RESUMECRAFT – BEST RESUME BUILDER AND SCREENER USING NLP AND ML

[1]Divina Godvia Violet R
[1]Francis Xavier Engineering College, Tirunelveli, TamilNadu, India
[1]divinar.pg25.cs@francisxavier.ac.in

***Abstract*—This paper presents ResumeCraft, a web-based application designed to assist users in creating, editing, and screening resumes using Natural Language Processing (NLP) and Machine Learning (ML) techniques. The system provides an integrated platform that simplifies the resume development process while enhancing its effectiveness for job applications. It begins with a user-friendly landing interface that guides individuals through the various functionalities of the system. The application includes a built-in resume editor with customizable templates, enabling users to design and modify resumes in a structured and visually appealing format, similar to modern design tools. Users can easily generate, update, and download resumes in multiple formats suitable for professional use. In addition to resume creation, the system incorporates a keyword recommendation module that suggests role-specific skills and terms based on industry requirements. This feature helps users improve resume quality and ensures better compatibility with Applicant Tracking Systems (ATS), which are widely used for automated resume filtering. The proposed system also includes a resume screening component that analyzes uploaded resumes and predicts suitable job roles using machine learning models trained on textual data. The system applies preprocessing techniques such as text cleaning, tokenization, and feature extraction using TF-IDF to improve classification accuracy and relevance. Furthermore, the application provides real-time feedback to users, allowing them to refine their resumes based on predicted outcomes and keyword suggestions. By combining resume building, keyword optimization, and automated screening into a single platform, ResumeCraft offers a comprehensive solution for job seekers. The system improves resume quality, increases job matching accuracy, and supports efficient career development processes through intelligent automation techniques. The system is designed to be scalable and adaptable across multiple domains, allowing users from different professional backgrounds to benefit from its features. It also enhances usability by providing a seamless and intuitive experience for both beginners and experienced users.**

***Index Terms*—Resume Screening, Natural Language Processing (NLP), Machine Learning, Keyword Extraction, Applicant Tracking Systems (ATS)**

I. Introduction

In today's dynamic job market, the power of a well-crafted resume cannot be overstated. Whether you're a seasoned professional seeking career advancement or a recent graduate embarking on your professional journey, the ability to create a compelling resume is paramount. Enter ResumeCraft – your trusted ally in the quest for professional success. With a suite of intuitive tools, personalized guidance, and industry-leading features, ResumeCraft empowers individuals to craft resumes that resonate with employers, showcase their unique talents, and unlock new opportunities.

At the heart of ResumeCraft lies a commitment to simplicity and effectiveness. We understand that the resume-building process can be daunting, filled with uncertainty and complexity. That's why we've designed ResumeCraft to be your go-to platform for creating professional resumes effortlessly. With our user-friendly interface and customizable templates, you'll navigate the resume-building process with confidence, regardless of your level of experience or expertise.

One of ResumeCraft's standout features is its comprehensive resume template, meticulously crafted to suit a diverse range of industries, professions, and career stages. Whether you're in finance, healthcare, technology, or creative fields, ResumeCraft offers templates that capture the essence of your professional identity and help you stand out in a crowded job market. From classic designs to modern layouts, our templates strike the perfect balance between aesthetics and functionality, ensuring that your resume leaves a lasting impression on employers.

In today's digital age, keyword optimization is crucial for ensuring that your resume gets noticed by applicant tracking systems (ATS) and human recruiters alike. With ResumeCraft's cutting-edge keyword optimization feature, you'll receive expert guidance on incorporating industry-specific keywords and phrases that elevate your resume's visibility and relevance. This approach improves resume visibility in Applicant Tracking Systems (ATS) and enhances the efficiency of the job application process.

But ResumeCraft is more than just a resume builder – it's a comprehensive career toolkit designed to empower you at every stage of your professional journey. With our real-time feedback and suggestions feature, you'll receive instant insights and actionable recommendations for improving your resume's clarity, coherence, and impact. Whether it's refining your language, restructuring your content, or enhancing your formatting, ResumeCraft's feedback engine ensures that your r esume shines brightest in the eyes of recruiters.

In this work, machine learning algorithms are used to classify resumes based on extracted textual features. When it comes to keyword prediction, these algorithms play a vital role in enhancing accuracy and efficiency. However, it's essential to consider both the training duration and accuracy aspects when evaluating the performance of these models.

In addition to crafting standout resumes, ResumeCraft offers robust export and sharing options, allowing you to download your resume in multiple formats – PDF, Word, or plain text – for seamless sharing, printing, or online submission. Our sharing features enable you to share your resume with recruiters, mentors, and peers effortlessly, fostering collaboration and networking opportunities that are essential for career advancement.

But ResumeCraft's mission extends beyond just creating resumes – it's about empowering you to take control of your career destiny. Through our comprehensive analytics and tracking capabilities, you'll gain invaluable insights into your resume's performance, including views, downloads, and shares. Armed with this data, you'll make informed decisions about your job search strategy, identify areas for improvement, and capitalize on emerging opportunities in your field.

II. DATASET

Developing a dataset for ResumeCraft involves compiling a diverse collection of resumes that encapsulate various industries, career levels, and professional backgrounds. The dataset serves as the foundation for training machine learning algorithms, enhancing keyword recognition, and improving resume parsing accuracy. To ensure comprehensive coverage, the dataset should encompass resumes from different geographical regions, educational backgrounds, and job functions, reflecting the diversity of the global workforce.

The dataset includes resumes in different formats, such as PDF, Word, and plain text, to accommodate varying file types commonly encountered in job applications. Each resume is meticulously annotated with metadata, including job titles, skills, experiences, education, and certifications, enabling precise extraction of relevant information during parsing. Additionally, the dataset incorporates labeled keywords and phrases associated with specific industries, roles, and job descriptions, facilitating keyword optimization and enhancing ATS compatibility.

Continuous refinement and expansion of the dataset are essential to adapt to evolving industry trends, job market demands, and emerging technologies. Regular updates ensure that ResumeCraft remains equipped with the latest insights and patterns, enabling users to create resumes that resonate with employers and stand out in competitive job markets. By leveraging a rich and diverse dataset, ResumeCraft empowers users to craft personalized, impactful resumes that capture their unique professional identities and propel them toward career success.

III. METHODOLOGY AND IMPLEMENTATION

Our process for developing a productive web application that receives user input and forecasts the result. Every stage is essential to the overall efficacy and efficiency of the suggested course of action.

3.1 Dataset Collection:

A collection of data in the format of .csv is imported into the python interpreter(Jupyter notebook) using the libraries in anaconda and using the function read_csv and stored in a variable name. To view the dataset, we can use the command head(). In our data analysis workflow, we begin by importing a dataset stored in the .csv format into the Python interpreter, specifically utilizing Jupyter notebook, a popular platform for collaborative

There are some publicly available datasets that may contain resumes along with their associated categories.

Websites like Kaggle or academic repositories might have datasets that could be used for this purpose.

Before using the dataset, it's essential to preprocess the resume text, which might involve tasks such as removing special characters, standardizing formats, handling missing data, and tokenizing the text into manageable units.

Once you have collected and prepared your dataset, you can split it into training, validation, and test sets for model training, evaluation, and testing purposes. It's crucial to ensure that your dataset is representative of the real-world scenarios you intend to apply your model to and that it covers a wide range of job categories and variations in resume formats.

3.1.1 Importing Essential Libraries:

In the development of applications like the one described, it's essential to import libraries that facilitate various functionalities. Streamlit, pickle, regular expressions (re), and the Natural Language Toolkit (NLTK) are among the crucial libraries utilized.

Streamlit is a popular Python library that simplifies the creation of interactive web applications. It offers intuitive APIs for building user interfaces, handling user inputs, and visualizing data. By importing Streamlit, developers can streamline the process of creating web-based applications for tasks like resume screening.

Pickle is a module in Python used for serializing and deserializing Python objects. It enables the saving and loading of machine learning models, allowing developers to persist trained models for later use. In the context of the provided code, pickle is used to load pre-trained models stored in pickle files, facilitating the classification of resumes.

Regular expressions are powerful tools for pattern matching and text manipulation in Python. They enable developers to search for and manipulate text based on specific patterns or rules. In the code snippet provided, the re module is utilized for cleaning resume text by removing URLs, special characters, and other unwanted elements.

NLTK is a comprehensive library for natural language processing (NLP) tasks in Python. It provides modules and tools for tasks such as tokenization, stemming, part-of-speech tagging, and syntactic analysis. In the context of the resume screening application, NLTK is used for tasks like tokenization and possibly stopword removal to preprocess the text data before classification.

By importing these essential libraries, developers gain access to a wide range of functionalities and tools that are instrumental in building robust and efficient applications for tasks like resume screening and classification.

3.1.2 Data Preprocessing:

The first aspect of data preprocessing involves cleaning the raw resume text to remove noise and irrelevant information. This process typically includes tasks such as removing URLs, special characters, mentions, and non-ASCII characters. Regular expressions (regex) are commonly used for pattern matching and substitution to clean the text effectively. By eliminating unnecessary elements, the cleaned resume text becomes more manageable and conducive to accurate classification.

After cleaning the text, the next step is tokenization, where the cleaned text is split into individual tokens or words. Tokenization allows the model to understand the structure and semantics of the text data better. Additionally, normalization techniques may be applied to standardize the text, such as converting all characters to lowercase to ensure consistency in word representations. NLTK (Natural Language Toolkit) is a powerful library commonly used for tokenization and normalization tasks in natural language processing applications like this one.

Once the text data is cleaned and tokenized, feature extraction techniques are applied to represent the text in a numerical format that machine learning models can understand. In the provided code, TF-IDF (Term Frequency-Inverse Document Frequency) vectorization is utilized for feature extraction. TF-IDF assigns weights to words based on their frequency in the document and across the entire corpus, helping to capture the importance of words in distinguishing between different job categories. By transforming the resume text into TF-IDF vectors, the data is prepared for classification by the machine learning model.

3.2 Overall Process:

The primary page of the application allows users to upload their resumes in either text or PDF format. Once a resume is uploaded, the application preprocesses the text, removing irrelevant elements like URLs, special characters, and non-ASCII characters. Then, it applies TF-IDF vectorization to convert the cleaned text into numerical features. These features are then fed into a pre-trained machine learning classifier to predict the job category or role associated with the resume. The predicted category is displayed to the user, providing insight into how their skills and experiences align with different job roles.

The application includes another page that contains all the keywords associated with various job roles. This page serves as a reference guide for users to understand the specific skills, qualifications, and attributes typically sought after for different positions. By exploring this page, users can gain insights into the requirements of different job roles and tailor their resumes accordingly to improve their chances of success in their job search.

Throughout the application, user interaction is facilitated through an intuitive interface provided by the Streamlit library. Users can easily upload their resumes, view the predicted job categories, and explore the keyword reference page. Additionally, the application provides feedback to users, helping them understand how their skills and experiences match with different job roles and guiding them in refining their resumes to better align with industry expectations.

By combining machine learning with user-friendly web application development tools, the resume screening application streamlines the job search process for users, helping them identify relevant job opportunities and tailor their resumes effectively to stand out to potential employers. The inclusion of the keyword reference page further enhances the user experience by providing valuable insights into the requirements of different job roles and empowering users to make informed decisions about their career paths.

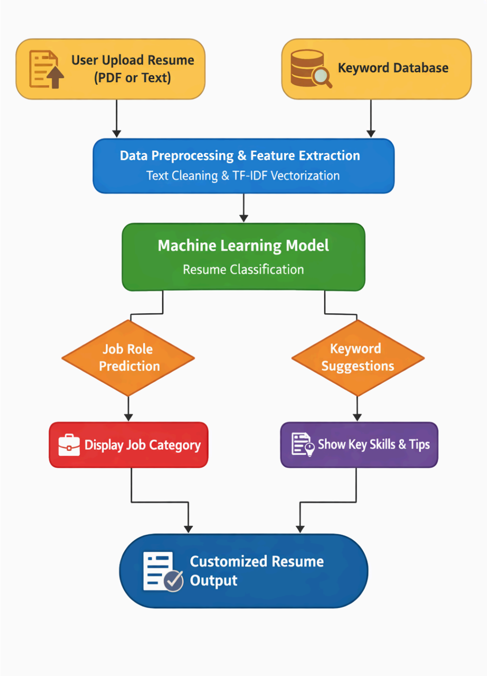

Fig. 1 shows the workflow of the proposed system.

3.3 Methodology:

The code begins by importing necessary libraries and loading pre-trained machine learning models using the pickle module. These models include a classifier and a TF-IDF vectorizer, which are essential for predicting job categories based on resume text. Additionally, the code initializes the Streamlit application and defines functions for cleaning resume text and predicting job categories.

The primary functionality of the application revolves around the resume screening page. Users can upload their resumes, which are then preprocessed to remove irrelevant elements and tokenized for feature extraction. The TF-IDF vectorizer transforms the cleaned text into numerical features, which are passed to the pre-trained classifier for prediction. The predicted job category is displayed to the user, providing insights into how their skills and experiences align with different roles.

In addition to the resume screening functionality, the application includes another page dedicated to displaying keywords for various job roles. This page serves as a reference guide for users, allowing them to explore the specific skills, qualifications, and attributes typically associated with different positions. By providing access to this information, the application empowers users to better understand the requirements of different roles and tailor their resumes accordingly to enhance their job prospects.

3.4 Model Evaluation and Persistence:

Before deployment, it's crucial to evaluate the performance of the machine learning model used for resume classification. This typically involves assessing metrics such as accuracy, precision, recall, and F1 score on a separate validation dataset. Cross-validation techniques may also be employed to ensure the model's generalization performance across different subsets of data. By rigorously evaluating the model, developers can gain insights into its strengths and weaknesses and fine-tune its parameters for optimal performance.

future use within the application. The pickle module in Python facilitates the serialization and deserialization of Python objects, allowing the trained model to be saved to disk and loaded back into memory when needed. By persisting the model, developers ensure that the application can efficiently classify resumes without the need for retraining the model each time the application is launched.

In addition to resume screening functionality, the application includes a separate page dedicated to displaying keywords for various job roles. These keywords serve as a reference guide for users, providing insights into the specific skills, qualifications, and attributes typically associated with different positions. By organizing keywords by job role, users can easily navigate and explore the requirements of different roles, empowering them to tailor their resumes effectively and improve their chances of success in the job market.

The seamless integration of model evaluation, persistence, and the keyword reference page enhances the user experience and utility of the application. Users can confidently upload their resumes, receive accurate predictions about potential job categories, and access valuable information about the keywords associated with different roles—all within a user-friendly and intuitive interface provided by Streamlit. By prioritizing usability and functionality, the application helps users make informed decisions about their career paths and stand out to potential employers in a competitive job market.

3.4.1 Table Structure:

CATEGORY	NUMBER OF RESUMES
IT	150
Marketing	70
Customer care	30

3.4.2 Integration with streamlit:

In this system, Streamlit is used to develop an interactive interface for resume upload and prediction. With Streamlit, developers can build dynamic and responsive user interfaces using familiar Python syntax, eliminating the need for complex HTML, CSS, or JavaScript code. Integration with Streamlit allows developers to focus on the application's functionality and logic, abstracting away the complexities of web development.

Streamlit offers a variety of interactive components that enhance user engagement and interactivity. Components such as file uploaders, buttons, sliders, and text inputs enable users to interact with the application, upload their resumes, and explore the predicted job categories. Streamlit's reactive programming model ensures that changes to input parameters automatically trigger updates to the application's output, providing a seamless and responsive user experience.

One of the key advantages of Streamlit is its seamless deployment process. Developers can deploy Streamlit applications using various platforms and cloud services, including Streamlit Sharing, Heroku, and AWS. Streamlit abstracts away the complexities of deployment, allowing developers to share their applications with users worldwide with just a few simple commands. This ease of deployment facilitates rapid iteration and collaboration, enabling developers to quickly iterate on their applications and incorporate user feedback to improve usability and functionality. Overall, integration with Streamlit empowers developers to build powerful and user-friendly web applications with minimal effort, accelerating the development and deployment process.

IV. RESULT

The primary outcome of the code is an interactive web application that streamlines the resume screening process. Users can effortlessly upload their resumes and receive prompt predictions regarding the job categories or roles that best match their qualifications and experiences. Additionally, the application's interface is enhanced by the presence of another dedicated page featuring keywords for various job roles. This supplementary resource empowers users to understand the specific skills and attributes sought after for different positions, enabling them to tailor their resumes more effectively. As a result, users benefit from a holistic tool that not only evaluates their resumes but also provides valuable insights to optimize their job search strategies. This integrated approach enhances user engagement and facilitates informed decision-making, ultimately improving the efficiency and effectiveness of the job application process.

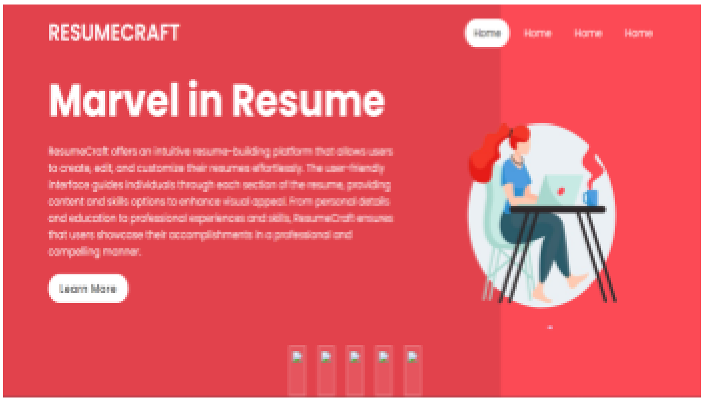

Fig. 2 represents the sample output of Resumecraft using NPL and ML.

V. CONCLUSION

In conclusion, the provided code presents a robust and user-friendly solution for resume screening and job role identification, augmented by an additional page containing comprehensive keyword references for various roles.

The resume screening application offers users a seamless experience by enabling them to upload their resumes and receive immediate predictions regarding the most suitable job categories or roles. Leveraging natural language processing and machine learning techniques, the application accurately analyzes resume content and provides valuable insights into potential career paths. This functionality not only expedites the job search process but also empowers users to make informed decisions about their professional aspirations.

Furthermore, the inclusion of the keyword reference page enhances the utility and versatility of the application. Users have access to a wealth of information regarding the specific skills, qualifications, and attributes associated with different roles. This resource serves as a valuable guide, allowing individuals to tailor their resumes to align with industry expectations and stand out to potential employers. By offering comprehensive keyword references, the application equips users with the knowledge and tools needed to navigate the competitive job market effectively.

Moreover, the seamless integration of the resume screening functionality and the keyword reference page underscores the application's commitment to user-centric design and functionality. Streamlit's intuitive interface and interactive components facilitate a smooth and engaging user experience, while the underlying machine learning models and data preprocessing techniques ensure accurate and reliable results. Through thoughtful integration and robust functionality, the application empowers users to optimize their job search efforts and pursue rewarding career opportunities with confidence.

In summary, the combination of advanced technology, user-friendly design, and comprehensive resources makes the resume screening application a valuable asset for job seekers. By leveraging the power of data-driven insights and tailored recommendations, the application facilitates meaningful connections between candidates and potential employers, ultimately fostering career growth and success.

Future investigations should focus on resolving the shortcomings of our analysis and investigating strategies to raise the models' level of accuracy and utility.

VI. ACKNOWLEDGEMENT

I express my sincere gratitude to Francis Xavier Engineering College for providing the necessary support, facilities and guidance to successfully carry out this research work. The institutional environment and resources greatly contributed to the completion of this project.

References

[1] T. Mitchell, Machine Learning, McGraw-Hill Education, 1997.

[2] C. D. Manning, P. Raghavan and H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008.

[3] S. Bird, E. Klein and E. Loper, Natural Language Processing with Python, O'Reilly Media, 2009.

[4] D. Jurafsky and J. H. Martin, Speech and Language Processing, 2nd ed., Pearson Education, 2009.

[5] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, 2016.

[6] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[7] F. Sebastiani, "Machine learning in automated text categorization," ACM Computing Surveys, vol. 34, no. 1, pp. 1–47, 2002.

[8] A. McCallum and K. Nigam, "A comparison of event models for Naive Bayes text classification," in Proc. AAAI Workshop, 1998, pp. 41–48.

[9] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space," arXiv preprint arXiv:1301.3781, 2013.

[10] J. Pennington, R. Socher and C. Manning, "GloVe: Global vectors for word representation," in Proc. EMNLP, 2014, pp. 1532–1543.

[11] D. Blei, A. Ng and M. Jordan, "Latent Dirichlet Allocation," Journal of Machine Learning Research, vol. 3, pp. 993–1022, 2003.

[12] N. Srivastava et al., "Dropout: A simple way to prevent neural networks from overfitting," Journal of Machine Learning Research, vol. 15, pp. 1929–1958, 2014.

[13] I. Sutskever, O. Vinyals and Q. Le, "Sequence to sequence learning with neural networks," in Advances in Neural Information Processing Systems, 2014, pp. 3104–3112.

[14] C. Manning et al., "The Stanford CoreNLP natural language processing toolkit," in Proc. ACL System Demonstrations, 2014, pp. 55–60.

[15] "Resume Dataset for Classification," Kaggle, Available: https://www.kaggle.com. Accessed: Apr. 15, 2026.

[16] "Natural Language Toolkit (NLTK) Documentation," Available: https://www.nltk.org. Accessed: Apr. 15, 2026.

[17] "Scikit-learn Documentation," Available: https://scikit-learn.org. Accessed: Apr. 15, 2026.

[18] "Streamlit Documentation," Available: https://streamlit.io. Accessed: Apr. 15, 2026.

[19] "TF-IDF Explanation and Applications," Available: https://towardsdatascience.com. Accessed: Apr. 15, 2026.

[20] "Applicant Tracking Systems Overview," Available: https://www.ibm.com. Accessed: Apr. 15, 2026.