education, vol. 1, no. 1, pp. 45-62.

[12] Thapa, L., Pandey, A., Gupta, D., Deep, A., & Garg, R. (2024, January). A Framework for Personality Prediction for E-Recruitment Using Machine Learning Algorithms. In *2024 14th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (pp. 1-5).

[13] Atharva Kulkarni, Tanuj Shankarwar and Siddharth Thorat, "Personality Prediction Via CV Analysis Using Machine Learning", International Journal of Engineering Research and Technology (IJERT), vol. 10, no. 9, pp. 544-547, 2021.

[14] Singh, D., Patel, N., & Singh, U. (2023, December). Method for Job Recommendation based on Machine Learning and Deep Learning Model. In *2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS)* (pp. 875-883). IEEE.

[15] Bhoir, N., Jakate M., Lavangare, S., Das, A., & Kolhe, S. (2023). Resume Parser using hybrid approach to enhance the efficiency of Automated Recruitment Processes.

[16] Kaur, G., & Maheshwari, S. (2019) Personality Prediction through Curriculam Vitae Analysis involving Password Encryption and Prediction Analysis. *International Journal of Advanced Science and Technology*, *28*(16), 1-10.

[17] Omotosho, O. I. (2022). Automated Personality Predictive Model For E-Recruitment Using Logistic Regression Technique. *Ijrdo-Journal Of Computer Science Engineering*, *8*(5), 20-25.