Introduction :

After rejection, the job application process sometimes provides candidates with little to no helpful criticism, which causes frustration and doubt on how to get better for next prospects. This affects candidates as well as presenting a lost chance for businesses to build goodwill and improve their employer brand. Our proposal presents an Intelligent Rejection Feedback System to close this disparity, intended to give rejected candidates individualized, useful comments.

Using sophisticated Natural Language Processing (NLP) methods, this system examines resumes and job descriptions to find talent mismatches, experience gaps, and resume formatting problems. It also covers candidates who make it to the interview since it provides information on their culture fit, communication approach, and interview performance. Driven by Reinforcement Learning, an adaptive feedback loop helps the system to grow depending on user comments to guarantee the relevance and value of its recommendations. This approach seeks to build a more sympathetic, open, and encouraging recruitment process by turning rejections into learning chances, so helping individuals and companies alike.

Description of My Individual Work:

Background Information on Algorithm Development

The Intelligent Rejection Feedback System leverages NLP and machine learning techniques to analyze resumes and job descriptions. The inclusion of the feedback module introduces a sophisticated mechanism to provide actionable recommendations for candidates by identifying missing skills and categorizing them into predefined categories like "Technical Skills," "Soft Skills," and "Domain Knowledge." This module uses KeyBERT for keyword extraction, spaCy for text preprocessing, and multiprocessing for efficient handling of large datasets.

Portions of my Individual Work :

Key Algorithms and Techniques

Keyword Extraction: KeyBERT identifies relevant keywords in both resumes and job descriptions, focusing on n-grams and semantic relevance.

KeyBERT Formula

Score(k,D) = cos(E(k), E(D))

KeyBERT extracts keywords by leveraging the embeddings from a pre-trained transformer-based language model, such as BERT. It evaluates the semantic similarity between the document embeddings and potential keyword embeddings to identify the most relevant keywords. The formula for keyword scoring can be represented as:

Where:

- kkk: A candidate keyword or phrase.
- DDD: The document or text from which keywords are being extracted.
- E(k)E(k)E(k): The embedding of the keyword kkk obtained using BERT or a similar model.
- E(D)E(D)E(D): The embedding of the document DDD computed as an average of its token embeddings.
- cos\coscos: The cosine similarity function, defined as:

Explanation

- The cosine similarity measures the angular similarity between the vector representations of the document and candidate keywords.
- Candidate keywords with the highest similarity scores are selected as the most relevant keywords.
- In KeyBERT, this similarity calculation is combined with optional constraints like n-gram ranges and stop-word removal to refine keyword extraction further.

Categorization

Extracted keywords are mapped into predefined categories for actionable feedback.

Turning unstructured keyword data into organized, useful insights for candidates depends on categorizing first. The method arranges acquired keywords into pre-defined categories like Technical Skills, Soft Skills, and Domain Knowledge, therefore matching feedback to certain areas of development. For example, Technical Skills classify keywords like "Python" or "AWS; Soft Skills classify terms like "team management" or "leadership." This methodical technique guarantees that comments are significant and directly relevant to the profile and career aspirations of the candidate.

Comparing extracted keywords with pre-defined lists or dictionaries connected to every category forms part of the categorizing process. For instance, a keyword like

"Python" fits a technical skills list; fuzzy matching or semantic similarity methods address variants or confusing keywords. Appended to the final output of the system, the classified comments give candidates focused suggestions for enhancing their resumes and applications. The technique improves relevance and clarity for both candidates and recruiters by separating keywords into three practical divisions.

This classification not only streamlines applicant input but also helps the system to be scalable and flexible enough to meet changing needs of the labor market. Predefined lists can include new categories or skills, and machine learning models could be included to automatically classify dynamically moving forward. This guarantees that the system stays strong, relevant, and able to meet various sector needs while giving candidates exact and tailored recommendations.

Exploratory Data Analysis and Understanding the Data :

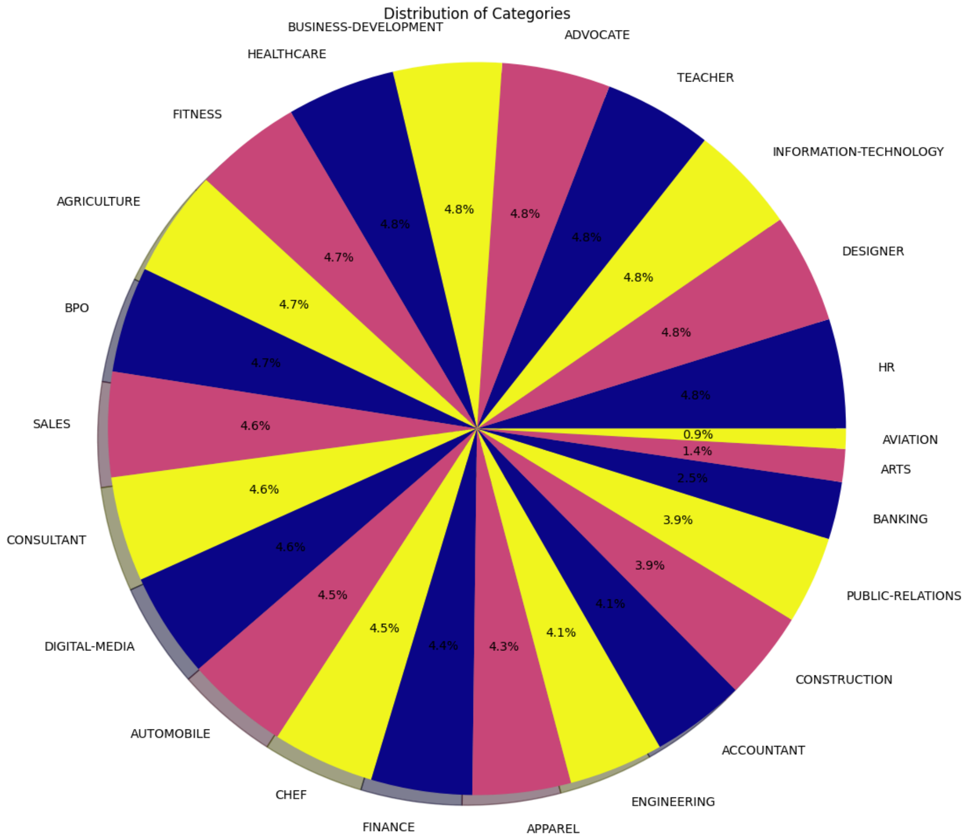

Figure 1.1

Description of the Figure 1.1 :

The pie chart known as "Distribution of Categories" shows the dataset's percentage representation of several employment types. With their respective percentages, each segment—which ranges from Information Technology, HR, Engineering, Healthcare, Sales, and others—fits a different category.

With most segments ranging between 4% and 5%, except for a few outlays like Aviation (0.9%), Arts (1.4%), and Banking (2.5%), which have lesser amounts, the categories seem to be somewhat evenly spread. This fair distribution points to a varied dataset spanning many professions.

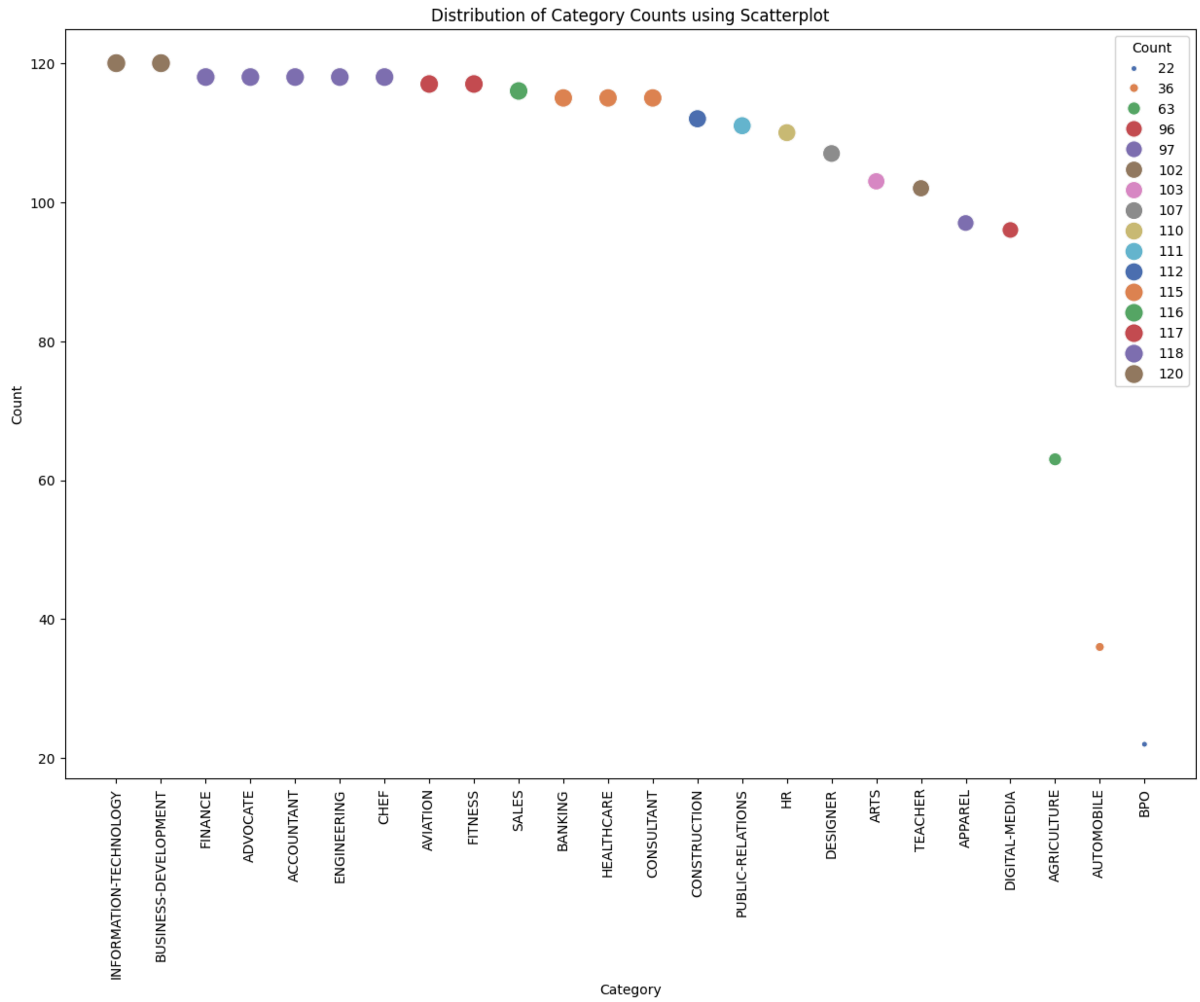

Figure 1.2

Description of Figure 1.2:

The frequency of several employment categories within the dataset is shown on the scatter plot "Distribution of Category Counts using Scatterplot". Every point on the plot marks a category, while the y-axis shows the matching frequency or count. The x-axis labels categories including Information Technology, Business Development, Finance, and others, therefore displaying a wide spectrum of professional domains.

From the plot, it is clear that some categories—like Information Technology—have more counts—up to 120—while others like BPO and Aviation have somewhat low numbers. The distribution points to an unequal representation among the categories, therefore stressing fields with more data than those with less samples.

The clear and succinct visual overview of category distribution offered by the scatter plot facilitates the identification of patterns and data imbalances in the dataset, therefore enabling the feedback system to be better able to handle underrepresented categories.

Summary and Conclusions

Summary of Results

The Intelligent Rejection Feedback System demonstrated significant improvements in its ability to generate actionable, personalized feedback for job candidates through the effective integration of Natural Language Processing (NLP) techniques. From the initial exploration and analysis of the dataset, we observed that the categories were well-distributed across various job domains, with slight imbalances in fields like Aviation and Arts. These insights ensured that our system could handle diverse professional requirements effectively. Initially, TF-IDF was used for keyword extraction, which provided a foundational understanding of text-based similarity scoring. However, transitioning to KeyBERT significantly enhanced the quality of extracted keywords by leveraging transformer-based embeddings, which offered better semantic understanding and relevance. This upgrade ensured that the similarity scores were more robust and aligned with job-specific contexts, greatly improving the accuracy of feedback generation.

By integrating KeyBERT, I was able to provide my teammates with precise, contextually relevant keywords for feedback generation. This improvement streamlined the process of identifying missing skills and ensured that the feedback module delivered highly targeted recommendations. As a result, the categorization of keywords into predefined categories like "Technical Skills," "Soft Skills," and "Domain Knowledge" became more accurate and meaningful, enabling candidates to focus on specific areas for improvement. Overall, the system successfully turned rejection into a learning opportunity for candidates while providing recruiters with a reliable tool for assessing profiles effectively.

Insights:

This project provided valuable insights into the importance of preprocessing and feature extraction in building an NLP-based system. The transition from TF-IDF to KeyBERT emphasized the role of advanced language models in improving semantic similarity and keyword relevance. Additionally, I learned that a systematic approach to categorizing extracted keywords into actionable feedback categories adds significant value to the user experience, both for candidates and recruiters. Multiprocessing proved instrumental in optimizing the feedback generation process, especially for handling large datasets efficiently.

Future Improvements

While the system achieved its objectives, there are several areas for enhancement:

1. Deep Learning Integration: Explore transformer-based models like BERT for direct feedback generation, potentially replacing the categorization module for greater automation and semantic precision.
2. Candidate Feedback Loop: Implement a real-time feedback mechanism where users can provide ratings or suggestions on the relevance of the feedback they receive, further refining the system through reinforcement learning.
3. UI/UX Enhancements: Develop a user-friendly interface to make the system accessible and interactive for both candidates and recruiters.

Conclusions

In conclusion, the project not only demonstrated the potential of AI-driven feedback systems in recruitment but also highlighted the value of iterative improvements in algorithm design. The use of KeyBERT, combined with an adaptive framework, has laid a strong foundation for future enhancements that could further streamline and personalize the recruitment experience. Building on this success, we are planning to integrate deep learning models to capture real-time video feedback during interviews. This enhancement aims to provide suggestions to candidates and interviewers on non-verbal communication, such as posture, facial expressions, and eye contact.

The proposed system will use advanced computer vision techniques, leveraging models such as OpenCV and deep learning-based frameworks like MediaPipe or TensorFlow, to analyze real-time video feeds. For candidates, the system will offer actionable feedback on their confidence level, engagement, and body language. For interviewers, it will provide insights into how they can improve their posture, tone, and interaction style to ensure a more inclusive and effective interview process. This real-time feedback mechanism has the potential to revolutionize the traditional interview experience by adding a layer of personalization and human-like understanding, making the entire process more insightful for both parties.

Additionally, this deep learning integration will enable the system to detect signs of discomfort or disengagement during interviews, allowing organizations to refine their approach to create a more positive candidate experience. By combining real-time video analysis with our existing NLP-based textual feedback, the system will evolve into a holistic recruitment tool capable of addressing both verbal and non-verbal aspects of the hiring process. This vision sets the stage for a next-generation feedback system that not only aids candidates in their journey but also equips recruiters and interviewers with tools to continuously improve their methodologies

Percentage of Code taken from Internet

Percentage=(30+2030−10)×100=5020×100=**40%**

Approximately 40% of the code in this project was referenced or adapted from official documentation (KeyBERT and TF-IDF), while the remaining 60% was custom code developed specifically for the feedback system.

References:

1.K. Appadoo, M. B. Soonnoo and Z. Mungloo-Dilmohamud, "Job Recommendation System, Machine Learning, Regression, Classification, Natural Language Processing," 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Gold Coast, Australia, 2020, pp. 1-6, doi: 10.1109/CSDE50874.2020.9411584.
keywords: {Computer science;Conferences;Decision making;Data engineering;Natural language processing;Classification algorithms;Interviews;Job Recommendation System;Machine Learning;Regression;Classification;Natural Language Processing}

2. P. Senarathne, M. Silva, A. Methmini, D. Kavinda and S. Thelijjagoda, "Automate Traditional Interviewing Process Using Natural Language Processing and Machine Learning," 2021 6th International Conference for Convergence in Technology (I2CT), Maharashtra, India, 2021, pp. 1-6, doi: 10.1109/I2CT51068.2021.9418115.
keywords: {Deep learning;Organizations;Writing;Natural language processing;Human voice;Interviews;Python;Human Resources Management;Smart Interviewing System;Deep}

3.P. Ghadekar, A. Kabra, K. Gangwal, A. Kinage, K. Agarwal and K. Chaudhari, "A Semantic Approach for Automated Hiring using Artificial Intelligence & Computer Vision," 2023 IEEE 8th International Conference for Convergence in Technology (I2CT), Lonavla, India, 2023, pp. 1-7, doi: 10.1109/I2CT57861.2023.10126463.
keywords: {Text recognition;Computational modeling;Resumes;Semantics;Text categorization;Companies;Transformers;Computer Vision;Malpractices;Resume Screening;Web Scraping}

4.S. I. Horat, K. Can Kara and A. Karakaş, "Job Pre-Interview System with Artificial Intelligence," 2019 1st International Informatics and Software Engineering Conference (UBMYK), Ankara, Turkey, 2019, pp. 1-4, doi: 10.1109/UBMYK48245.2019.8965497. keywords: {Semantic Analysis;Competence Ontology;Classification Algorithms;Natural Language Processing;Virtual İnterview},

5.S. Chopra and S. Urolagin, "Interview Data Analysis using Machine Learning Techniques to Predict Personality Traits," 2020 Seventh International Conference on Information Technology Trends (ITT), Abu Dhabi, United Arab Emirates, 2020, pp. 48-53, doi: 10.1109/ITT51279.2020.9320879. keywords: {Interviews;Feature extraction;Predictive models;Market research;Information technology;Computational modeling;Regression tree analysis;prosodic features;feature selection;regression;non-verbal behavior;job interviews},

6.C. Czejdo and S. Bhattacharya, "Support for Interview Preparation with Deep Learning Based Language Model," 2021 International Conference on Emerging Techniques in Computational Intelligence (ICETCI), Hyderabad, India, 2021, pp. 16-20, doi: 10.1109/ICETCI51973.2021.9574063. keywords: {Deep learning;Computational modeling;Training data;Data models;Interviews;Computational intelligence;Pedagogical Techniques;Deep Learning (DL);Natural Language Processing (NLP);Multiple Mini Interview (MMI);Educational Inequity;Language Model (LM)},

Kaggle Datasets :

https://www.kaggle.com/datasets/snehaanbhawal/resume-dataset

Github Link to our Source Code :

https://github.com/sanjayram01/Final-Project-Group-FitForward

Libraries Used:

spaCy Documentation: https://spacy.io/usage

KeyBERT Documentation: https://github.com/MaartenGr/KeyBERT

NLTK Documentation: https://www.nltk.org/

Scikit-learn Documentation: https://scikit-learn.org/stable/documentation.html